European Community on Computational Methods in Applied Sciences

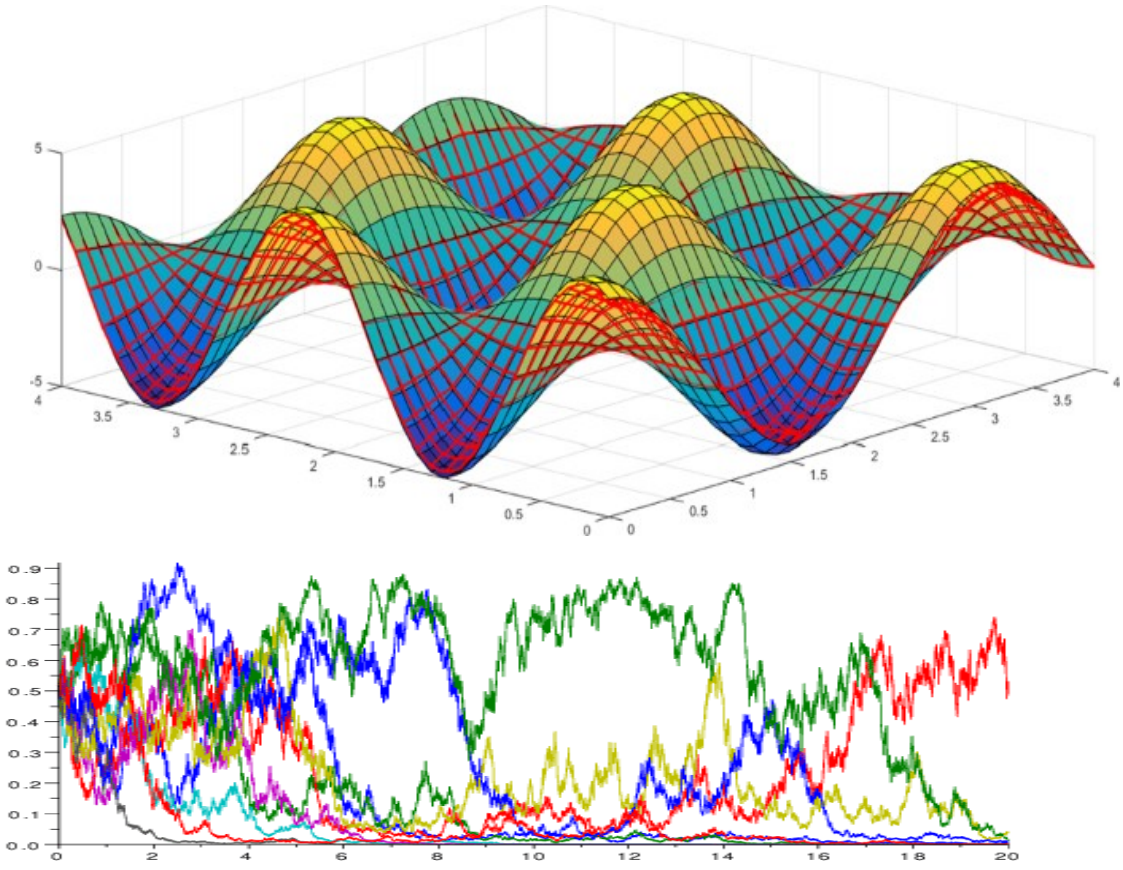
ECCOMAS Thematic Conference

UNCECOMP 2025

6th International Conference on Uncertainty Quantification in Computational Science and Engineering

PROCEEDINGS

M. Papadrakakis, V. Papadopoulos, G. Stefanou (Eds.)



UNCECOMP 2025

Uncertainty Quantification in Computational Science and Engineering

Proceedings of the 6th International Conference on Uncertainty
Quantification in Computational Science and Engineering
Held in Rhodes Island, Greece
15-18 June 2025

Edited by:

M. Papadrakakis
National Technical University of Athens, Greece

V. Papadopoulos
National Technical University of Athens, Greece

G. Stefanou
Aristotle University of Thessaloniki, Greece

A publication of:

Institute of Structural Analysis and Antiseismic Research
School of Civil Engineering
National Technical University of Athens (NTUA)
Greece

UNCECOMP 2025
Uncertainty Quantification in Computational Science and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (Eds.)

First Edition, November 2025

© The authors

ISBN: **978-618-5827-03-8**

PREFACE

This volume contains the full-length papers presented in the International Conference on Uncertainty Quantification in Computational Science and Engineering (UNCECOMP 2025) that was held in Rhodes Island, Greece on June 15-18, 2025.

UNCECOMP 2025 is a Thematic Conference of ECCOMAS, with the objective to reflect the recent research progress in the field of analysis and design of engineering systems under uncertainty, with emphasis in multiscale simulations. The aim of the conference is to enhance the knowledge of researchers in stochastic methods and the associated computational tools for obtaining reliable predictions of the behavior of complex systems. The UNCECOMP conference series, held in conjunction with the COMPDYN conferences, gives the opportunity to the participants to interact with the Computational Dynamics community for their mutual benefit.

The UNCECOMP 2025 Conference is supported by the National Technical University of Athens (NTUA) and the Greek Association for Computational Mechanics (GRACM).

The editors of this volume would like to thank all authors for their contributions. Special thanks go to the colleagues who contributed to the organization of the Minisymposia and to the reviewers who, with their work, contributed to the scientific quality of this e-book.

Editors

M. Papadrakakis
National Technical University of Athens, Greece

V. Papadopoulos
National Technical University of Athens, Greece

G. Stefanou
Aristotle University of Thessaloniki, Greece

ACKNOWLEDGEMENTS

The conference organizers acknowledge the support towards the organization of the "6th International Conference on Uncertainty Quantification in Computational Science and Engineering", to the following organizations:

- European Community on Computational Methods in Applied Sciences (ECCOMAS)
- European Committee on Computational Solids and Structural Mechanics of ECCOMAS (ECCSM)
- Greek Association for Computational Mechanics (GRACM)
- School of Civil Engineering, National University of Athens (NTUA)

Plenary Speakers and Invited Session Organizers

We would also like to thank the Plenary and Semi-Plenary Speakers and the Minisymposia Organizers for their help in the setting up of a high standard Scientific Programme.

Plenary and Semi-Plenary Speakers: Manfred Bischoff, Ludovic Chamoin, Eleni Chatzi, Laura De Lorenzis, Geert Degrande, Matthew DeJong, Alireza Doostan, De-Cheng Feng, Roger Ghanem, Muneo Hori, Boris Jeremic, Marc P. Mignolet, Giuseppe Muscolino, Udo Nackenhorst, Martin Ostoja-Starzewski, Costas Papadimitriou, Iason Papaioannou, K. C. Park, Alessandro Reali, Gianluigi Rozza, Alexandros Taflanidis, Enrico Tubaldi, Marcos Valdebenito, Wolfgang Wall.

MS Organizers: Joseph Bakarji, Michael Beer, Jean-Marc Bourinet, Sebastian Brandstäter, Matteo Broggi, Eleni Chatzi, Manolis Chatzis, Jian-Bing Chen, Alice Cicirello, Marco Colatosti, Elizabeth Cross, Vasilis Dertimanis, Matthias Faes, De-Cheng Feng, M. Giselle Fernández-Godino, Gianluca Geraci, Dimitris Giovanis, Ling Guo, Michael Hanss, Wouter Huberts, Elisa Iacomini, Oindrila Kanjilal, George Em Karniadakis, Denise-Penelope N. Kontoni, Geert Lombaert, Dimitrios Loukrezis, Emil Løvbak, Meng-Ze Lyu, Stefano Marelli, Gian Marco Melito, David Moens, Greta Ongaro, Yannis Pantazis, Costas Papadimitriou, Iason Papaioannou, Edoardo Patelli, Paris Perdikaris, Dirk Pflüger, Marco Pingaro, Elizabeth Qian, Bojana Rosić, Ambrosios Antonios Savvides, Friederike Schäfer, Jakob Scheffels, Daniele E. Schiavazzi, Michael Shields, Alba Sofi, Gowri Srinivasan, George Stefanou, Panos Stinis, Daniel Straub, Bruno Sudret, Alexandros Taflanidis, Patrizia Trovalusci, Marcos Valdebenito, Daniele Venturi, Miroslav Vořechovský.

SUMMARY

Preface..... iii

Acknowledgements..... iv

Contents..... vii

Minisymposia

MS 1: UNCERTAINTY QUANTIFICATION, RELIABILITY, AND SENSITIVITY ANALYSIS UNDER LIMITED DATA 1
Organized by Matthias Faes, David Moens, Michael Hanss, Alba Sofi and Edoardo Patelli

MS 2: PHYSICS-ENHANCED MACHINE LEARNING IN ENGINEERING 112
Organized by Alice Cicirello, Elizabeth Cross and Eleni Chatzi

MS 3: METHODOLOGICAL DEVELOPMENTS AND APPLICATIONS OF GLOBAL SENSITIVITY ANALYSIS IN SCIENCE AND ENGINEERING 122
Organized by Gian Marco Melito, Sebastian Brandstäter, Friederike Schäfer and Wouter Huberts

MS 5: UNCERTAINTY PROPAGATION AND PROBABILISTIC PERFORMANCE ASSESSMENT IN ENGINEERING STRUCTURES 137
Organized by Meng-Ze Lyu, De-Cheng Feng, Jian-Bing Chen and Michael Beer

MS 6: UNCERTAINTY QUANTIFICATION AND SCIENTIFIC MACHINE LEARNING FOR RELIABLE DATA-DRIVEN MODELING AND SIMULATION IN SCIENCE AND ENGINEERING 164
Organized by Dimitris Giovanis, Bojana Rosić, Dimitrios Loukrezis and Miroslav Vořechovský

MS 7: SURROGATE MODELS FOR UNCERTAINTY QUANTIFICATION: NEW TRENDS 184
Organized by Jean-Marc Bourinet, Michael Shields, Bruno Sudret and Alexandros Taflanidis

MS 8: UNCERTAINTY QUANTIFICATION IN VIBRATION BASED MONITORING AND STRUCTURAL DYNAMICS SIMULATIONS 222
Organized by Eleni Chatzi, Manolis Chatzis, Vasilis Dertimanis, Geert Lombaert and Costas Papadimitriou

MS 9: SOFTWARE FOR UNCERTAINTY QUANTIFICATION 293
Organized by Matteo Broggi, Stefano Marelli, Edoardo Patelli and Dirk Pflüger

MS 12: MULTISCALE AND MULTIPHYSICS MODELLING FOR COMPLEX MATERIALS (MMCM 21) 357
Organized by Marco Pingaro, Patrizia Trovalusci, George Stefanou, Greta Ongaro and Marco Colatosti

MS 14: OPTIMIZATION UNDER UNCERTAINTY 416
Organized by Oindrila Kanjilal, Iason Papaioannou, Daniel Straub and Marcos Valdebenito

MS 17: BAYESIAN COMPUTATION METHODS FOR INFERENCE IN SCIENCE AND ENGINEERING 430
Organized by Oindrila Kanjilal, Iason Papaioannou, Daniel Straub, Geert Lombaert and Costas Papadimitriou

MS 19: UNCERTAINTY QUANTIFICATION AND MACHINE LEARNING APPLICATIONS IN SCIENCES AND ENGINEERING 476
Organized by Ambrosios Antonios Savvides and Denise-Penelope N. Kontoni

MS 20: ADVANCING SCIENTIFIC DISCOVERY WITH SCIML: NEURAL OPERATORS, GENERATIVE MODELING, AND PHYSICS-INFORMED TECHNIQUES 511
Organized by Daniele Venturi, Panos Stinis and Paris Perdikaris

Thematic Sessions

TS 16: SYSTEM RELIABILITY ANALYSIS, DESIGN AND RISK ASSESSMENT 523

TS 17: UNCERTAINTY QUANTIFICATION 537

CONTENTS

Minisymposia

MS 1: UNCERTAINTY QUANTIFICATION, RELIABILITY, AND SENSITIVITY ANALYSIS UNDER LIMITED DATA

HETEROSCEDASTIC GAUSSIAN PROCESS THROUGH KERNEL SMOOTHING…… 1
Ghifari Adam Faza, Nasrulloh Ratu Bagus Satrio Loka, David Moens

AN EMPIRICAL EVALUATION OF INTERVAL NEURAL NETWORKS FOR UNCERTAINTY QUANTIFICATION ON REGRESSION TASKS…… 16
Kaizheng Wang, Ghifari Adam Faza, Keivan Shariatmadar, David Moens, Hans Hallez

FORWARD UNCERTAINTY PROPAGATION FOR FINITE ELEMENT MODELS WITH NON-GAUSSIAN PARAMETERS…… 31
Renan Barros, Iuri Rocha, Maria Nogal, Oswaldo Morales-Napoles

POSSIBILISTIC NEURAL NETWORKS: RELIABLE CONFIDENCE PREDICTIONS UNDER LIMITED DATA …… 43
Jan Schneider, Tom Könecke, Michael Hanss

BAYESIAN ROBUST DESIGN OPTIMIZATION: AN EXTREME VALUE APPROACH…… 56
Augustin Persoons, Pascal Lafon, David Moens

EFFICIENT CONFIDENCE BOUND ESTIMATION FOR A DYNAMIC COMPOSITE PLATE WITH EPISTEMIC UNCERTAINTIES USING NON-INTRUSIVE IMPRECISE STOCHASTIC SIMULATION…… 72
Ewerton Grotti, Matteo Broggi, Herbert Martins Gomes, Michael Beer

CAPACITY BOUNDS OF UNREINFORCED MASONRY WALLS WITH INTERVAL MECHANICAL PROPERTIES …… 84
Alba Sofi, Elyas Bayat, Federica Tubino

LEVERAGING PHYSICS-INFORMED MODELS AND PRODUCTION DATA FOR QUALITY ASSESSMENT IN RESISTANCE SPOT WELDING…… 93
Miriam Dodt, Bouwe Verkens, Augustin Persoons, Matthias Faes, David Moens

MS 2: PHYSICS-ENHANCED MACHINE LEARNING IN ENGINEERING

PHYSICALLY-INFORMED CHANGE-POINT KERNELS FOR INTERMITTENT INCLUSION OF PHYSICAL KNOWLEDGE …… 112
Daniel Pitchforth, Samuel Gibson, Elizabeth Cross

MS 3: METHODOLOGICAL DEVELOPMENTS AND APPLICATIONS OF GLOBAL SENSITIVITY ANALYSIS IN SCIENCE AND ENGINEERING

PREDICTING BUILDING OPERATIONAL ENERGY UNDER MATERIAL DEGRADATION AND CLIMATE UNCERTAINTY: A SENSITIVITY ANALYSIS…… 122
Anna Maria Koniari, Charalampos Andriotis, Simona Bianchi, Pablo Morato, Seyran Khademi, Mauro Overend

MS 5: UNCERTAINTY PROPAGATION AND PROBABILISTIC PERFORMANCE ASSESSMENT IN ENGINEERING STRUCTURES

PROPAGATING ML MODEL UNCERTAINTIES IN OFFSHORE WIND TURBINE FATIGUE DAMAGE ACCUMULATION…… 137
Francisco de Nolasco Santos, Nandar Hlaing, Wout Weijtjens, Christof Devriendt

DATA DRIVEN METHODS FOR UNCERTAINTY PROPAGATION IN PREDICTING BRITTLE RESPONSES OF REINFORCED CONCRETE STRUCTURES…… 148
Afaq Ahmad, Vagelis Plevris

MS 6: UNCERTAINTY QUANTIFICATION AND SCIENTIFIC MACHINE LEARNING FOR RELIABLE DATA-DRIVEN MODELING AND SIMULATION IN SCIENCE AND ENGINEERING

UNCERTAINTY QUANTIFICATION IN LASER POWDER BED FUSION USING MULTI-PHYSICS NUMERICAL SIMULATION AND COPULA-BASED ANALYSIS…… 164
Yunfan Zhang, Maria Nogal, Iuri Rocha, Oswaldo Morales-Napoles

MS 7: SURROGATE MODELS FOR UNCERTAINTY QUANTIFICATION: NEW TRENDS

UNCERTAINTY QUANTIFICATION FOR THE SIMULATION OF AUTOMOTIVE COMPONENT LOADS USING STOCHASTIC SURROGATE MODELS…… 184
Paul Strähle, Steve Wolff-Vorbeck, Thomas Leyendecker, Carsten Proppe

COMBINING NEURAL NETWORKS WITH GAUSSIAN PROCESSES FOR ROBUST AND INTERPRETABLE PREDICTION…… 198
Amalia Macali, Georgios Soimoiris, Nathan Eskue, Vahid Yaghoubi

PHYSICS-INFORMED NEURAL NETWORKS COMBINED WITH POLYNOMIAL CHAOS EXPANSION TO PROPAGATE RANDOM FIELD PROPERTIES OF COMPOSITES…… 207
Damien Bonnet-Eymard, Augustin Persoons, Panagiotis Gavallas, Matthias Faes, George Stefanou, David Moens

MS 8: UNCERTAINTY QUANTIFICATION IN VIBRATION BASED MONITORING AND STRUCTURAL DYNAMICS SIMULATIONS

BRIDGE MODEL UPDATING USING MONITORED SEISMIC RESPONSES: INSIGHTS FROM A NUMERICAL PARAMETRIC STUDY…… 222
Pooria Mesbahi, Enrique García-Macías, Marco Breccolotti, Filippo Ubertini

A GENERIALIZED CONTACT POINT RESPOSNE ESTIMATION VIA ON-BOARD SENSING…… 240
Ho Man Siu, Elias Dimitrakopoulos, Costas Papadimitriou

OPTIMAL SENSOR PLACEMENT FOR VIBRATION MONITORING OF BRIDGE STRUCTURES…… 253
Daniele Giunta, Eugenio Chioccarelli, Eleni Chatzi

COMBINING EXPERIMENTAL DATA AND NUMERICAL MODELING TO ANALYZE TRAIN-STRUCTURE INTERACTION ON THE HISTORIC SAN MICHELE BRIDGE (ITALY, 1889)..... 266
Lorenzo Ermolli, Michele Guerini, Charikleia Stoura, Rosalba Ferrari, Vasilis Dertimanis, Eleni Chatzi, Egidio Rizzi

UPDATING DIGITAL TWINS SUBJECTED TO UNCERTAIN CORROSION CONDITIONS USING THE UNSCENTED KALMAN FILTER..... 278
Ioannis Prentzas, Spyridon Diamantopoulos, Alexandros Michailidis, Michalis Fragiadakis

MS 9: SOFTWARE FOR UNCERTAINTY QUANTIFICATION

RISQ: RANDOM INTERVAL SPATIAL UNCERTAINTY QUANTIFICATION SOFTWARE..... 293
Ben Van Bavel, Robin Callens, David Moens

BAYESIAN INVERSE UNCERTAINTY QUANTIFICATION FOR COMPUTATIONAL ENGINEERING USING SIMULATION-BASED INFERENCE..... 301
Maternus Herold, Jan Boelts

POLYUQ: A SOFTWARE TOOL FOR THE SIMULTANEOUS CONSIDERATION OF ALEATORY AND EPISTEMIC UNCERTAINTIES IN ENGINEERING PROBLEMS..... 318
Martin Drieschner

MULTI-DISCIPLINARY DESIGN OPTIMIZATION UNDER UNCERTAINTY: THE OPEN SOURCE CAPABILITIES OF GEMSEO..... 331
Matthias De Lozzo, Clément Laboulfie, Olivier Sapin, Nicolas Roussouly, François Gallard, Amine Aziz-Alaoui, Antoine Dechaume, Anne Gazaix

MS 12: MULTISCALE AND MULTIPHYSICS MODELLING FOR COMPLEX MATERIALS (MMCM 21)

A MACHINE LEARNING APPROACH FOR A2-POINT STATISTICAL HOMOGENIZATION METHOD..... 357
Luzie Schmollack, Sandra Klinge

STOCHASTIC FINITE ELEMENT ANALYSIS OF CONCRETE BEAMS..... 364
Leontios Kotropoulos, George Stefanou

MULTISCALE STRUCTURAL OPTIMIZATION WITH EMBEDDED HIERARCHICAL MATERIAL ARCHITECTURES 383
Andrea Chiozzi, Andrea Nale

ACOUSTIC WAVE TRANSMISSION OF MICROPOLAR PLATES..... 391
Annamaria Pau, Christian Meso, Zine El Abiddine Fellah, E. Ogam

SECTIONAL HOMOGENIZATION WITH A GENERAL NONLINEAR CONSTITUTIVE LAW FOR CORRUGATED BOARD ANALYSIS..... 399
Tomasz Garbowski, Anna Szymczak-Graczyk, Aram Cornaggia

ACTIVE LEARNING ENHANCED STOCHASTIC SIZE-DEPENDENT DYNAMIC ANALYSIS OF THE PEROVSKITE SOLAR CELL…… 408
Luo Bo, Jize Zhang

MS 14: OPTIMIZATION UNDER UNCERTAINTY

STOCHASTIC OPTIMIZATION FOR SPACE MISSION ANALYSIS…… 416
Thomas Caleb, Roberto Armellin, Spencer Boone, Stéphanie Lizy-Destrez

MS 17: BAYESIAN COMPUTATION METHODS FOR INFERENCE IN SCIENCE AND ENGINEERING

BAYESIAN CALIBRATION OF A NON-LINEAR CONTROLLED DYNAMIC SYSTEM WITH MISSING DATA: APPLICATION TO HIGH SPEED-TRAIN…… 430
Romain Jorge Do Marco, Guillaume Perrin, Christine Funfschilling, Christian Soize

VIRTUAL LOAD MONITORING OF OFFSHORE WIND FARMS VIA UNCERTAINTY-AWARE DEEP LEARNING MODELS…… 438
Nandar Hlaing, Pablo Morato, Francisco de Nolasco Santos, Wout Weijtjens, Christof Devriendt

BAYESIAN ESTIMATION OF SEISMIC FRAGILITY CURVES BASED ON VARIATIONAL REFERENCE PRIORS USING NEURAL NETWORKS…… 453
Nils Baillie, Antoine Van Biesbroeck, Cyril Feau, Clément Gauchy

MS 19: UNCERTAINTY QUANTIFICATION AND MACHINE LEARNING APPLICATIONS IN SCIENCES AND ENGINEERING

LABEL-EFFICIENT CARDIAC VOLUME ESTIMATION IN REAL-TIME MRI USING SPARSE BAYESIAN LEARNING…… 476
Felix Terhag, Philipp Knechtges, Anja Bach, Darius Gerlach, Jens Tank, Achim Basermann, Raúl Tempone

IMAGE-WISE UNCERTAINTY QUANTIFICATION FOR MEDICAL IMAGING…… 490
Philipp Rosauer, Angelina Händeler, Felix Terhag, Alexander Rüttgers, Michael Felderer

COUPLED CANN-DEM SIMULATION FOR HYPERELASTIC SOLIDS…… 500
Stefan Hildebrand, Jonathan Georg Friedrich, Sandra Klinge

MS 20: ADVANCING SCIENTIFIC DISCOVERY WITH SCIML: NEURAL OPERATORS, GENERATIVE MODELING, AND PHYSICS-INFORMED TECHNIQUES

VIBRATIONS ANALYSIS OF STRUCTURAL PROPERTIES WITH UNCERTAINTIES USING OSCILLATORY PHYSICS-INFORMED NEURAL NETWORKS (OPINN)…… 511
Stefan Hildebrand, Sandra Klinge

Thematic Sessions

TS 16: SYSTEM RELIABILITY ANALYSIS, DESIGN AND RISK ASSESSMENT

TIERED MULTI-SCALE DAMAGE ASSESSMENT OF CRITICAL INFRASTRUCTURE USING COHERENCE CHANGE DETECTION AND SAM-BASED DEEP LEARNING..... 523
Nadiia Kopiika, Sotirios Argyroudis, Jelena Ninic, Zehao Ye, Stergios-Aristoteles Mitoulis

TS 17: UNCERTAINTY QUANTIFICATION

SENSITIVITY ANALYSIS OF IMAGE CLASSIFICATION MODELS USING GENERALIZED POLYNOMIAL CHAOS 537
Lukas Bahr, Lucas Poßner, Konstantin Weise, Stephan Matzka, Sophie Gröger, Rüdiger Daub

INPUT PARAMETER SPECIFIC UNCERTAINTY QUANTIFICATION OF DEEP LEARNING MODELS USING MONTE CARLO DROPOUT..... 549
Eugen Boos, Maik Linnemann, Hajo Wiemer, Verena Psyk, Steffen Ihlenfeldt

A MULTI-FIDELITY REDUCED-ORDER MODEL TO QUANTIFY AERODYNAMIC FORCES ON A VERTICAL-AXIS WIND TURBINE WITH UNCERTAIN ROTATIONAL SPEED..... 569
Fausto Dicech, Lucia Parussini

A HIERARCHICAL BAYESIAN APPROACH FOR POPULATION-BASED SHM IN SHIP HULL STRUCTURES..... 584
Georgios Aravanis, Nicholas Silionis, Jacopo Bardiani, Marco Giglio, Konstantinos Anyfantis, Claudio Sbarufatti

DOMAIN ADAPTATION IN STRUCTURAL HEALTH MONITORING USING LEARNING BOUNDS AND GENERATIVE NETWORKS..... 597
C.A. Lindley, N. Dervilis, K. Worden

UNCERTAINTY QUANTIFICATION OF WIRE TOLERANCES IN COIL ASSEMBLY..... 605
Romas Zubavicius, Maria Nogal, Gan Fu, Mitrofan Curti, Oswaldo Morales-Nápoles

TRAINING AND DEPLOYMENT OF A DEEP ABSTAINING CLASSIFIER UNDER FEDERATED LEARNING..... 621
Cristina Garcia Cardona, Jamaludin Mohd Yusof

ADDRESSING EPISTEMIC UNCERTAINTY IN STATISTICAL TOLERANCE ANALYSIS: A NON-DETERMINISTIC APPROACH..... 639
Sonia Carolina Garcia Gomez, Pierre Beaurepaire, Nicolas Gayton, Chahrazed Amrane

NAVIGATING UNCERTAINTY IN IMAGE-BASED STRUCTURAL HEALTH MONITORING: THE ROLE OF FALSE POSITIVES, FALSE NEGATIVES, AND BASE RATE BIAS..... 656
Vagelis Plevris, Alejandro Jiménez Rios

IDENTIFICATION OF THE LATERAL TRACK IRREGULARITIES FROM THE BEHAVIOR OF THE TRAIN'S AXLES ON A SURROGATE RAILWAY DYNAMICS MODEL..... 669
Malek Chihaoui, Denis Duhamel, Christine Funfschilling, Guillaume Perrin

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

HETEROSCEDASTIC GAUSSIAN PROCESS THROUGH KERNEL SMOOTHING

Ghifari A. Faza[1,3], **Nasrulloh R.B.S. Loka**[2], **and David Moens**[1,3]

[1] LMSD, Department of Mechanical Engineering, KU Leuven, Belgium e-mail: {adam.faza,david.moens}@kuleuven.be

[2] Department of Computer Science, Helsinki University
e-mail: nasrulloh.satrio@helsinki.fi

[3] Flanders Make @KU Leuven, Belgium

Abstract. *Gaussian process (GP) regression is widely used in practice due to its ability to handle noisy problems. In many scientific experiments or stochastic simulators, observation noise can vary across the input space (i.e. heteroscedastic). This paper proposes a new approach to handle heteroscedastic noise that is inherently present in the data. Here, we follow a post-modelling learning strategy similar to the most likely heteroscedastic Gaussian process (MLHGP) algorithm. Unlike the MLHGP and its variations, which require multiple GP models, our proposed methodology only requires one GP model to fit the main function and uses the trained kernel parameters to estimate the noise level via kernel smoothing regression. We test our proposed model on three benchmark experiments: a one-dimensional analytical case, a four-dimensional computational simulation case, and a one-dimensional simple Bayesian optimisation problem to analyse how heteroscedastic modelling performs in a risk-averse setting. Our proposed method is able to reduce the computational complexity from $\mathcal{O}(2\mathcal{N}^3)$ to $\mathcal{O}(\mathcal{N}^3 + \mathcal{N}^2)$. As we don't need to train the kernel smoothing, our empirical results show that the proposed method is able to achieve speedup up to almost 2 times during training while not compromising the predictive capability. Additionally, in our Bayesian optimisation test case, the result shows that the proposed method outperforms MLHGP in case of a low number of initial observations and remains competitive in medium and high initial observation settings, all while being faster in every case.*

Keywords: Gaussian process, Heteroscedastic noise, Kernel smoothing.

1 INTRODUCTION

Data-driven methods have gained traction as a viable alternative to numerical simulation for solving problems. Especially in problems that demand extensive computational resources such as design exploration, optimisation and reliability analysis, a data-driven surrogate model is often utilized to replace the original numerical simulation. These machine learning surrogate models, such as support vector regression [1], Gaussian process [2], and neural networks [3, 4], leverage data to approximate complex relationships, circumventing the need for computationally intensive simulations for every configuration.

Compared to the other methods, the Gaussian process (GP) model is very attractive since it naturally provides a predictive distribution as an uncertainty estimate besides the predictive mean. This capability of GPs to characterize uncertainty has been leveraged in various applications, such as Bayesian optimisation [5–8], sensitivity analysis [9] and reliability analysis with active learning [10, 11]. The capability underscores the importance of developing more appropriate GP models that can effectively capture uncertainty, as it directly impacts the performance of applications leveraging this property.

In the standard GP regression, the noise level that is inherently present in the data is typically presumed to be constant throughout the input space (homoscedastic). However, in many real-world problems [12–14], the observation variability can heavily depend on the input (heteroscedastic). Hence, assuming that the data is homoscedastic can lead to underestimating the variance of high-noise regions and overestimating it in low-noise regions. As a result, a homoscedastic GP model may misrepresent uncertainty and provide suboptimal predictions in heteroscedastic settings. The illustration given in Figure 1 depicts the main difference between homoscedastic and heteroscedastic problems.

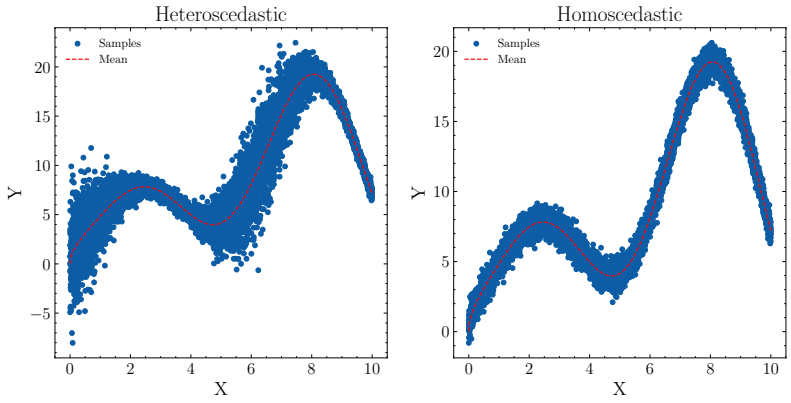


Figure 1: Varying noise in heteroscedastic function (left) and constant noise in homoscedastic function (right).

Various heteroscedastic Gaussian process (HGP) models [15–24] have been proposed to properly handle the noise level that varies across the input space instead of assuming a constant noise. The typical HGP configuration uses two GPs, with the first GP modelling the main function and the second GP learning the input-dependent noise level. Despite the apparent simplicity of the idea, the combination of the two GPs produces a joint posterior distribution over the main function and the heteroscedastic noise which is analytically intractable.

Markov chain Monte Carlo (MCMC) sampling is often viewed as the gold standard to obtain the joint posterior distribution [15]. However, due to its expensive cost, especially in large datasets, various alternatives have been developed to achieve a trade-off between accuracy and

computational efficiency such as the expectation propagation (EP) [16, 17], Laplace approximation [18], variational inferences [19–21], and most likely noise approaches [22–24]. Among these alternatives, the most likely noise approaches, including the most likely heteroscedastic GP (MLHGP) [22], improved MLHGP (IMLHGP) [23], and the nearest neighbour point estimate HGP (NNPEHGP) [24] are the simplest and most computationally attractive, in which the noise posterior is replaced by a point estimate at its most likely level. Thus, the predictive posterior distribution can be obtained analytically.

In this research, we propose a modification to the most likely noise approaches by using a kernel smoothing (KS) function [25–27] in place of the second GP. Our proposed method, named KS-MLHGP and KS-IMLHGP, is designed to enhance the adaptability of both MLHGP and IMLHGP. The primary goal of our proposed method is to address the issue of noise level overfitting, particularly prevalent in IMLHGP when the training data is limited, while also reducing the computational cost of MLHGP. To validate the performances of the proposed methods, we compare their predictive capabilities in terms of the predicted distribution on several heteroscedastic problems consisting of one 1-dimensional analytical case, one 4-dimensional computational simulation, and one simple Bayesian optimisation case.

This paper is organized as follows: In section 2 we review the related work on heteroscedastic GP. Then we discuss our proposed algorithm using Kernel smoothing regression to approximate the noise level in section 3. Section 4 discusses the performance of our proposed algorithm compared to the current ones and finally, the conclusion and future works are given in Section 5.

2 PRELIMINARIES AND RELATED WORKS

The goal of using a Gaussian process as a surrogate model is to recover an unknown black-box function $f(\boldsymbol{x})$ from a dataset $D = \{\boldsymbol{x}_i, y_i\}_{i=1}^n$, where $\boldsymbol{x} \in \mathbb{R}^d$ is the input vector of dimension d. The observed output of the unknown function is a scalar denoted by $y_i \in \mathbb{R}$ such that

$$y_i = f(\boldsymbol{x}_i) + \varepsilon_i, \tag{1}$$

where the observation error ε_i is typically assumed to be:

$$\varepsilon_i \sim \mathcal{N}(0, \sigma_o^2). \tag{2}$$

The observation noise σ_o^2 can be constant (homoscedastic) or varying in the input space governed by another unknown function $\sigma_o = g(\boldsymbol{x}_i)$ (heteroscedastic).

We can place a GP prior on the unknown black-box function $f(\boldsymbol{x})$ in equation 1, so it follows a multivariate normal distribution fully specified by the mean $\boldsymbol{\mu}$ and the covariance matrix $\boldsymbol{K}$, written as:

$$p(f|\boldsymbol{X}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{K}), \tag{3}$$

where $\boldsymbol{X} = \{\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_n}\}$ is a collection of data point $\boldsymbol{x_i}$. A zero-mean GP prior is often assumed over the function value for simplification, hence $\boldsymbol{\mu} = \boldsymbol{0}$. The entries of the covariance matrix $\boldsymbol{K}$ are calculated from the covariance function $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ at input points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. In the present study, we mainly focus on using the squared exponential (SE) kernel [2] as the covariance function, since we assume that our target function is smooth. The SE kernel is written as:

$$k_{SE}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sigma_f^2 \exp\left(-\frac{||\boldsymbol{x}_i - \boldsymbol{x}_j||^2}{2l^2}\right). \tag{4}$$

The l variable usually refers to the kernel length-scale parameter, which roughly translates to how far information of a data point could affect others, σ_f^2 is the signal amplitude or the scaling

parameter, and $||\cdot||$ denotes the Euclidean distance between input locations $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. This means the SE kernel is driven by two parameters $\theta_f = \{\sigma_f^2, l\}$ that measures the similarity between two observations.

The assumption of homoscedasticity and a zero-mean GP prior leads to the formation of the homoscedastic GP hyperparameters $\boldsymbol{\theta}_y = \{\boldsymbol{\theta}_f, \sigma_o^2\}$, where the model hyperparameters can be estimated from the data by maximizing the log marginal likelihood, written as:

$$\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}_y) = -\frac{1}{2}(\boldsymbol{y}-\boldsymbol{\mu})^T(\boldsymbol{K}+\sigma_o^2\boldsymbol{I})^{-1}(\boldsymbol{y}-\boldsymbol{\mu}) - \frac{1}{2}\log|\boldsymbol{K}+\sigma_o^2\boldsymbol{I}| - \frac{n}{2}\log(2\pi). \quad (5)$$

This can be obtained by using any optimisation algorithm with the goal of reaching an acceptable local maximum or ideally, global optimum.

In the heteroscedastic Gaussian process (HGP), the main idea is to employ another GP model to approximate the observed noise level $g(\boldsymbol{x}_i) = \sigma_o$, with a separate covariance function $k_z(\boldsymbol{x}_i, \boldsymbol{x}_j)$ parametrised by $\boldsymbol{\theta}_z$. Therefore, two GPs are involved in HGP modelling, where the first GP learns the unknown main function (the y-process) and the second GP learns the unknown function of varying noise (the z-process). Now, the predictive posterior distribution over the test output y_* is given by:

$$p(y_*|\boldsymbol{x}_*, \boldsymbol{\theta}_f, \boldsymbol{\theta}_z, D) = \iint p(y_*|\boldsymbol{x}_*, \boldsymbol{\theta}_f, \boldsymbol{z}, z_*, D)p(\boldsymbol{z}, z_*|\boldsymbol{x}_*, \boldsymbol{\theta}_z, D)d\boldsymbol{z}dz_*, \quad (6)$$

where $\boldsymbol{z}$ is the noise level at input training and z_* is the noise level at $\boldsymbol{x}_*$. However the full posterior of the noise level $p(\boldsymbol{z}, z_*|\boldsymbol{x}_*, \boldsymbol{\theta}_z, D)$ is not solvable analytically and therefore one has to provide an approximation to solve it. Since the integral is intractable, various approximation methods have been proposed such as MCMC [15], variational inference [19–21], expectation propagation [16, 17], and Laplace approximation [18].

Other approaches that are also popular are the Most likely heteroscedastic Gaussian process (MLHGP) [22] and its variations, including the improved MLHGP (IMLHGP) [23] and the nearest neighbour point estimate HGP (NNPEHGP) [24]. These methods propose a simple and computationally cheap approach to approximate HGP. MLHGP replaces the full posterior of the varying noise with a point estimate at the most likely value such that the predictive posterior can be treated analytically. This reads as

$$p(\boldsymbol{z}, z_*|\boldsymbol{x}_*, \boldsymbol{\theta}_z, D) \approx \delta(\tilde{\boldsymbol{z}}, \tilde{z}_*), \quad (7)$$

where $(\tilde{\boldsymbol{z}}, \tilde{z}_*)$ is the most likely (logarithmic) noise level, and δ is the Dirac delta function with $\delta(\tilde{\boldsymbol{z}}, \tilde{z}_*) = 1$ when $\tilde{\boldsymbol{z}} = \tilde{z}_*$ and zero otherwise. Thus, the predictive posterior distribution over the output y_* is approximated as:

$$\begin{aligned} p(y_*|\boldsymbol{x}_*, \boldsymbol{\theta}_f, \boldsymbol{\theta}_z, D) &\approx \iint p(y_*|\boldsymbol{x}_*, \boldsymbol{\theta}_f, \boldsymbol{z}, z_*, D)\delta(\tilde{\boldsymbol{z}}, \tilde{z}_*)d\boldsymbol{z}dz_* \\ &\approx p(y_*|\boldsymbol{x}_*, \boldsymbol{\theta}_f, \tilde{\boldsymbol{z}}, \tilde{z}_*, D), \end{aligned} \quad (8)$$

where the most likely noise level is given by:

$$(\tilde{\boldsymbol{z}}, \tilde{z}_*) = \arg\max_{(\tilde{\boldsymbol{z}}, \tilde{z}_*)} p(\boldsymbol{z}, z_*|\boldsymbol{x}_*, \boldsymbol{\theta}_z, D). \quad (9)$$

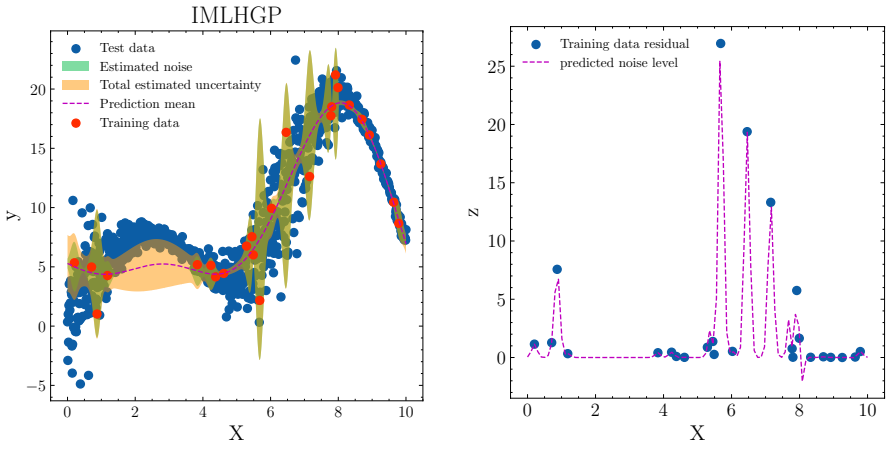
As the input-dependent noise is also modelled by a GP regression, its most likely noise level is expressed as $(\tilde{\boldsymbol{z}}, \tilde{z}_*) = (\boldsymbol{\mu}_{\boldsymbol{z}}, \mu_{z_*})$. Thus, the posterior distribution over the output can be further expressed as:

$$p(y_*|\boldsymbol{x}_*, \boldsymbol{\theta}_f, \boldsymbol{\theta}_z, D) \approx p(y_*|\boldsymbol{x}_*, \boldsymbol{\theta}_f, \boldsymbol{\mu}_{\boldsymbol{z}}, \mu_{z_*}, D) \quad (10)$$

The main difference between MLHGP, IMLHGP, and NNPEHGP is that in MLHGP [22], an iterative approach similar to the expectation-maximization (EM) algorithm is used to obtain the noise estimation. IMLHGP [23] uses a correction factor to approximate the final result, making the algorithm faster. Meanwhile the NNPEHGP [24] was developed to address the overfitting issue in IMLHGP by using k-nearest neighbour regression to transform the scattered Bayesian residual into a smoother function.

3 HETEROSCEDASTIC GAUSSIAN PROCESS WITH KERNEL SMOOTHING

The training procedure of the MLHGP involves an iterative process similar to the EM algorithm, making the computational complexity of MLHGP becomes prohibitively high. To address this problem, the IMLHGP [23] model was then developed. In some cases, IMLHGP even provides a better prediction of the noise and the full distribution compared to MLHGP under the condition of a large number of training data sizes. However, when only a few training data are available, we observe that IMLHGP tends to overfit the noise level as shown in Figure 2a. The overfitting phenomenon is reasonable when only a few observations are available. When the number of training samples is low, the noise residuals are likely scattered and it becomes difficult for a GP to model the noise level properly, as shown in Figure 2b. To avoid the overfitting problem, NNPEHGP [24] then proposes to utilize k-nearest neighbour regression to transform the scattered residual into a smoother function. However, this approach requires the correct definition of the number of the neighbours k to be able to accurately model the heteroscedastic noise.



(a) (b)

Figure 2: (a) Noise level overfitting behaviour in IMLHGP, (b) Second GP overfitting in estimating the residual.

In contrast to NNPEHGP, our proposed approach aims to avoid overfitting in the noise level approximation and reduce the computational complexity of training multiple GPs to approximate the noise level by adopting the parameters of the trained main function GP. Since we use the squared exponential kernel, defined in equation 4, this means we use the lengthscales l parameter from the trained GP directly without retraining for the noise approximation.

In practice, we relax the procedure of the typical heteroscedastic GP that uses two GPs by replacing the noise-GP with a kernel smoothing function [25–27] with a length scale identical to the first Gaussian process (GP). The main motivation of this study comes from our observation

that the GP model assumes neighbouring data points are similar and correlated following the kernel function 4. As the second GP training is prone to overfitting in case of a low number of observations, we hypothesise that the length scale parameter from the first GP is already a good approximation to describe how neighbouring data noise correlates to each other.

Given $\boldsymbol{l_1}$ as the length scale parameter from the trained first GP, the Nadaraya-Watson estimator (kernel smoothing) function [27] is defined as:

$$\hat{y}(\boldsymbol{x}_*; \boldsymbol{l_1}) = \sum_{i=1}^{n} W_i(\boldsymbol{x}_*) Y_i, \tag{11}$$

where $W_i(x)$ is the weighting function derived from the kernel function:

$$W_i(\boldsymbol{x}_*) := \frac{k_{l_1}(\boldsymbol{x}_*, \boldsymbol{x}_i)}{\sum_{j=1}^{n} k_{l_1}(\boldsymbol{x}_*, \boldsymbol{x}_j)}. \tag{12}$$

The subscript in k_{l_1} indicates that the squared exponential kernel in equation 4 is using $\boldsymbol{l_1}$ as its length scale parameter. The σ_f parameters in the kernel will be eventually cancelling each other in equation 12. Thus, we don't need to provide a specific value. Since our proposed method is very flexible, it can be used in the fashion of either MLHGP or IMLHGP. Thus, as a comparison we introduce the kernel smoothing variant of both cases named KSMLHGP and KSIMLHGP respectively.

4 EXPERIMENTAL RESULTS

To quantitatively assess the model predictive performance and uncertainty quantification (UQ) quality, we follow [22–24] to use several metrics. First, to assess the noise level prediction, we use the standardized mean square error (SMSE) between the true noise standard deviation and the noise level prediction by the GP model, written as

$$SMSE_{(g)} = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} \frac{(\tilde{g}_i - g_{*,i})^2}{\text{var}(g_*)}, \tag{13}$$

where n_{test} is the number of test points, $\tilde{g}_i$ is the predicted noise standard deviation, $g_{*,i}$ is the true noise standard deviation, and $\text{var}(g_*)$ is the variance of the true noise standard deviation at all test points.

The goal of heteroscedastic modelling is to predict the full distribution of a specific point. Hence, to assess the GP model in predicting the full distributions, we follow [24, 28] using the expectation of the normalized 2-Wasserstein distance at all test points $x_{*,i}$, defined as

$$\varepsilon = \mathbb{E}[d(Y(x_*), \hat{Y}(x_*)], \tag{14}$$

where d is the normalized 2-Wasserstein distance between the probability distribution over the true response $Y(x_*)$ and the prediction from GP model $\hat{Y}(x_*)$

$$d(Y_1, Y_2) = \frac{d_{WS}(Y_1, Y_2)}{\sigma(Y_1)}, \tag{15}$$

and the Wasserstein distance d_{WS} is written as

$$d_{WS}(Y_1, Y_2) := ||Q_1 - Q_2||_2 = \sqrt{\int_0^1 (Q_1(u) - Q_2(u))^2 du}. \tag{16}$$

Q_1 and Q_2 are the quantile function of Y_1 and Y_2 respectively.

Additionally, we also employ the average of the negative log probability density (NLPD) to quantify the predictive quality of the test outputs, since in practice, the true standard deviation in noise is sometimes unavailable. Such metric is evaluated by:

$$NLPD_{(y)} = -\frac{1}{N}\sum_{i=1}^{N} \log p(y_{*,i}|x_{*,i}, D). \tag{17}$$

4.1 One-dimensional analytical problem

We consider a one-dimensional mathematical function with heteroscedastic noise, defined as:

$$\begin{aligned} g(x) &= x\sin(x) + 4\sqrt{x}, \\ h(x) &= \exp(\cos(x)), \\ f(x) &\sim \mathcal{N}(g(x), (h(x))^2), \end{aligned} \tag{18}$$

with input variable x within $[0, 10]$. This one-dimensional function has a nonlinear mean function $g(x)$ and nonlinear noise level $h(x)$ that varies in the input domain x. We use the training dataset $n = 25$ for this problem, and the number of test points is set to 1000. One realization of different models is evaluated as shown in Figure 3.

Here, we observe that IMLHGP is overfitting on the noise level prediction indicated by the non-smooth noise level prediction. Even NNPEHGP which is supposed to solve the overfitting problem from IMLHGP still looks overfitted at $X = [4, 10]$. We can also observe that using kernel smoothing approximation alleviates the overfitting problem in KSIMLHGP. Meanwhile, comparing MLHGP and KSMLHGP we observe that the prediction distribution looks very similar.

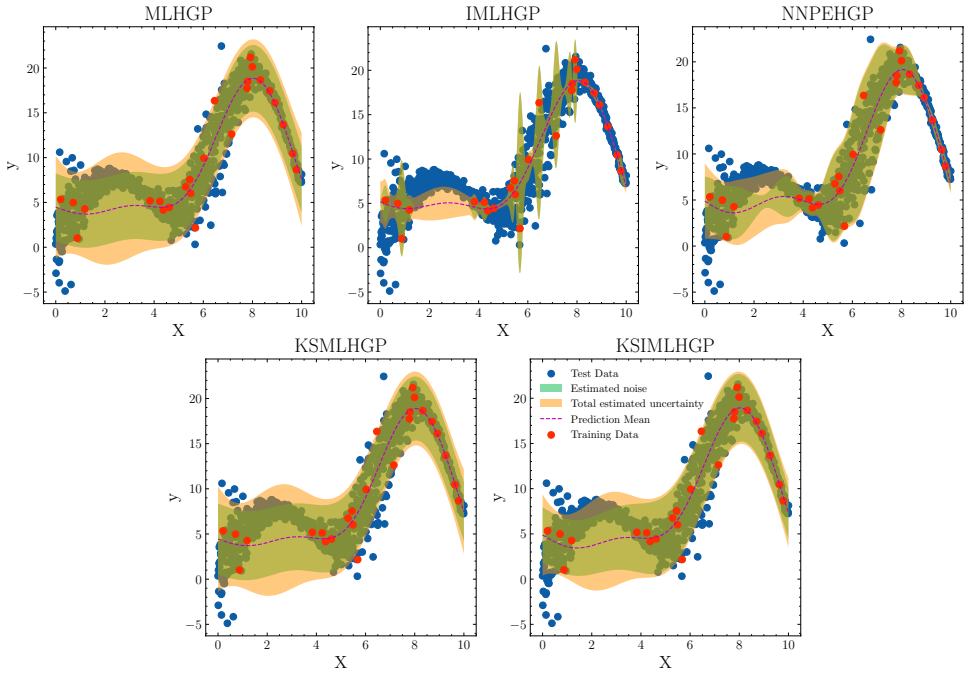
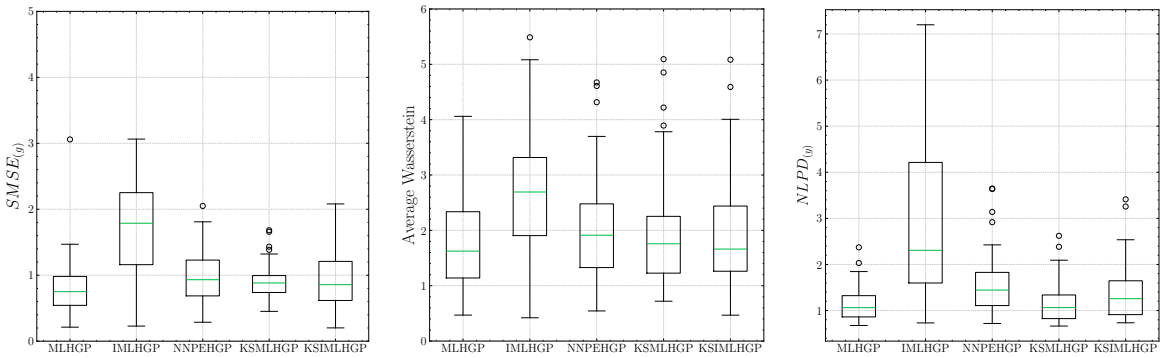


Figure 3: One realization of multiple model runs for the one-dimensional analytical case.

The performances of the models are evaluated on standardized mean square error (SMSE), averaged 2-Wasserstein distance (ε), and negative log probability density (NLPD) which are

shown in Figure 4a, 4b, and 4c respectively. In these evaluations, we only consider a small number of training datasets ($n = 25$) and repeat each model's training and evaluations with 50 different training datasets with different random seeds. The results show that in this problem IMLHGP has the worst performance compared to the other and employing the kernel smoothing for noise level prediction provides improvement in its predictive capability. Meanwhile, KSMLHGP holds a similar performance to the MLHGP while reducing the computational complexity.



(a) (b) (c)

Figure 4: (a) Standardized mean squared error, (b) Averaged 2-Wasserstein distance, and (c) Negative log probability density for one-dimensional analytical problem.

4.2 Four-dimensional computational simulation

To represent a more realistic engineering problem, we follow [24] to evaluate the 4-dimensional computational simulation of a solid sphere immersed in fluid [13]. This heat transfer problem models the temperature of the sphere in a fluid with a higher temperature as the initial condition, where the heat is transferred between the sphere and the fluid via convection and via conduction inside the sphere. The temperature inside the sphere T is described as the solution of a partial differential equation. The solution is explicitly given by

$$T = u_2 + u_3 \sum_{i=1}^{\infty} \frac{4(\sin \eta_i - \eta_i \cos \eta_i)}{2\eta_i - \sin(2\eta_i)} \exp(-\eta_i^2 \pi_2) \frac{\sin(\eta_i u_1)}{\eta_i u_1}, \tag{19}$$

where η_i is the solution of:

$$\begin{aligned} 1 - \eta_i \cot \eta_i = \pi_1, \quad \eta_i \in ((i-1)\pi, i\pi) \\ \pi_1 = \frac{u_4 u_7}{u_6}, \quad \pi_2 = \frac{u_6 u_5}{u_8 u_9 u_7^2}. \end{aligned} \tag{20}$$

We define the input variable as the distance from the sphere's centre u_1, the fluid initial temperature u_2, the initial temperature of the sphere u_3, and the convective heat transfer coefficient u_4. Each variable is specified in Table 1. The time variable u_5 is set to 800 s, and the other variables $u_6, \ldots, u_9$ are treated as random variables that are distributed normally, given in Table 2.

Although the PDE solution in equation 19 is explicit, the exact noise function is unknown. Typically in this situation, we evaluate the negative log probability density (NLPD). However, since the problem is computationally cheap, we estimate the output distribution using multiple realizations per test input. This allows us to use SMSE and 2-Wasserstein distance as metrics.

Table 1: Input variable ranges for the heat transfer problem.

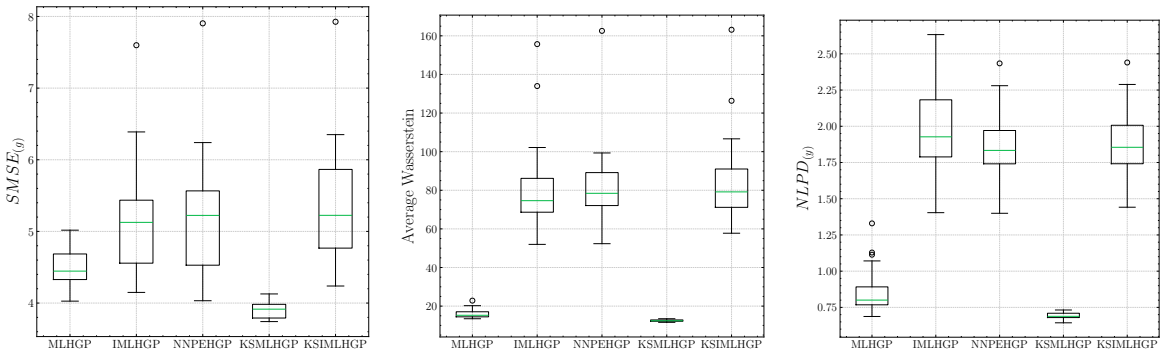
Variable	Description	Units	Minimum	Maximum
u_1	Distance from the center of sphere	-	0	1
u_2	Temperature of fluid	K	250	270
u_3	Initial sphere temperature	K	-100	-30
u_4	Convective heat transfer coefficient	$kgs^{-3}K^{-1}$	180	210

Table 2: Fixed and random variable parameters for the heat transfer problem.

Variable	Description	Units	Value/Distribution
u_5	Time	s	800
u_6	Thermal conductivity	$kgs^{-3}K^{-1}$	$\mathcal{N}(65, 10.3^2)$
u_7	Sphere's radius	m	$\mathcal{N}(0.1, 0.02^2)$
u_8	Specific heat	$m^2s^{-2}K^{-1}$	$\mathcal{N}(400, 33.3^2)$
u_9	Density	kgm^{-3}	$\mathcal{N}(8000, 333.3^2)$

Here, we use 10^3 test data points in the input spaces with 10^4 realizations at each point. The number of training samples is set to 160. In this problem, we repeat each experiment with a different random seed 25 times.

Figure 5 clearly shows that MLHGP-based methods perform better than IMLHGP-based methods on a fairly complex problem with noticeable differences. Here, we especially highlight the KSMLHGP method that outperforms the IMLHGP-based methods and is still relatively comparable to the original MLHGP while reducing the computational complexity from $\mathcal{O}(2\mathcal{N}^3)$ to $\mathcal{O}(\mathcal{N}^3 + \mathcal{N}^2)$.



(a) (b) (c)

Figure 5: (a) Standardized mean squared error, (b) Averaged 2-Wasserstein distance, and (c) Negative log probability density for the four-dimensional heat conduction simulation.

4.3 Risk-averse Bayesian optimisation on one-dimensional function

Bayesian optimisation (BO) [7] is a common application of GPs that aims to identify the global optimum (in this work, we consider minimization) of a black-box function, represented as $x^* \in \min_{x \in \mathcal{X}} f(x)$, where $\mathcal{X} \in \mathbb{R}^d$. Such functions are often expensive to evaluate, noisy, or both. The common approach in BO is an iterative process where a surrogate model, typically a GP, is first trained to approximate the target function. Subsequently, an acquisition function (AF)

is optimized to determine the next query point for efficiently locating the function's minimum.

This section focuses on a scenario where the function has heteroscedastic noise. Rather than seeking a simple global minimum like in a non-noisy BO setting, the goal is to identify a risk-averse optimum. To tackle the risk aversion problem, we follow the framework proposed by [29] and adopt the widely used mean-variance (MV) objective [30]. This objective defines a trade-off between the mean return $f(x)$ and the risk represented by a variance proxy $\rho^2(x)$:

$$\mathrm{MV}(x) = f(x) + \alpha\rho^2(x), \tag{21}$$

where $\alpha > 0$ is the absolute risk tolerance parameter, assumed to be known to the learner. In this work, we focus on a minimization task, aiming to achieve a solution where $\mathrm{MV}(x)$ is as low as possible (i.e., a solution with a low function value and minimal uncertainty).

To evaluate our model in BO , we apply the Risk-Averse Heterogeneous Bayesian optimisation (RAHBO) AF [29, 31]. RAHBO extends the GP Upper Confidence Bound (GP-UCB) [32] AF by incorporating input dependent noise prediction. It is defined as:

$$\mathrm{UCB}(x) = \mu(x) + \beta\sigma(x), \qquad \mathrm{RAHBO}(x) = \mathrm{UCB}(x) + \alpha g^2(x). \tag{22}$$

We evaluate our approach using four metrics: the *risk-averse regret*, defined as $\mathrm{Rt} = \mathrm{MV}(x^{**}) - \mathrm{MV}(x_t)$, where $x^{**} \in \arg\max_{x\in\mathcal{X}} \mathrm{MV}(x)$; the *cumulative risk-averse regret*, given by $\sum_{t=1}^{T} (\mathrm{MV}(x^{**}) - \mathrm{MV}(x_t))$; the *simple regret*, expressed as $f(x^*) - f(x_t)$; and the *cumulative simple regret*, computed as $\sum_{t=1}^{T} (f(x^*) - f(x_t))$. The cumulative metrics are useful for online settings, while the simple regret is used to compare the method against non-risk-averse baseline.

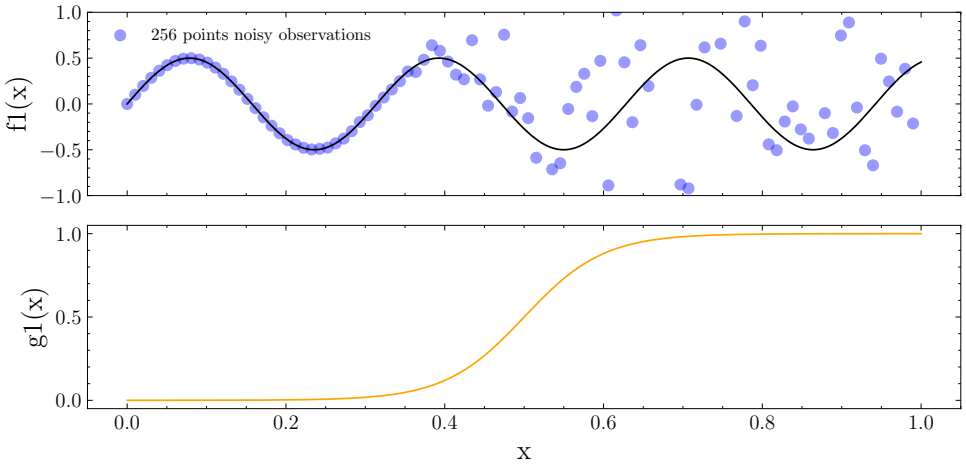


Figure 6: Synthetic heteroscedastic Bayesian optimisation function with 256 noisy evaluation points.

We consider a one-dimensional noisy optimisation problem, illustrated in Figure 6. The problem features three global (mean) minima, each subject to varying levels of noise. The function is defined as follows:

$$f(x) = \mathrm{f1}(x) + \epsilon_1, \qquad \epsilon_1 \sim \mathcal{N}(0, \mathrm{g1}(x)), \tag{23}$$

where $\mathrm{f1}$ and $\mathrm{g1}$ are defined as:

$$\mathrm{f1}(x) = 0.5\sin(20x), \qquad \mathrm{g1}(x) = \frac{1}{1 + e^{-(20x-10)}}. \tag{24}$$

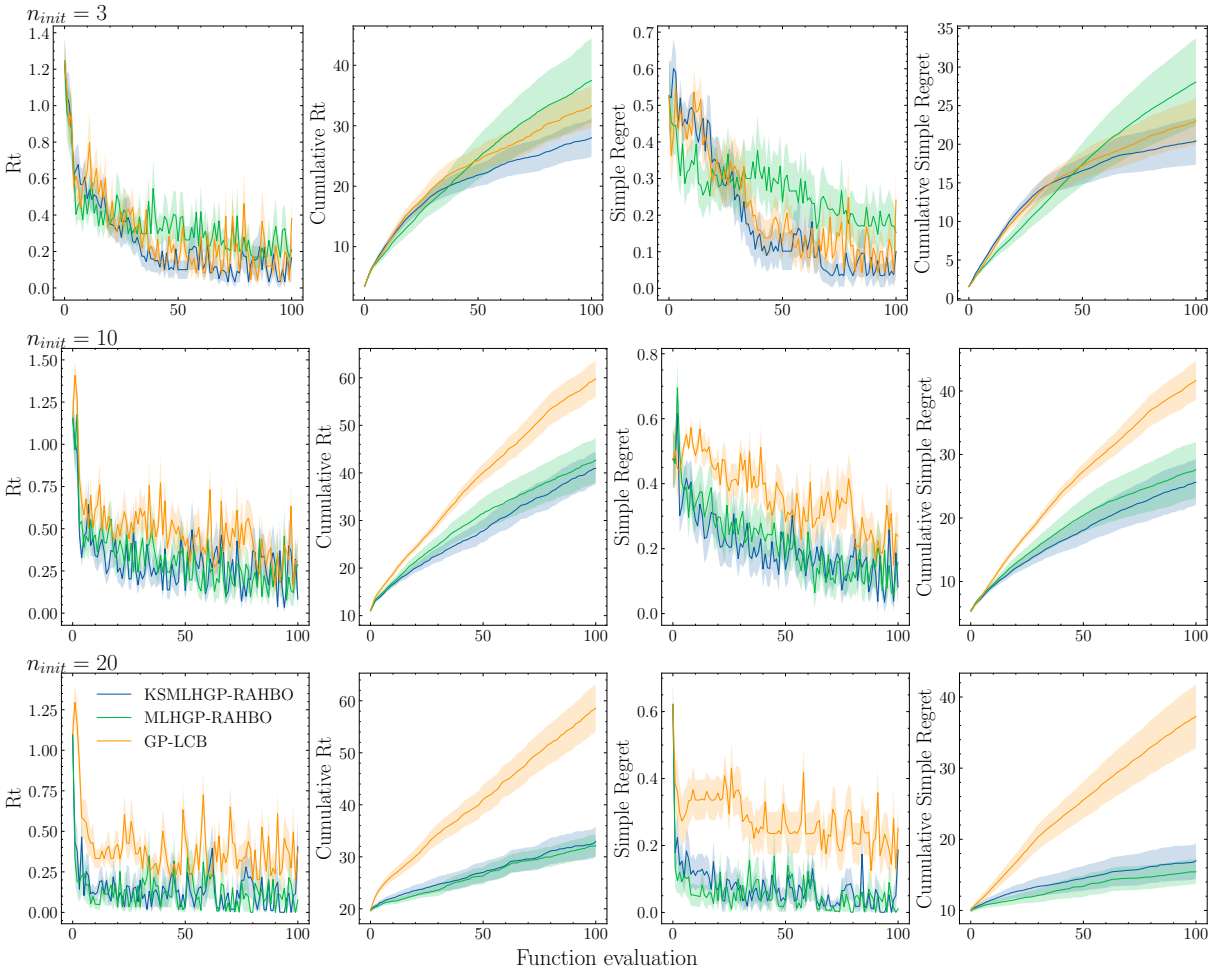


Figure 7: Risk-averse heteroscedastic Bayesian optimisation result, from top to bottom, initial sample $\mathrm{n_{init}} = 3, 10, 20$.

We conduct experiments to test the performance of the KSMLHGP with the RAHBO AF on the above 1D function problem. We compare it with two baselines: first, the MLHGP with the RAHBO AF, and second, a non-risk-averse baseline—a GP model with the lower confidence bound (LCB) AF. Notably, we use LCB instead of the UCB [32] in this work, as our problem involves minimisation instead of maximisation. We assume a fixed, known $\alpha = 1$ for both $\mathrm{MV}(x)$ in the evaluation metrics calculation and the RAHBO AF. The β parameters for RAHBO and LCB are set to 0.2, following the default setting of the popular BO package BoTorch [33][1]. We optimize the AF using random search with 10K candidates[2]. We ran the experiments with 100 BO iterations starting from initial observations and repeated 10 times with different initial observation configurations for consistency. We use different numbers of initial observations—3, 10, and 20—to examine how the models behave under varying numbers of initial observations.

The results of the experiments are presented in Figure 7. We observe that KSMLHGP-RAHBO outperforms MLHGP-RAHBO in the low initial observation setting ($\mathrm{n_{init}} = 3$) while remaining competitive in the mid and high initial observation settings ($\mathrm{n_{init}} = 10, 20$, respectively). This demonstrates that KSMLHGP is more robust when trained on a small number of

[1]https://botorch.org/

[2]For higher-dimensional problems, a more sophisticated optimizer, such as a multi-start gradient-based or evolutionary algorithm, is preferable due to the curse of dimensionality.

initial observations compared to MLHGP. Moreover, in the cumulative regret metrics, risk-averse methods exhibit significant performance gains compared to non-risk-averse ones. While this result is expected, it highlights that adopting a risk-averse approach might be highly beneficial in online settings, particularly in safety-critical applications.

Additionally, we tested the average runtime for each model (independent of n_{init}) with $n_{\text{iter}} = 100$. The results are as follows: GP-LCB: 75 seconds, KSMLHGP-RAHBO: 372 seconds, MLHGP-RAHBO: 665 seconds. All experiments were conducted on a MacBook with an M1 Pro chip. While our method is still slower compared to the non-risk-averse method, it shows a significant speed improvement over the state-of-the-art MLHGP while maintaining good performance and even outperforming it in low initial observation settings. The code that we used for the experiments is available on GitHub [3].

5 CONCLUSION

In this paper, we introduce kernel smoothing as a viable alternative to the second Gaussian process (GP) for modelling the noise level in heteroscedastic data. In this approach, we set the kernel length scale of the kernel smoothing method to be identical to the first GP model that approximates the mean function of the data by assuming that neighbouring data points should have similar noise levels. This approach presents an enhancement over the IMLHGP by enhancing its resilience to overfitting in scenarios where the observed data is limited. Furthermore, it also improves upon the MLHGP by decreasing the computational complexity, as it only necessitates training the first GP, while still maintaining comparable predictive performance to the standard MLHGP method.

We conduct a comparative analysis of our proposed kernel smoothing methods, KSMLHGP and KSIMLHGP, against established post-modeling HGP methods, including MLHGP, IMLHGP, and NNPEHGP, to validate the feasibility and effectiveness of our approaches. In our experiments, we utilize three metrics - the noise level SMSE, averaged 2-Wasserstein distance, and NLPD - to quantitatively evaluate the predictive distribution of the heteroscedastic data. The results of the experiments demonstrate that our proposed methods outperform IMLHGP and NNPEHGP in scenarios with limited data and maintain strong predictive performance compared to MLHGP. Moreover, in the simple heteroscedastic Bayesian optimisation test case we show that our proposed model outperforms MLHGP in small initial sample settings and remains competitive in larger initial samples while reducing the computational time.

While the heteroscedastic GP methods in general are quite powerful for modelling regression problems with input-dependent noise, the predictive interval from GP relies on the Gaussian assumption and the well-specification of the priors that are not always appropriate. Furthermore, in the absence of prior information, it might be difficult to fully specify the exact parameter of the GP model. Therefore it would be desirable in the future to conduct research related to conformalized-GP [34, 35] as well as imprecision in GP [36].

ACKNOWLEDGEMENTS

This work was supported by granted H2020 FETOPEN-2018-2019-2020-01 European project, *Epistemic AI* under grant agreement No. 964505 (E-pi).

[3] https://github.com/fazaghifari/hetgp-eval

REFERENCES

[1] Harris Drucker, Christopher J. C. Burges, Linda Kaufman, Alex Smola, and Vladimir Vapnik. Support vector regression machines. In M.C. Mozer, M. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9. MIT Press, 1996.

[2] Carl Edward Rasmussen and Christopher K I Williams. *Gaussian processes for machine learning*. Adaptive Computation and Machine Learning series. MIT Press, London, England, November 2005.

[3] Rohit K Tripathy and Ilias Bilionis. Deep uq: Learning deep neural network surrogate models for high dimensional uncertainty quantification. *Journal of computational physics*, 375:565–588, 2018.

[4] Gang Sun and Shuyue Wang. A review of the artificial neural network surrogate modeling in aerodynamic design. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 233(16):5863–5872, 2019.

[5] Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13:455–492, 1998.

[6] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25, 2012.

[7] Roman Garnett. *Bayesian optimization*. Cambridge University Press, 2023.

[8] Timothy M.S. Jim, Ghifari A. Faza, Pramudita S. Palar, and Koji Shimoyama. A multiobjective surrogate-assisted optimisation and exploration of low-boom supersonic transport planforms. *Aerospace Science and Technology*, 128:107747, September 2022.

[9] Lavi Rizki Zuhal, Ghifari Adam Faza, Pramudita Satria Palar, and Rhea Patricia Liem. Performance assessment of kriging with partial least squares for high-dimensional uncertainty and sensitivity analysis. *Structural and Multidisciplinary Optimization*, 66(5), April 2023.

[10] Benjamin Echard, Nicolas Gayton, and Maurice Lemaire. Ak-mcs: an active learning reliability method combining kriging and monte carlo simulation. *Structural safety*, 33(2):145–154, 2011.

[11] Lavi Rizki Zuhal, Ghifari A. Faza, Pramudita S. Palar, and Rhea P. Liem. Fast and adaptive reliability analysis via kriging and partial least squares. In *AIAA Scitech 2021 Forum*. American Institute of Aeronautics and Astronautics, January 2021.

[12] B. W. Silverman. Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 47(1):1–21, September 1985.

[13] Matthias H. Y. Tan. Monotonic quantile regression with bernstein polynomials for stochastic simulation. *Technometrics*, 58(2):180–190, April 2016.

[14] Haichen Shi, Keith Worden, and Elizabeth J. Cross. A cointegration approach for heteroscedastic data based on a time series decomposition: An application to structural health monitoring. *Mechanical Systems and Signal Processing*, 120:16–31, April 2019.

[15] Paul Goldberg, Christopher Williams, and Christopher Bishop. Regression with input-dependent noise: A gaussian process treatment. In M. Jordan, M. Kearns, and S. Solla, editors, *Advances in Neural Information Processing Systems*, volume 10. MIT Press, 1997.

[16] Luis Muñoz-González, Miguel Lázaro-Gredilla, and Aníbal R. Figueiras-Vidal. Heteroscedastic gaussian process regression using expectation propagation. In *2011 IEEE International Workshop on Machine Learning for Signal Processing*, pages 1–6, 2011.

[17] Ville Tolvanen, Pasi Jylänki, and Aki Vehtari. Expectation propagation for nonstationary heteroscedastic gaussian process regression. In *2014 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, 2014.

[18] Marcelo Hartmann and Jarno Vanhatalo. Laplace approximation and natural gradient for gaussian process regression with heteroscedastic student-t model. *Statistics and Computing*, 29(4):753–773, October 2018.

[19] Miguel Lazaro-Gredilla and Michalis K. Titsias. Variational heteroscedastic gaussian process regression. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 841–848, Madison, WI, USA, 2011. Omnipress.

[20] Marianne Menictas and Matt P. Wand. Variational inference for heteroscedastic semiparametric regression. *Australian and New Zealand Journal of Statistics*, 57(1):119–138, March 2015.

[21] Haitao Liu, Yew-Soon Ong, and Jianfei Cai. Large-scale heteroscedastic regression via gaussian process, 2020.

[22] Kristian Kersting, Christian Plagemann, Patrick Pfaff, and Wolfram Burgard. Most likely heteroscedastic gaussian process regression. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, page 393–400, New York, NY, USA, 2007. Association for Computing Machinery.

[23] Qiu-Hu Zhang and Yi-Qing Ni. Improved most likely heteroscedastic gaussian process regression via bayesian residual moment estimator. *IEEE Transactions on Signal Processing*, 68:3450–3460, 2020.

[24] Muhammad D. Robani, Pramudita S. Palar, and Lavi Rizki Zuhal. Heteroscedastic gaussian process regression using nearest neighbor point estimates. In *AIAA Scitech 2021 Forum*. American Institute of Aeronautics and Astronautics, January 2021.

[25] E. A. Nadaraya. On estimating regression. *Theory of Probability and Its Applications*, 9(1):141–142, January 1964.

[26] Geoffrey S. Watson. Smooth regression analysis. *Sankhyā: The Indian Journal of Statistics, Series A (1961-2002)*, 26(4):359–372, 1964.

[27] Herman J Bierens. The nadaraya–watson kernel regression function estimator. In *Topics in advanced econometrics*, chapter 10, page 212–247. Cambridge University Press, Cambridge, UK, 1994.

[28] Xujia Zhu and Bruno Sudret. Emulation of stochastic simulators using generalized lambda models. *SIAM/ASA Journal on Uncertainty Quantification*, 9(4):1345–1380, January 2021.

[29] Anastasia Makarova, Ilnura Usmanova, Ilija Bogunovic, and Andreas Krause. Risk-averse heteroscedastic bayesian optimization. *Advances in Neural Information Processing Systems*, 34:17235–17245, 2021.

[30] Amir Sani, Alessandro Lazaric, and Rémi Munos. Risk-aversion in multi-armed bandits. *Advances in neural information processing systems*, 25, 2012.

[31] Marshal Arijona Sinaga, Julien Martinelli, Vikas Garg, and Samuel Kaski. Heteroscedastic preferential bayesian optimization with informative noise distributions, 2024.

[32] Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings of the 27th International Conference on Machine Learning*, pages 1015–1022. Omnipress, 2010.

[33] Maximilian Balandat, Brian Karrer, Daniel R. Jiang, Samuel Daulton, Benjamin Letham, Andrew Gordon Wilson, and Eytan Bakshy. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems 33*, 2020.

[34] Harris Papadopoulos. Guaranteed coverage prediction intervals with gaussian process regression, 2023.

[35] Edgar Jaber, Vincent Blot, Nicolas Brunel, Vincent Chabridon, Emmanuel Remy, Bertrand Iooss, Didier Lucor, Mathilde Mougeot, and Alessandro Leite. Conformal approach to gaussian process surrogate evaluation with coverage guarantees, 2024.

[36] Francesca Mangili. A prior near-ignorance gaussian process model for nonparametric regression. *International Journal of Approximate Reasoning*, 78:153–171, November 2016.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

AN EMPIRICAL EVALUATION OF INTERVAL NEURAL NETWORKS FOR UNCERTAINTY QUANTIFICATION ON REGRESSION TASKS

Kaizheng Wang[1,3,4]* Ghifari Adam Faza[2,3,4]*
Keivan Shariatmadar[2,3,4] David Moens[2,3] Hans Hallez[1,4]

[1]DistriNet, Department of Computer Science, KU Leuven, Belgium
[2]LMSD, Department of Mechanical Engineering, KU Leuven, Belgium
[3]Flanders Make@KU Leuven, Belgium [4]M-Group, Leuven, Belgium
{kaizheng.wang, adam.faza, keivan.shariatmadar, david.moens, hans.hallez}@kuleuven.be

Abstract. *Uncertainty quantification in deep neural networks has become increasingly important, as effective uncertainty quantification can significantly enhance the robustness and reliability of learning systems, particularly in safety-critical applications. Probabilistic frameworks, such as Bayesian neural networks, are widely regarded as effective approaches. However, despite their popularity and demonstrated success in various applications, these methods have some key limitations: they require the specification of a prior distribution, which forces practitioners to make assumptions about its form. An alternative body of research has increasingly focused on interval neural networks (INNs) in regression tasks, which employ deterministic intervals to model and estimate the uncertainty of the predictions or the weights of the neural networks. Compared to traditional probabilistic methods, INNs require fewer probabilistic assumptions while offering theoretical guarantees of reliability and robustness. INNs also provide a more intuitive interpretation by representing uncertainty through the range of possible values, from minimum to maximum. Moreover, certain INN models featuring interval-formed weights can address interval-based input data, making them especially useful in specific applications. Nevertheless, current evaluations of INNs are generally restricted to synthetic or simple regression datasets. In this research, we comprehensively assess distinct INN models for continuous value prediction (i.e., regression) in real-world engineering applications. Our evaluation mainly encompasses three test cases: a standard regression, a realistic remaining useful lifetime (RUL) prediction task, and the RUL task in cases of interval-valued data.*

Keywords: uncertainty quantification, interval neural networks, evaluation on regression

*Equal contribution.

1 INTRODUCTION

In recent years, uncertainty quantification (UQ) for neural network predictions has garnered growing interest. Effective UQ in regression tasks enables engineers to make more informed and safer decisions when utilizing engineering surrogate models [1–3], particularly in high-stakes and safety-critical applications.

Concerning regression tasks, Nix et al. [4] were among the first to introduce a neural network architecture with two output nodes to quantify uncertainty, one to generate predictions and the other to estimate data noise. However, such neural networks do not account for parameter uncertainty, as they maintain point estimates for these parameters. Consequently, they fail to fully quantify the uncertainty associated with network predictions [5]. To address this limitation, Bayesian neural networks (BNNs), which model parameters as probability distributions, have emerged as a preferred approach [6–9]. Although BNNs are actively applied to different tasks [10, 11], there are still challenges, e.g., the huge computational complexity during training and inference [12]; also the user- or case-dependency to design reasonable priors [13]. An alternative to BNNs, interval neural networks (INNs) have gained increasing attention due to their ability to UQ without relying heavily on probabilistic and distributional assumptions [14,15]. Research on INNs can be categorized into two main approaches. The first type of approach, named light interval neural networks (light-INNs) in our work, involves INNs to generate deterministic interval predictions while maintaining fixed-point estimates for network weights and biases. In addition, ensemble techniques are also employed to account for parameter uncertainty [14, 16–18]. The second type of approach, called full interval neural networks (full-INNs) in this study, involves representing network parameters as intervals and predicting intervals through interval arithmetic during the propagation process [19–22]. Despite their simple probabilistic assumptions and considerable potential, INNs have yet to undergo rigorous evaluation on modern deep neural architectures in complex real-world scenarios, particularly in comparison to Bayesian techniques.

In addition to conventional regression tasks, the engineering domain has increasingly emphasized the need for robust UQ methods in regression with interval-valued data (IVD), of which features are represented as intervals and embed measurement uncertainty and variability [23–25]. For example, in scenarios where sensor measurements or experimental observations are subject to inherent noise or imprecision, UQ frameworks designed for IVD can significantly enhance the reliability of downstream analyzes by explicitly propagating uncertainty through models [26]. Among existing neural network-based UQ methodologies, full-INNs distinguish themselves due to their unique compatibility with interval arithmetic. However, current research on full-INNs for regression tasks [15, 18, 27] remains restricted to shallow architectures, primarily limited to multilayer perceptrons (MLP) with few hidden layers. Although a recent approach, termed interval batch normalization (IBN), has demonstrated the feasibility of integrating full-INNs with ResNet50 architecture [28] for classification tasks, the potential of modern, deep full-INN frameworks for real-world engineering regression remains to be investigated.

To address the aforementioned gaps in evaluating INNs for regression tasks, we systematically assessed the UQ capabilities of INNs and benchmarked them against BNNs. This evaluation spans three distinct case studies: a one-dimensional analytical function, a realistic remaining useful lifetime (RUL) prediction task, and the RUL task in cases of IVD.

The remainder of this paper is organized as follows: Section 2 introduces the preliminary neural network approach for uncertainty quantification under our evaluation. Section 3 provides a detailed description of our empirical evaluation experiments. Finally, Section 4 summarizes

our findings and outlines directions for future work.

2 NEURAL NETWORKS FOR UNCERTAINTY QUANTIFICATION

In this section, we introduce three distinct approaches for uncertainty quantification in regression tasks: Bayesian neural networks in Section 2.1, light interval neural networks that feature fix-valued weights and generate prediction intervals in Section 2.2, and full interval neural networks that apply interval-valued weights in Section 2.3.

2.1 Bayesian neural networks

Bayesian neural networks (BNNs) have emerged as a prominent framework for uncertainty quantification, where neural network parameters are modeled as probability distributions. After assigning prior distributions, BNNs leverage Bayesian inference to derive posterior distributions from the training data. To mitigate the significant computational complexity associated with BNNs, extensive research has been conducted, focusing on various approximation techniques [6, 12, 29–31]. Among these, Monte Carlo dropout (MC Dropout) [6] and deep ensembles [8] have widely served as Bayesian baselines for several benchmarks [32–35], due to their robust uncertainty quantification capabilities, adaptability to diverse datasets and network architectures, and relatively low computational overhead.

- MC Dropout [6] can be implemented by training a standard neural network (SNN) with dropout, approximating inference through stochastic forward passes during inference.
- Deep ensembles [8] can be constructed by training a set of SNNs under random initialization. Deep ensembles marginalize over multiple network models to derive a predictive distribution usually treated as a BNN inference approximation.

Given a training dataset $\mathbb{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$ for a univariate regression task, both MC Dropout and deep ensembles can be trained by minimizing the Gaussian negative log-likelihood (NLL) loss, as follows:

$$\sum\nolimits_{n=1}^{N} \frac{\log \sigma_i^2(\boldsymbol{x}_n)}{2} + \frac{\big(\mu_i(\boldsymbol{x}_n)) - y_n\big)^2}{2\sigma_i^2(\boldsymbol{x}_n)}, \tag{1}$$

where $\mu_i(\boldsymbol{x}_n)$ and $\sigma_i^2(\boldsymbol{x}_n)$ are the mean and variance of the predicted single Gaussian distribution $\mathcal{N}(\mu_i(\boldsymbol{x}_n), \sigma_i^2(\boldsymbol{x}_n))$ for the data point $(\boldsymbol{x}_n, y_n)$. $\mathcal{N}(\mu_i(\boldsymbol{x}_n), \sigma_i^2(\boldsymbol{x}_n))$ is obtained from the i-th stochastic forward pass in MC Dropout or the i-th SNN ensemble member in deep ensembles.

When presenting a new sample $\boldsymbol{x}^*$, these models can predict a Gaussian mixture distribution $\mathcal{N}(\mu_*(\boldsymbol{x}^*), \sigma_*^2(\boldsymbol{x}^*))$ from a set of M sampled single Gaussian distributions, as follows:

$$\mu_*(\boldsymbol{x}^*) = M^{-1} \sum\nolimits_{m=1}^{M} \mu_m(\boldsymbol{x}^*) \quad \sigma_*^2(\boldsymbol{x}^*) = M^{-1} \sum\nolimits_{m=1}^{M} \big(\sigma_m^2\big(\boldsymbol{x}^*\big) + \mu_m^2(\boldsymbol{x}^*)\big) - \mu_*^2(\boldsymbol{x}^*). \tag{2}$$

Here, M represents the number of inference samples in MC Dropout or the number of SNNs in deep ensembles. The variance can serve as the measure of prediction uncertainty.

2.2 Light interval neural networks

As an alternative to traditional probabilistic techniques that rely heavily on probabilistic and distributional assumptions [14, 15], interval neural networks (INNs) have also garnered increasing attention for their application in uncertainty quantification for regression tasks.

A significant body of research has focused on INNs that produce deterministic interval predictions while maintaining fixed point estimates for network weights and biases [14, 16–18], referred to as light-INNs in this work. The widely accepted principle for training light-INNs is to generate a high-quality prediction interval that remains as narrow as possible while ensuring a specified coverage proportion of the training data. Specifically, the primary objective is to predict a prediction interval (PI), namely $[\hat{y}^L\ \hat{y}^U]$, which bounds the target y and satisfies:

$$\text{PICP} := P_r(\hat{y}^L \leq y \leq \hat{y}^U) \geq P_c. \tag{3}$$

Here, P_c denotes the predefined confidence level, usually set as $P_c = 0.95$. P_r is the prediction interval coverage probability (PICP). Following this principle, a recent advancement [18] has introduced a loss function that is less dependent on hyperparameters, as follows:

$$\underbrace{\sum_{n=1}^{N} \frac{(y_n - \hat{y}_n)^2}{2 \cdot \text{MPIW}^2} + \frac{\log(\text{MPIW}^2)}{2}}_{\mathcal{L}_{\text{UE}}} + \lambda \underbrace{\max\{0, P_c - \text{PICP}\}^2}_{\mathcal{L}_{\text{PI}}}. \tag{4}$$

Here, $\mathcal{L}_{\text{UE}}$ approximates the Gaussian NLL loss (equation (1)) and consists of a residual regression term and an uncertainty regularization component [18]. $\hat{y}_n$ is the predicted midpoint of the prediction interval $[\hat{y}_n^L\ \hat{y}_n^U]$ for the n-th data pair and MPIW represents mean prediction interval width [14, 17, 18] obtained from $\text{MPIW} = N^{-1}\sum_{n=1}^{N}(\hat{y}_n^U - \hat{y}_n^L)$. $\mathcal{L}_{\text{PI}}$ spontaneously measures the quality of the constructed PIs and PICP can be obtained from $\text{PICP} = \sum_{n=1}^{N} c_n$ where $c_n = 1$ for $y_n \in [\hat{y}_n^L\ \hat{y}_n^U]$ and $c_n = 0$ otherwise. λ serves as the training hyperparameter to balance the importance of two loss components [18]. A larger λ empirically demonstrates greater values of the PICP and MPIW.

The ensemble concept is widely applied to light-INNs [14,17,18] to account for the uncertainty associated with neural network parameters and improve uncertainty quantification performance. Specifically, given a collection of M independently trained light-INNs, the ensemble PIs, $[\overline{\hat{y}}^L\ \overline{\hat{y}}^U]$ can be computed as follows [18]:

$$\overline{\hat{y}}^L = \left(M^{-1}\sum_{m=1}^{M} \hat{y}_m^L\right) - \sigma_{\hat{y}^L} \quad \overline{\hat{y}}^U = \left(M^{-1}\sum_{m=1}^{M} \hat{y}_m^U\right) + \sigma_{\hat{y}^U}, \tag{5}$$

where $[\hat{y}_m^L\ \hat{y}_m^U]$ represent the PI from m-th ensemble light-INN and

$$\sigma_{\hat{y}^L}{}^2 = \frac{1}{M-1}\sum_{m}^{M}\left(\hat{y}_m^L - \frac{1}{M}\sum_{m}^{M}\hat{y}_m^L\right)^2 \quad \sigma_{\hat{y}^U}{}^2 = \frac{1}{M-1}\sum_{m}^{M}\left(\hat{y}_m^U - \frac{1}{M}\sum_{m}^{M}\hat{y}_m^U\right)^2. \tag{6}$$

Consequently, the prediction uncertainty can be quantified as the range of $[\overline{\hat{y}}^L\ \overline{\hat{y}}^U]$.

2.3 Full interval neural networks

Unlike light-INNs and BNNs, the alternative approach to INNs is to utilize intervals over parameters (referred to as full-INNs in this work). The full-INN also incorporates deterministic interval inputs and outputs for each node and therefore conducts the interval forward propagation as follows [21, 22]:

$$[\underline{\boldsymbol{a}}, \overline{\boldsymbol{a}}]^l = f^l\Big([\underline{\boldsymbol{W}}, \overline{\boldsymbol{W}}]^l \odot [\underline{\boldsymbol{a}}, \overline{\boldsymbol{a}}]^{l-1} \oplus [\underline{\boldsymbol{b}}, \overline{\boldsymbol{b}}]^l\Big), \tag{7}$$

where $\oplus$, $\ominus$, and $\odot$ represent interval addition, subtraction, and multiplication, respectively [36]. The quantities $[\underline{\boldsymbol{a}}, \overline{\boldsymbol{a}}]^l$, $[\underline{\boldsymbol{a}}, \overline{\boldsymbol{a}}]^{l-1}$, $[\underline{\boldsymbol{W}}, \overline{\boldsymbol{W}}]^l$ and $[\underline{\boldsymbol{b}}, \overline{\boldsymbol{b}}]^l$ are the interval outputs of the l-th, the

interval outcomes of the preceding $(l-1)$-th layer, the interval weights and interval biases of the l-th layer, accordingly. $f^l(\cdot)$ corresponds to the activation function of the l-th layer, which is constrained to be monotonically increasing. When $[\underline{\boldsymbol{a}}, \overline{\boldsymbol{a}}]^{l-1}$ is non-negative, equation (7) can be simplified as follows [21, 22]:

$$\begin{aligned} \underline{\boldsymbol{a}}^l &= f^l\left(\min\{\underline{\boldsymbol{W}}^l, \boldsymbol{0}\}\overline{\boldsymbol{a}}^{l-1} + \max\{\underline{\boldsymbol{W}}^l, \boldsymbol{0}\}\underline{\boldsymbol{a}}^{l-1} + \underline{\boldsymbol{b}}^l\right) \\ \overline{\boldsymbol{a}}^l &= f^l\left(\max\{\overline{\boldsymbol{W}}^l, \boldsymbol{0}\}\overline{\boldsymbol{a}}^{l-1} + \min\{\overline{\boldsymbol{W}}^l, \boldsymbol{0}\}\underline{\boldsymbol{a}}^{l-1} + \overline{\boldsymbol{b}}^l\right) \end{aligned}. \tag{8}$$

In our implementation of full-INNs, we adopt the methodology outlined in [22] to effectively ensure the validity of weight and bias intervals ($\underline{\boldsymbol{W}} \leq \overline{\boldsymbol{W}}$ and $\underline{\boldsymbol{b}} \leq \overline{\boldsymbol{b}}$) throughout the propagation process, as detailed below:

$$[\underline{\boldsymbol{W}}, \overline{\boldsymbol{W}}] = [\boldsymbol{W}_c - \boldsymbol{W}_r, \boldsymbol{W}_c + \boldsymbol{W}_r] \quad [\underline{\boldsymbol{b}}, \overline{\boldsymbol{b}}] = [\boldsymbol{b}_c - \boldsymbol{b}_r, \boldsymbol{b}_c + \boldsymbol{b}_r]. \tag{9}$$

Here, $\boldsymbol{W}_c$, $\boldsymbol{b}_c$ and $\boldsymbol{W}_r \geq \boldsymbol{0}$, $\boldsymbol{b}_r \geq \boldsymbol{0}$ are the centers (midpoints) and radii (half of ranges) of the weight and bias intervals, accordingly [22]. In this context, full-INNs can be initialized and trained using standard techniques [21, 22]. Additionally, interval batch normalization (IBN) [22], an interval-based extension of the conventional batch normalization technique, has been employed to improve the stability and adaptability of deep architectures. Figure 1 presents a simple MLP-based full-INN architecture for a univariate regression task.

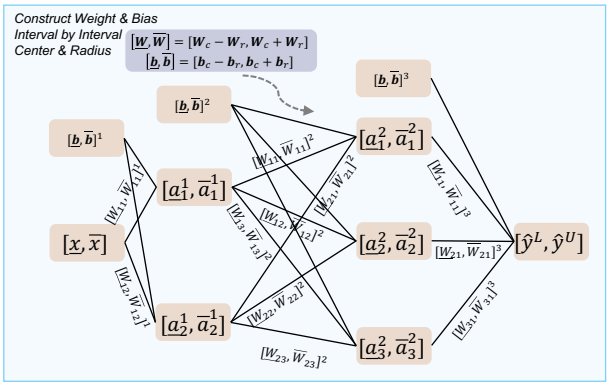


Figure 1: A simple MLP-based full-INN architecture for a univariate regression task. [22].

Given a standard training dataset $\mathbb{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^N$, the full-INN can be trained by setting $\underline{\boldsymbol{x}}_n = \overline{\boldsymbol{x}}_n = \boldsymbol{x}_n$ for the input and minimizing the following loss [21]:

$$\sum\nolimits_{n=1}^{N} \Big(\max\{y_n - \hat{y}_n^U, 0\}^2 + \max\{\hat{y}_n^L - y_n, 0\}^2 + \beta \cdot (\hat{y}_n^U - \hat{y}_n^L) \Big), \tag{10}$$

in which $[\hat{y}_n^L \ \hat{y}_n^U]$ is the full-INN prediction for the n-th data point, and the hyperparameter parameter $\beta > 0$ determines how outlier-sensitive the intervals are trained [21]. The prediction interval range subsequently functions as a quantitative measure of prediction uncertainty.

In addition to standard datasets, full-INNs also encode the ability to handle uncertain interval-valued data. Specifically, when training data incorporate interval target values, i.e., $\{\boldsymbol{x}_n, [y_n^L \ y_n^U]\}_{n=1}^N$, one of widely applied training strategies for full-INN is to minimize the following loss function [37]

$$\sum\nolimits_{n=1}^{N} \frac{1}{2}\Big((\hat{y}_n^L - y_n^L)^2 + (\hat{y}_n^U - y_n^L)^2 \Big). \tag{11}$$

3 EXPERIMENTAL EVALUATION

In this section, we conduct three case evaluation studies: a one-dimensional analytical function in Section 3.1, a real-world remaining useful lifetime (RUL) prediction task in Section 3.2.1, and the RUL task involving interval-valued data (IVD) in Section 3.2.2.

3.1 One-dimensional analytical function

In our evaluation, we consider a one-dimensional analytical function, as follows:

$$f(x) = x\sin(x) + \epsilon_1 x + \epsilon_2, \tag{12}$$

where $\epsilon_1, \epsilon_2 \sim \mathcal{N}(0, 0.3)$. This function incorporates both homoscedastic (ϵ_2) and heteroscedastic (ϵ_1) aleatoric uncertainty, requiring the model to estimate both components. We generate 1,000 training samples with $x \in [0, 10]$, while out-of-training-set samples are constructed using 200 samples for $x \in [10, 15]$.

For Bayesian baselines, we train five SNNs with different random seed initializations to form deep ensembles and apply MC Dropout, both utilizing the Gaussian NLL loss in equation (1). Similarly, for INNs, we train five light-INNs with different random seed initializations to construct ensembles of light-INNs, employing the loss function in equation (4) with $\lambda = 5$. However, we are unable to obtain a well-trained full-INN model using the training strategy in equation (10), as severe underfitting is observed across various values of the training hyperparameter β. Specifically, all models follow the same MLP architecture and training settings as outlined in [9].

Figure 2 illustrates the comparison of the three approaches for prediction uncertainty. Compared to Bayesian baselines, the ensemble of light-INNs performs best in the out-of-training-set region due to its higher estimated prediction uncertainty. However, in the in-training-set region, the smoothness of the predicted lower boundary—particularly for $x \in [6, 10]$ is inferior to that of deep ensembles. This is primarily due to the incorporation of the training data's coverage proportion in the training procedure of light-INNs, and the training noise is largely increased. Additionally, we do not present results for MC Dropout with larger sampling numbers M because increasing M from 5 to 100 did not significantly affect its poor performance.

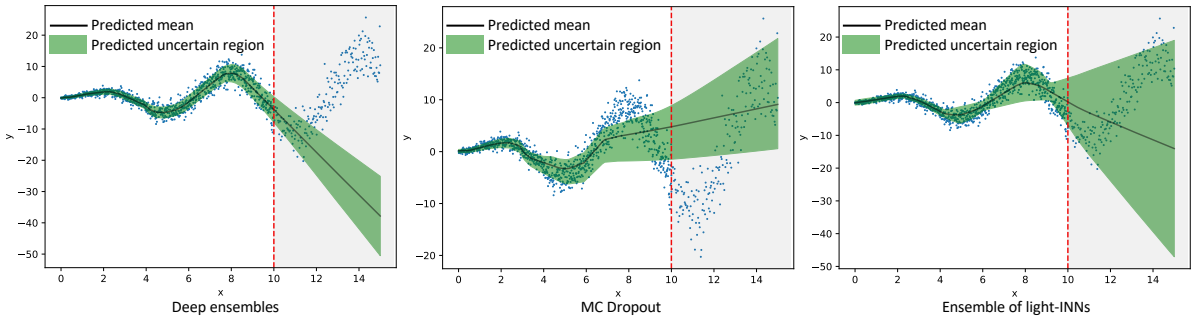


Figure 2: Comparison of three approaches for prediction uncertainty quantification in a one-dimensional analytical function regression. The predicted uncertainty regions for deep ensembles and MC Dropout are computed using $\mu_* \pm \sigma_*$ whereas for the ensemble of light-INNs, it is determined by $\hat{y}^U - \hat{y}^L$. The out-of-training-set region is marked in light gray.

3.2 Jet engine remaining useful lifetime prediction

Remaining useful lifetime (RUL) prediction plays a pivotal role in prognostics and health management (PHM), enhancing system reliability while ultimately reducing operational and maintenance costs for mechanical systems. Given its predictive nature, accuracy has been the primary focus of research efforts. To improve prediction performance, various models have been proposed, including Kalman filter-ensemble multi-layer perception (MLP) [38], recurrent neural networks (RNN) [39], and deep convolutional neural networks (DCNN) [40–42]. However, the presence of noise in the data as well as assumptions being used to generate the data, have heightened the importance of uncertainty estimation. In this test case, we explore how INNs could serve as a viable alternative to the existing Bayesian methods.

In this evaluation, we use the NASA turbofan engine degradation simulation dataset generated by C-MAPSS platform [43]. The dataset consists of four sub-datasets simulated under different two working conditions and two fault modes. A total of 21 sensor measurements are collected in each sub-dataset. However, some of the sensor measurements were dropped since they didn't give any useful information (i.e. variances are smaller than 1×10^{-6}), leaving the remaining set of 14 sensors which numbered as sensor numbers: 2,3,4,7,8,9,11,12,13,14,15,17,20, and 21. Following [42], we choose the FD001 and FD003 sub-datasets to be evaluated in this test case since they share the same operating conditions but have different fault modes. Since the problem is a time-series prediction, we employed a sliding window embedding method to generate the training and testing samples, where the window length is set to 30. In pre-processing the target label, we adopted a piecewise linear function with a constant RUL equal to 125 in the early stage [41, 42].

A modified and shortened architecture of ResNet, called MResNet9 [42], shown in Figure 3 is used as the main neural network architecture to predict the RUL. The number before *Conv2d* represents the convolution kernel size, the number after *Conv2d* is the quantity of the convolution kernels, and $/2$ means the dimensions are reduced by half. *BN* represents the batch normalization, *ReLU* represents the rectified linear unit activation function, and *Dropout* represents the dropout layer with 0.2 rate.

3.2.1 Light-INN approach

In this section, we evaluate the ensemble of light-INNs while choosing MC Dropout and deep ensembles as baselines. The ensemble number is set as $M = 5$. To evaluate the model's performance, we primarily consider two aspects: prediction accuracy and uncertainty. To ensure a fair comparison of accuracy, we employ two metrics. First, following the original C-MAPSS data challenge [43], we utilize the scoring function defined in equation 13, where N represents the number of engines in the test set, and $h = (\text{Estimated RUL} - \text{True RUL})$.

$$\text{SF} = \begin{cases} \sum_{i=1}^{N} \left(e^{-\frac{h_i}{13}} - 1 \right), & \text{for } h_i < 0 \\ \sum_{i=1}^{N} \left(e^{\frac{h_i}{10}} - 1 \right), & \text{for } h_i \geq 0 \end{cases}. \tag{13}$$

The scoring function (SF) is intentionally designed to be asymmetric, penalizing late predictions more severely than early ones. This design choice reflects the aerospace industry's priority, where delayed maintenance can be significantly more detrimental than the premature allocation of maintenance resources. In addition to the scoring function, we employ the standard root mean

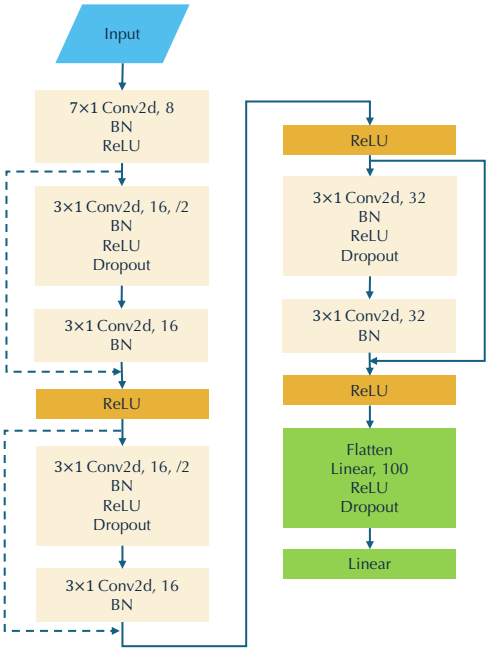


Figure 3: MResNet9 architecture.

squared error (RMSE) of the predicted RUL, defined as follows:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} h_i^2}, \tag{14}$$

A Log-likelihood function is often used to assess the prediction uncertainty quality. However, this approach assumes that the uncertainty estimation outputs a normal distribution with mean $\hat{f}(\boldsymbol{x})$ and standard deviation σ_{pred}. Due to the nature of interval prediction where the output distribution is unknown, this approach is not suitable. Instead, we employ the prediction interval coverage probability (PICP), defined in equation (3).

The main drawback of the PICP comes when the model returns a very wide prediction interval, thus each of the test set observations falls inside the prediction interval. While the PICP score will be exceptional, this behaviour is unwanted since such behaviour is essentially a completely random process. Thus, we also employ sharpness, or the average interval length in conjunction with PICP to indicate the width of the interval, defined as follows:

$$\text{IL} = N^{-1}\sum_{i=1}^{N} \bar{\hat{y}}_i^U - \bar{\hat{y}}_i^L. \tag{15}$$

Where IL indicates the length of the predicted $[\bar{\hat{y}}^L \; \bar{\hat{y}}^U]$ of the ensemble of light-INNs. In the context of deep ensembles and MC Dropout, the prediction interval applied in equation (15) (2) is obtained from $[\mu_* - \sigma_* \; \mu_* + \sigma_*]$. The best model would be the one that maximizes the PICP while keeping the interval length minimum.

The comparison of precision metrics given in Table 1 shows that the deep ensemble ensemble and the ensemble of light-INNs are very close to each other in terms of predictive performance. While the MC Dropout, despite having a similar architecture is not as accurate as the others. In terms of uncertainty estimation shown in Table 2 and Figures 4 and 5, the ensemble of

Table 1: Accuracy metrics comparison between methods.

Datasets	FD001		FD003	
Metrics	RMSE	SF	RMSE	SF
MC Dropout	15.13 ± 1.02	579.75 ± 498.51	15.05 ± 0.77	534.58 ± 120.21
Deep ensembles	12.65 ± 0.17	247.61 ± 8.26	12.48 ± 0.41	361.57 ± 60.60
Ensemble of light-INNs	12.96 ± 0.50	260.63 ± 31.31	13.11 ± 0.61	483.64 ± 178.88

Table 2: Uncertainty metrics comparison between methods.

Datasets	FD001		FD003	
Metrics	PICP	IL	PICP	IL
MC Dropout	0.75 ± 0.03	38.74 ± 1.27	0.79 ± 0.03	38.96 ± 1.48
Deep ensembles	0.88 ± 0.01	39.50 ± 0.42	0.90 ± 0.01	39.19 ± 0.61
Ensemble of light-INNs	0.35 ± 0.03	12.55 ± 0.30	0.42 ± 0.03	11.99 ± 0.39

light-INNs produces a rather over-confident estimation compared to the other methods. In addition, our experiments indicate that the performance of light-INN method is more sensitive to hyperparameters and training configurations. This heightened sensitivity primarily stems from the finely designed loss function (equation (4)) used in light-INN training, which necessitates a careful balance between the accuracy of mean prediction and the quality of PCIP.

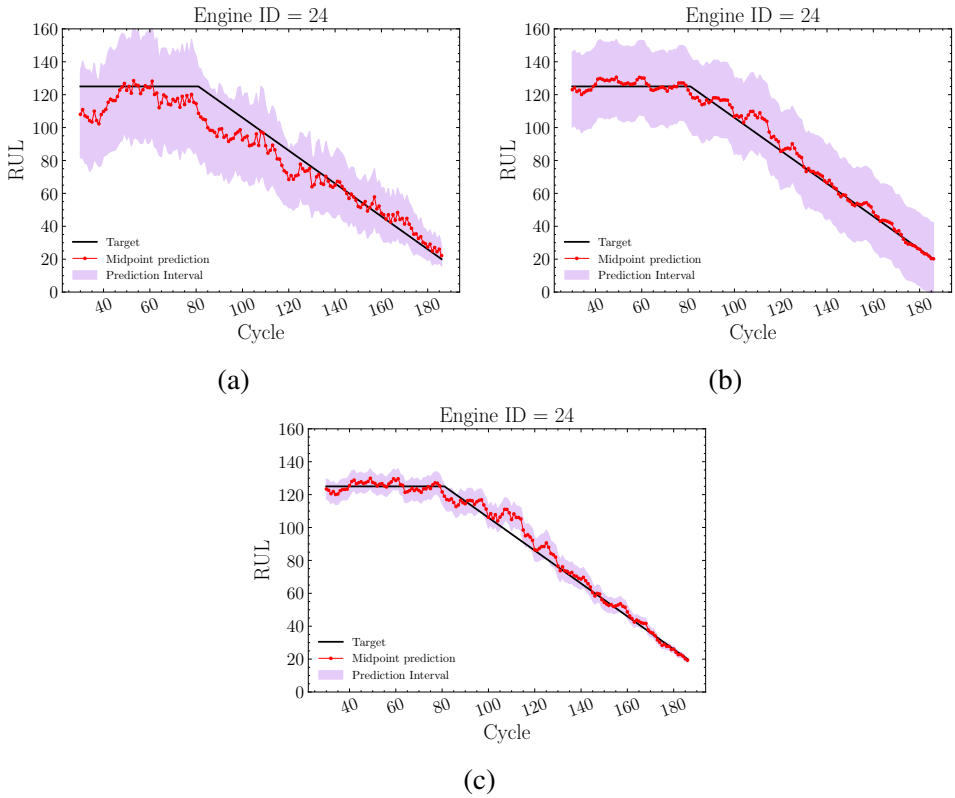


Figure 4: Test data FD001 engine 24 RUL prediction using (a) MC Dropout, (b) deep ensembles, and (c) ensemble of light-INNs.

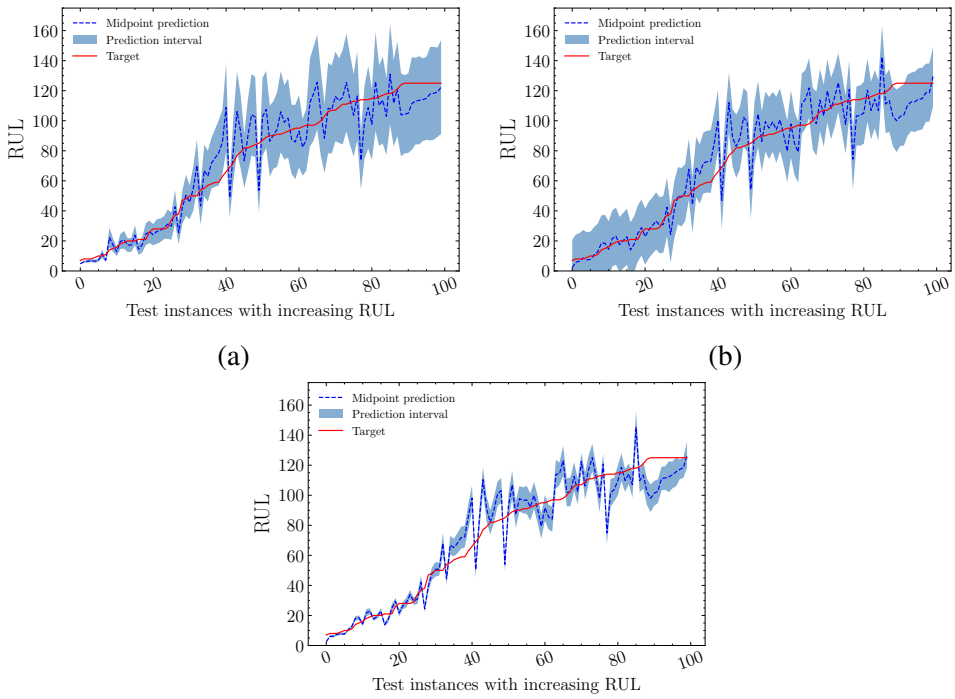


Figure 5: Test data FD001 instances prediction with increasing RUL using (a) MC Dropout, (b) deep ensembles, and (c) ensemble of light-INNs.

3.2.2 Full-INN approach

In this section, we explore the capability of full-INNs to handle imprecise data (interval-valued data). In the RUL test case, we modify the existing dataset to have an imprecise value by adding and subtracting the existing data with a random value between 10 and 20, written as:

$$\begin{aligned} y_i^{\mathrm{U}} &= y_i + \mathcal{U}(10,20) \\ y_i^{\mathrm{L}} &= y_i - \mathcal{U}(10,20) \end{aligned}, \tag{16}$$

where the minimum and maximum values sampled from the uniform distribution $\mathcal{U}$ are arbitrary.

Converting the modified ResNet architecture in Figure 3 into the full-INN architecture can be done by replacing the corresponding layers and activation function types with their interval-valued layers counterpart [21, 22]. To train the full interval-version MResNet9, we employ the loss function in equation (11).

To evaluate the full-INN, we propose several different metrics than the light-INN. Since we have interval-valued data (IVD) in the full-INN, we first propose to use a modified version of RMSE, where the final value is the average of the lower and upper RMSE, written as:

$$\text{I-RMSE} = \frac{1}{2}\text{RMSE}\left(\hat{y}^{\mathrm{L}} - y^{\mathrm{L}}\right)^2 + \frac{1}{2}\text{RMSE}\left(\hat{y}^{\mathrm{U}} - y^{\mathrm{U}}\right)^2. \tag{17}$$

We also slightly modified the scoring function (SF) in equation 13 to be suited for IVD. At its core, we still follow the original SF paradigm, which penalizes late predictions more than early predictions. However, we also define a safe space, which is defined as the area between the upper

Table 3: Accuracy metrics comparison between methods.

Datasets	FD001		FD003	
Metrics	I-RMSE	I-SF	I-RMSE	I-SF
full-INN	16.22 ± 1.00	558.53 ± 184.83	15.34 ± 0.78	616.24 ± 151.08

bound and the lower bound ground truth. Therefore, early prediction of the upper bound and late prediction of the lower bound are not penalized. The interval SF is defined as:

$$\text{I-SF} = \begin{cases} \sum_{i=1}^{N} \left(e^{-\frac{h^L}{13}} - 1\right) + \left(e^{\frac{h^U}{10}} - 1\right), & \text{for } h^L < 0 \;\&\; h^U \geq 0 \\ \sum_{i=1}^{N} \left(e^{-\frac{h^L}{13}} - 1\right), & \text{for } h^L < 0 \;\&\; h^U < 0 \\ \sum_{i=1}^{N} \left(e^{\frac{h^U}{10}} - 1\right), & \text{for } h^L \geq 0 \;\&\; h^U \geq 0. \\ 0, & \text{else.} \end{cases} \tag{18}$$

PICP and IL metrics are used to quantify the quality of uncertainty estimation in the light-INN case in section 3.2.1. However, since the problem is interval-valued we propose to use two metrics to evaluate the uncertainty estimation in full-INNs. First, we introduce the area coverage ratio (ACR) defined as the ratio of the overlapping area between the predicted interval and the true interval relative to the total area of the true interval. The expression is written as:

$$\begin{aligned} \text{ACR} &= \frac{A_{\text{overlap}}}{A_{\text{true}}} \\ A_{\text{true}} &= \int_{x \in X} (y_{\text{up}}(x) - y_{\text{low}}(x)) dx \\ A_{\text{overlap}} &= \int_{x \in X} \max(0, \min(\hat{y}_{\text{up}}(x), y_{\text{up}}(x)) - \max(\hat{y}_{\text{low}}(x), y_{\text{low}}(x))) \, dx \end{aligned} \tag{19}$$

The ACR can be interpreted as how much the model correctly predicts the true interval. Thus, a higher ACR indicates a better model. However, ACR can be easily achieved if the model predicts a very wide interval that encloses the whole space. Thus, to avoid such conditions, we introduce the non-overlap (outside) area ratio (OAR). The OAR is defined as the non-overlapping area of the predicted interval relative to the total area of the predicted interval, written as:

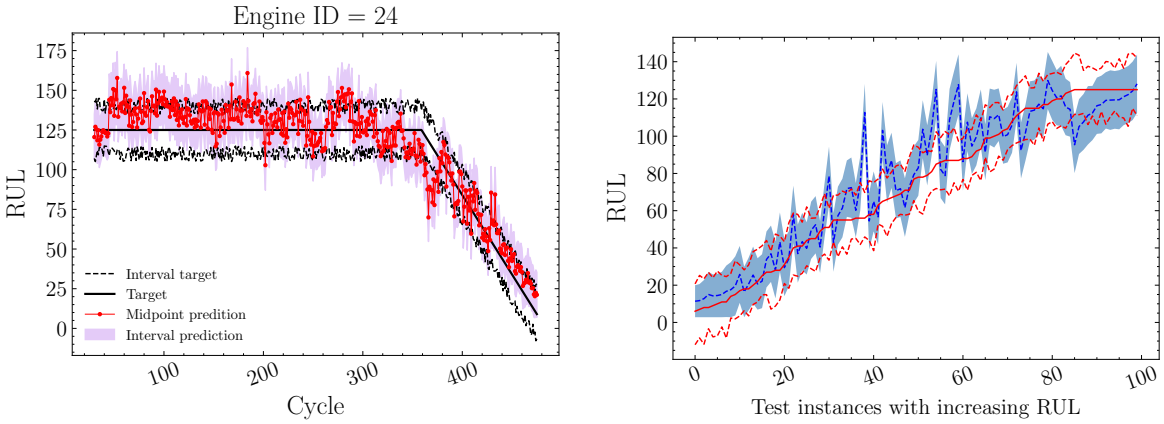
$$\begin{aligned} \text{OAR} &= \frac{A_{\text{pred}} - A_{\text{overlap}}}{A_{\text{pred}}} \\ A_{\text{pred}} &= \int_{x \in X} (\hat{y}_{\text{up}}(x) - \hat{y}_{\text{low}}(x)) dx \end{aligned} \tag{20}$$

Intuitively, the OAR measures how much the model mispredicts the true interval. Thus, lower OAR indicates a better model.

The prediction plot is given in Figure 6. Tables 3 and 4 show the value of the proposed metrics for the full-INN model with 10 repetitions. However, because there are no comparable methods and models to learn from RUL data, no methodological comparison is provided. Instead, we provide the result as a benchmark for future research.

Table 4: Uncertainty metrics comparison between methods.

Datasets	FD001		FD003	
Metrics	ACR	OAR	ACR	OAR
full-INN	0.60 ± 0.03	0.37 ± 0.02	0.62 ± 0.03	0.35 ± 0.02



(a) (b)

Figure 6: (a) Interval prediction on engine 24, and (b) Test instances prediction with sorted increasing RUL.

4 CONCLUSION

In this study, we evaluated two types of interval neural networks (INNs)—light-INNs and full-INNs—for uncertainty quantification in regression tasks. Our evaluation covered standard regression, a realistic remaining useful lifetime (RUL) prediction task, and the RUL task under interval-valued data (IVD) settings. The results suggest that light-INNs have the potential to serve as an alternative approach for uncertainty quantification. However, they require a careful trade-off between mean prediction performance and the quality of prediction interval coverage probability, particularly when handling complex problems.

Furthermore, the capability of full-INNs to process interval-based data and integrate with deep architectures was validated through the RUL task with IVD settings. However, their performance on standard datasets in our test cases was suboptimal, indicating the need for further investigation into training strategies and the inherent behavior of these models.

For future work, we aim to deepen our understanding of the training dynamics and loss function design of INNs. Additionally, we plan to explore their performance in time-series forecasting and partial differential equation problems.

ACKNOWLEDGMENT

This work was supported by granted H2020 FETOPEN-2018-2019-2020-01 European project, *Epistemic AI* under grant agreement No. 964505 (E-pi).

REFERENCES

[1] Houyu Lu, Sergio Cantero-Chinchilla, Xin Yang, Konstantinos Gryllias, and Dimitrios Chronopoulos. Deep learning uncertainty quantification for ultrasonic damage identification in composite structures.

Composite Structures, 338:118087, 2024.

[2] Lavi Rizki Zuhal, Ghifari A. Faza, Pramudita S. Palar, and Rhea P. Liem. Fast and adaptive reliability analysis via kriging and partial least squares. In *AIAA Scitech 2021 Forum*. American Institute of Aeronautics and Astronautics, January 2021.

[3] Gang Sun and Shuyue Wang. A review of the artificial neural network surrogate modeling in aerodynamic design. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 233(16):5863–5872, 2019.

[4] David A Nix and Andreas S Weigend. Estimating the mean and variance of the target probability distribution. In *Proceedings of 1994 ieee international conference on neural networks (ICNN'94)*, volume 1, pages 55–60. IEEE, 1994.

[5] Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine learning*, 110(3):457–506, 2021.

[6] Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

[7] Aryan Mobiny, Pengyu Yuan, Supratik K Moulik, Naveen Garg, Carol C Wu, and Hien Van Nguyen. Dropconnect is effective in modeling uncertainty of Bayesian deep networks. *Scientific Reports*, 11(1):1–14, 2021.

[8] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in Neural Information Processing Systems*, 30, 2017.

[9] Matias Valdenegro-Toro and Daniel Saromo Mori. A deeper look into aleatoric and epistemic uncertainty disentanglement. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1508–1516. IEEE, 2022.

[10] Manuel A Vega and Michael D Todd. A variational Bayesian neural network for structural health monitoring and cost-informed decision-making in miter gates. *Structural Health Monitoring*, 21(1):4–18, 2022.

[11] Yongchan Kwon, Joong-Ho Won, Beom Joon Kim, and Myunghee Cho Paik. Uncertainty quantification using Bayesian neural networks in classification: Application to biomedical image segmentation. *Computational Statistics & Data Analysis*, 142:106816, 2020.

[12] Laurent Valentin Jospin, Hamid Laga, Farid Boussaid, Wray Buntine, and Mohammed Bennamoun. Hands-on Bayesian neural networks—A tutorial for deep learning users. *IEEE Computational Intelligence Magazine*, 17(2):29–48, 2022.

[13] Vincent Fortuin. Priors in Bayesian deep learning: A review. *International Statistical Review*, 2022.

[14] Tim Pearce, Alexandra Brintrup, Mohamed Zaki, and Andy Neely. High-quality prediction intervals for deep learning: A distribution-free, ensembled approach. In *International conference on machine learning*, pages 4075–4084. PMLR, 2018.

[15] Krasymyr Tretiak, Georg Schollmeyer, and Scott Ferson. Neural network model for imprecise regression with interval dependent variables. *Neural Networks*, 161:550–564, 2023.

[16] Abbas Khosravi, Saeid Nahavandi, Doug Creighton, and Amir F. Atiya. Lower upper bound estimation method for construction of neural network-based prediction intervals. *IEEE Transactions on Neural Networks*, 22(3):337–346, 2011.

[17] Tárik S. Salem, Helge Langseth, and Heri Ramampiaro. Prediction intervals: Split normal mixture from quality-driven deep ensembles. In Jonas Peters and David Sontag, editors, *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 124 of *Proceedings of Machine Learning Research*, pages 1179–1187. PMLR, 03–06 Aug 2020.

[18] Yuandu Lai, Yucheng Shi, Yahong Han, Yunfeng Shao, Meiyu Qi, and Bingshuai Li. Exploring uncertainty in regression neural networks for construction of prediction intervals. *Neurocomputing*, 481:249–257, 2022.

[19] Hisao Ishibuchi, Hideo Tanaka, and Hidehiko Okada. An architecture of neural networks with interval weights and its application to fuzzy regression analysis. *Fuzzy Sets and Systems*, 57(1):27–39, 1993.

[20] Z.A. Garczarczyk. Interval neural networks. In *2000 IEEE International Symposium on Circuits and Systems (ISCAS)*, volume 3, pages 567–570 vol.3, 2000.

[21] Luis Oala, Cosmas Heiß, Jan Macdonald, Maximilian März, Wojciech Samek, and Gitta Kutyniok. Interval neural networks: Uncertainty scores, 2020.

[22] Kaizheng Wang, Keivan Shariatmadar, Shireen Kudukkil Manchingal, Fabio Cuzzolin, David Moens, and Hans Hallez. CreINNs: Credal-Set Interval Neural Networks for Uncertainty Estimation in Classification Tasks. *Neural Networks*, 185:107198, 2025.

[23] L. Billard and E. Diday. Regression analysis for interval-valued data. In *Studies in Classification, Data Analysis, and Knowledge Organization*, pages 369–374. Springer Berlin Heidelberg, 2000.

[24] Eufrásio de A. Lima Neto and Francisco de A.T. de Carvalho. Constrained linear regression models for symbolic interval-valued variables. *Computational Statistics and Data Analysis*, 54(2):333–347, February 2010.

[25] Ghifari A. Faza, Keivan Shariatmadar, Hans Hallez, and David Moens. Interval reduced order surrogate modelling framework for uncertainty quantification. In *AIAA SCITECH 2024 Forum*. American Institute of Aeronautics and Astronautics, January 2024.

[26] Ronay Ak, Valeria Vitelli, and Enrico Zio. An interval-valued neural network approach for uncertainty quantification in short-term wind speed prediction. *IEEE Transactions on Neural Networks and Learning Systems*, 26(11):2787–2800, 2015.

[27] David Betancourt and Rafi L Muhanna. Interval deep learning for computational mechanics problems under input uncertainty. *Probabilistic Engineering Mechanics*, 70:103370, 2022.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[29] Radford M Neal et al. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2(11):2, 2011.

[30] Matthew D Hoffman, Andrew Gelman, et al. The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623, 2014.

[31] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International conference on machine learning*, pages 1613–1622. PMLR, 2015.

[32] Hendrik A Mehrtens, Alexander Kurz, Tabea-Clara Bucher, and Titus J Brinker. Benchmarking common uncertainty estimation methods with histopathological images under domain shift and label noise. *Medical image analysis*, 89:102914, 2023.

[33] Bálint Mucsányi, Michael Kirchhof, and Seong Joon Oh. Benchmarking uncertainty disentanglement: Specialized uncertainties for specialized tasks. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

[34] Kaizheng Wang, Fabio Cuzzolin, Shireen Kudukkil Manchingal, Keivan Shariatmadar, David Moens, and Hans Hallez. Credal deep ensembles for uncertainty quantification. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[35] Kaizheng Wang, Fabio Cuzzolin, Keivan Shariatmadar, David Moens, and Hans Hallez. Credal wrapper of model averaging for uncertainty estimation in classification. In *The Thirteenth International Conference on Learning Representations*, 2025.

[36] Timothy Hickey, Qun Ju, and Maarten H Van Emden. Interval arithmetic: From principles to implementation. *Journal of the ACM (JACM)*, 48(5):1038–1068, 2001.

[37] Hisao Ishibuchi, Hideo Tanaka, and Hidehiko Okada. An architecture of neural networks with interval weights and its application to fuzzy regression analysis. *Fuzzy Sets and Systems*, 57(1):27–39, 1993.

[38] Leto Peel. Data driven prognostics using a kalman filter ensemble of neural network models. In *2008 International Conference on Prognostics and Health Management*, pages 1–6, 2008.

[39] Felix O. Heimes. Recurrent neural networks for remaining useful life estimation. In *2008 International Conference on Prognostics and Health Management*, pages 1–6, 2008.

[40] Giduthuri Sateesh Babu, Peilin Zhao, and Xiao-Li Li. *Deep Convolutional Neural Network Based Regression Approach for Estimation of Remaining Useful Life*, page 214–228. Springer International Publishing, 2016.

[41] Xiang Li, Qian Ding, and Jian-Qiao Sun. Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliability Engineering and System Safety*, 172:1–11, April 2018.

[42] Zhibin Zhao, Jingyao Wu, David Wong, Chuang Sun, and Ruqiang Yan. Probabilistic remaining useful life prediction based on deep convolutional neural network. *SSRN Electronic Journal*, 2020.

[43] Abhinav Saxena, Kai Goebel, Don Simon, and Neil Eklund. Damage propagation modeling for aircraft engine run-to-failure simulation. In *2008 International Conference on Prognostics and Health Management*, pages 1–9, 2008.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

FORWARD UNCERTAINTY PROPAGATION FOR FINITE ELEMENT MODELS WITH NON-GAUSSIAN PARAMETERS

Renan M. Barros[1], **Iuri B. C. M. Rocha**[1], **Maria Nogal**[1] **and Oswaldo Morales-Nápoles**[1]

[1]Faculty of Civil Engineering and Geosciences, Delft University of Technology
Stevinweg 1, 2628 CN Delft, The Netherlands
e-mail: r.m.barros, i.rocha, m.nogal, o.moralesnapoles@tudelft.nl

Abstract. *Due to the inherent uncertainties in manufacturing properties and intrinsic variability of materials, the assumption of homogeneous input variables is generally not justified. As a result, stochastic forward problems have emerged as a tool to incorporate these uncertainties into numerical simulations, improving model prediction capability in structural analyses. Although most of the existing methods focus on the consideration of stochastic loading, the recently developed statFEM employs a Bayesian paradigm to incorporate data and propagate uncertainties from random physical properties in finite element models. This tool, however, is not developed for cases when Gaussian assumptions are inadequate. The present work provides a copula-based approach embedded into the statFEM methodology to propagate uncertainty from arbitrarily distributed physical properties. The random variables are defined in terms of a Gaussian Copula Process, where samples are drawn from a latent variable governed by a Gaussian Process and then brought respectively to copula and marginal spaces, producing random variables with the desired distribution while retaining the usually desired smooth Gaussian dependence in the spatial domain. The quality of the approximated results is then assessed in a simplified 1D Poisson problem by comparing with Monte Carlo sampling results for different random diffusion coefficients, demonstrating that the method is capable of providing good responses for non-Gaussian physical parameters.*

Keywords: Stochastic PDEs, Non-Gaussian variables, Finite elements, Copula Processes.

1 INTRODUCTION

The manufacturing of industrial components and the performance of engineering systems usually relies on finite element (FE) models to better determine their physical behavior. These processes are laden with uncertainties, as material properties, geometry, and loading may vary over time and under repeated production. For that end, many physical quantities can be defined by a Gaussian random field (or transformations thereof), ensuring a smooth spatial dependence that often reflects the natural continuity in materials [1].

Alternatively, copula-based approaches can also be used to determine a dependence structure between multiple random variables. Despite being more commonly used in fields such as risk analysis and financial assessment to model time variability, they have also been employed for characterizing stochasticity in the spatial domain [2]. As a feature, copula functions decouple the dependence between random variables from their marginal distributions; but, in the trade-off, the variations within the spatial domain usually do not present a smooth behavior.

In order to propagate the uncertainties to the analysis output, e.g. displacements, statFEM is a statistical tool that provides a methodology for synthesizing measurement data and finite element models [3]. It decomposes the observed data and employs a Bayesian approach to update prior beliefs. In turn, the prior for the finite element component is obtained through a probabilistic forward problem by adopting Gaussian parameters and solving a linear stochastic partial differential equation. It has been used for real-world applications [4], and has also been enhanced with other approaches to solve nonlinear PDEs [5], but solely considering stochastic forcing.

The stochastic behavior in the physical properties is usually not accounted for within stochastic forward problem frameworks, as it complicates the uncertainty propagation process; and even when it is, e.g., as a random diffusion coefficient in [3], it is limited to a Gaussian-distributed variable. However, the quantities related to material properties and geometry are often bounded by conventional standards. Moreover, if there is a lack of information on a random variable, not assuming a bell-shaped distribution may be reasonable. In both cases, using Gaussian marginal distributions is not suitable. To address this gap, a copula-based statFEM approach is proposed to propagate the uncertainty from random physical properties with arbitrary marginal distributions, while keeping the desired smooth spatial variability on the domain. The effectiveness of the presented methodology is assessed by a 1D Poisson toy problem with distinct input distributions.

The rest of the paper is outlined as follows: in Section 2, the statFEM framework is explained, focusing on the stochastic forward problem. In Section 3, the concepts of copulas and copula processes are both presented, while in 3.3, the proposed novel methodology is detailed. The method is applied to a case study in Section 4, presenting results and discussion. Finally, the conclusions of the paper are drawn in Section 5.

2 STATISTICAL FINITE ELEMENT METHOD (statFEM)

This section briefly expands on the statFEM original formulation, which employs a Bayesian statistical approach to integrate finite element models and measurement data. It additively decomposes data into three components: a scaled FE prior, a model misspecification error, and a noise component. Although Bayes' rule can then be used to infer posterior densities and hyper-parameters, the main contribution in this study is related to the FE prior response; hence, solely this part of the formulation is briefly introduced. The interested reader is referred to [3] for an in-depth description.

2.1 Probabilistic forward formulation

For the FE stochastic forward problem, the Poisson equation is considered on a domain $\Omega \subset \mathbb{R}^n$ and boundary $\partial\Omega$:

$$\begin{aligned} -\nabla \cdot (\mu(\mathbf{x})\nabla u(\mathbf{x})) &= f(\mathbf{x}), \quad \text{in } \Omega \\ u(x) &= 0, \qquad \text{on } \partial\Omega \end{aligned} \tag{1}$$

where $u(\mathbf{x}) \in \mathbb{R}$ is the unknown field, ∇ is the vector differential operator, $\mu(\mathbf{x})$ and $f(\mathbf{x})$ are respectively a random diffusion coefficient and a random forcing term. Both diffusion coefficient and forcing fields are assumed to be Gaussian Processes:

$$\begin{aligned} \mu(\mathbf{x}) &\sim \mathcal{GP}(\overline{\mu}(\mathbf{x}),\, c_\mu(\mathbf{x}, \mathbf{x}') \\ f(\mathbf{x}) &\sim \mathcal{GP}(\overline{f}(\mathbf{x}),\, c_f(\mathbf{x}, \mathbf{x}') \end{aligned} \tag{2}$$

where $\overline{\mu}(\mathbf{x}) = \mathbb{E}[\mu(\mathbf{x})]$ and $\overline{f}(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})]$ are simplified to be mean values evaluated pointwise, c_μ and c_f are the covariance kernels. A squared exponential kernel is adopted in both cases:

$$c_\alpha(\mathbf{x}, \mathbf{x}') = \sigma_\alpha^2 \, \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|}{2l_\alpha^2}\right) \tag{3}$$

in which $\sigma_\alpha \in \mathbb{R}^+$ is the scaling parameter and $l_\alpha \in \mathbb{R}^+$ is the length scale parameter of the random variable α.

After the domain Ω is subdivided into a set of non-overlapping elements, and the unknown field $u(\mathbf{x})$ is approximated by linear Lagrange basis functions, the discretization of the weak form in Eq. (1) yields the linear system of equations:

$$\mathbf{A}(\boldsymbol{\mu})\mathbf{u} = \mathbf{f} \tag{4}$$

where $\mathbf{A}(\boldsymbol{\mu})$ is the system matrix, $\boldsymbol{\mu}$ is the vector of diffusion coefficients, $\mathbf{u}$ is the nodal displacements vector, and $\mathbf{f}$ is the nodal source vector.

The diffusion coefficient is assumed constant in each element, being evaluated at the element centroids. The diffusion coefficient vector can then be expressed by the multivariate Gaussian density:

$$\boldsymbol{\mu} \sim p(\boldsymbol{\mu}) = \mathcal{N}\left(\overline{\boldsymbol{\mu}}\left(\mathbf{X}^{(c)}\right),\, \boldsymbol{\Sigma}_\mu\left(\mathbf{X}^{(c)}, \mathbf{X}^{(c)}\right)\right) \tag{5}$$

where the mean vector $\overline{\boldsymbol{\mu}}$ and covariance matrix $\boldsymbol{\Sigma}_\mu$ are evaluated with respect to the matrix of centroid coordinates $\mathbf{X}^{(c)}$. Then, the element system matrices are defined by:

$$\mathbf{A}_{ij}^e = \int_{\Omega_e} \mu_e \frac{\partial \phi_i(\mathbf{x})}{\partial \mathbf{x}} \frac{\partial \phi_j(\mathbf{x})}{\partial \mathbf{x}} \, d\Omega_e \tag{6}$$

in which μ_e is the diffusion coefficient of the element with index e and $\phi_{i,j}$ are from the local set of shape functions. On the other hand, the source vector is given by the multivariate Gaussian density:

$$\mathbf{f} \sim p(\mathbf{f}) = \mathcal{N}(\bar{\mathbf{f}}, \boldsymbol{\Sigma}_f). \tag{7}$$

Next, it is possible to express the probability density for a solution vector $\mathbf{u}$ in terms of a diffusion coefficient $\boldsymbol{\mu}$ by solving Eq. (4):

$$\mathbf{u} = \mathbf{A}(\boldsymbol{\mu})^{-1}\mathbf{f} \tag{8}$$

If $\boldsymbol{\mu}$ is known, the right-hand side of the equation is a linear transformation of the Gaussian source vector $\mathbf{f}$ such that:

$$p(\mathbf{u}|\boldsymbol{\mu}) = \mathcal{N}\left(\mathbf{A}(\boldsymbol{\mu})^{-1}\bar{\mathbf{f}}, \mathbf{A}(\boldsymbol{\mu})^{-1}\boldsymbol{\Sigma}_f\mathbf{A}(\boldsymbol{\mu})^{-\intercal}\right) \tag{9}$$

Consequently, the unconditional density can be obtained by marginalizing it over $\boldsymbol{\mu}$, yielding:

$$p(\mathbf{u}) = \int p(\mathbf{u}|\boldsymbol{\mu})p(\boldsymbol{\mu})\, d\boldsymbol{\mu} \tag{10}$$

This integral can be evaluated numerically using either Monte Carlo or Markov Chain Monte Carlo sampling, for example. However, since this approach may be impractical for large scale problems, a first-order approximation for $p(\mathbf{u})$ is given by the multivariate Gaussian density:

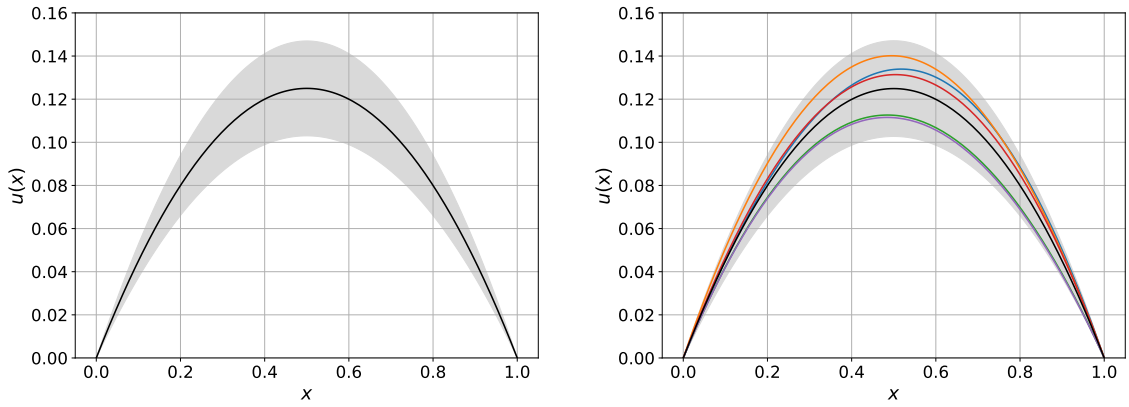
$$p(\mathbf{u}) = \mathcal{N}(\bar{\mathbf{u}}, \boldsymbol{\Sigma}_u) \tag{11}$$

where:

$$\begin{aligned}
&\bar{\mathbf{u}} = \mathbf{A}(\overline{\boldsymbol{\mu}})^{-1}\bar{\mathbf{f}} \\
&\boldsymbol{\Sigma}_u = \mathbf{A}(\overline{\boldsymbol{\mu}})^{-1}\boldsymbol{\Sigma}_f\,\mathbf{A}(\overline{\boldsymbol{\mu}})^{-\intercal} + \\
&\qquad \sum_e \sum_d (\boldsymbol{\Sigma}_\mu)_{ed}\,\mathbf{A}(\overline{\boldsymbol{\mu}})^{-1}\frac{\partial \mathbf{A}(\overline{\boldsymbol{\mu}})}{\partial \mu_e}\mathbf{A}(\overline{\boldsymbol{\mu}})^{-1}(\boldsymbol{\Sigma}_f + \bar{\boldsymbol{f}} \otimes \bar{\boldsymbol{f}})\,\mathbf{A}(\overline{\boldsymbol{\mu}})^{-\intercal}\frac{\partial \mathbf{A}(\overline{\boldsymbol{\mu}})^{\intercal}}{\partial \mu_d}\mathbf{A}(\overline{\boldsymbol{\mu}})^{-\intercal}
\end{aligned} \tag{12}$$

in which the indices e, d are referring to arbitrary elements, as the process is made element-wise.

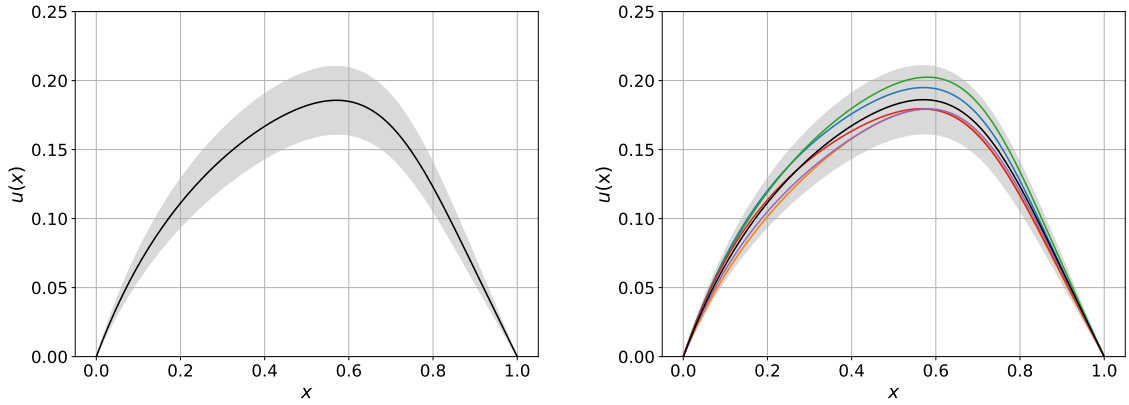
The goodness of this approximation is assessed by comparing it against the much more computationally expensive sampling output. As an example, in Fig. 1 the forcing term is stochastic, with parameters $\overline{f}(x) = 1, \sigma_f = 0.1, l_f = 0.4$, and deterministic diffusion coefficient $\mu = 1$; whereas in Fig. 2 the diffusion coefficient is stochastic, with parameters $\overline{\mu}(x) = 0.7 + 0.3\sin(2\pi x)$, $\sigma_\mu = 0.1$, $l_\mu = 0.25$, and deterministic forcing $f = 1$. It is possible to observe that the approximation provides a good agreement with the numerical response in both cases. The colored lines are samples drawn from Eq. (9) for Fig. 1 and from Eq. (5) then $u(x, \mu)$ for Fig. 2, while the grey shaded areas are the 95% confidence regions for each case.



(a) StatFEM solution.

(b) MC sampling solution.

Figure 1: Comparison between (a) StatFEM and (b) MC sampling for stochastic f. Grey shaded areas represent the 95% confidence regions.



(a) StatFEM solution.

(b) MC sampling solution.

Figure 2: Comparison between (a) StatFEM and (b) MC sampling for stochastic μ. Grey shaded areas represent the 95% confidence regions.

3 COPULA-BASED STATFEM

Throughout this section, a novel methodology for considering stochasticity in physical properties is proposed, with a copula approach being embedded into the previously defined statFEM formulation for uncertainty propagation. The concepts of copulas and copula processes are presented in Sections 3.1 and 3.2, respectively. Then, in Section 3.3, the methodology is explained for both the numerical (sampling) and the linear approximation solutions.

3.1 Copulas

The configuration of scatter plots between random variables is often influenced by their scales and marginal distributions. To better assess their dependence structure, the probability integral transformation can be used in each as a standardization technique, which enables the characterization between uniformly [0,1] distributed random variables. This behavior is illustrated in Fig. (3). Consequently, this approach decouples the dependence structure between random variables from their marginal distributions. More specifically, in this context, the corresponding joint distribution function employed to define the dependence structure between random variables with uniform marginals is known as a copula [6].

Delving into the mathematical foundation, copulas are based on Sklar's Theorem [7]. This theorem states that any multivariate joint distribution can be expressed in terms of both univariate marginal distributions and a copula that captures the dependence structure between the random variables:

$$H(y_1, y_2, ..., y_n) = C(F_1(y_1), F_2(y_2), ..., F_n(y_n)) = C(z_1, z_2, ..., z_n) \tag{13}$$

where H represents the joint distribution function, C is the copula, and F_i denotes the cumulative distribution function (CDF) of each random variable. It is important to remark that, if all F_i are continuous, the copula C is unique.

Furthermore, the inverse relation also holds, allowing the copula to be defined as:

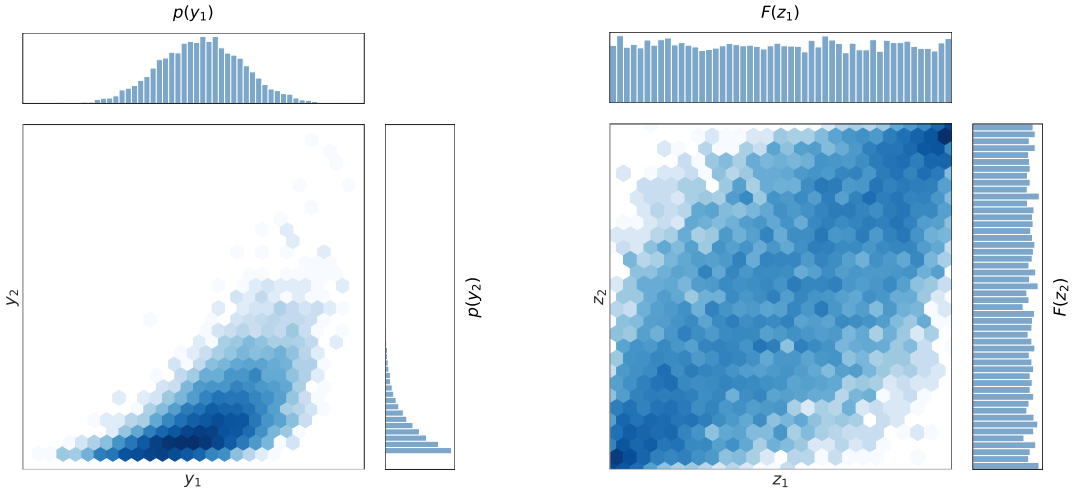
$$C(z_1, z_2, ..., z_n) = H(F_1^{-1}(z_1), F_2^{-1}(z_2), ..., F_n^{-1}(z_n)) \tag{14}$$

This formulation can then be used to construct various types of copulas [8]. This work, however, focuses on the Gaussian copula due to its simplicity and computational efficiency. The

multivariate Gaussian copula is given by:

$$C(\mathbf{u}; \Sigma) = \Phi_{\Sigma}(\Phi^{-1}(z_1), \Phi^{-1}(z_2), ..., \Phi^{-1}(z_n)) \tag{15}$$

where Φ^{-1} denotes the inverse standard normal CDF, and Φ_{Σ} represents the standard multivariate normal distribution with a zero mean vector, unit variances, and symmetric positive definite correlation matrix Σ. Consequently, by substituting $z_i = F_i(y_i)$, the joint distribution H is defined by a Gaussian dependence and arbitrary F_i marginals, as long as they are continuous.



(a) Scatter plot based on probability distribution functions.

(b) Scatter plot based on cumulative distribution functions.

Figure 3: Schematic example of a copula function.

3.2 Copula Processes

Consider a latent variable $\kappa(\mathbf{x})$ that follows a Gaussian process. Sampling from this process yields a collection of Gaussian variables, with their spatial dependence structure encoded by a chosen kernel function. More specifically, in this work, the latent variable $\kappa(\mathbf{x})$ is such that:

$$\kappa(\mathbf{x}) \sim \mathcal{GP}(\overline{\kappa}(\mathbf{x}),\ c_{\kappa}(\mathbf{x}, \mathbf{x}')) \tag{16}$$

where:

$$\overline{\kappa}(\mathbf{x}) = 0, \quad c_{\kappa}(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|}{2l_{\kappa}^2}\right) \tag{17}$$

Thus, the dependence structure is solely influenced by the length scale l_{κ}. These sampled values are then transformed using a Gaussian CDF, providing a collection of $\mathbf{Z}$ uniformly [0,1] distributed latent random variables:

$$\mathbf{Z} = \Phi(\kappa) \tag{18}$$

Consequently, although the samples transition from the latent space to the copula space, their dependence structure remains governed by the underlying Gaussian process. Thus, these resulting uniform samples can be defined as a draw from a Gaussian Copula Process, due to both the inherited dependence structure and the type of CDF used [9].

Moreover, these samples in copula space can also be mapped to another arbitrary marginal distribution by applying the desired CDF, transitioning from copula space to marginal space:

$$\underbrace{\kappa(x) \sim \mathcal{GP}\big(0, c_\kappa(\mathbf{x}, \mathbf{x}')\big)}_{\text{Step 1: Sample from latent GP}} \xrightarrow{\text{Gaussian CDF}} \underbrace{\mathbf{Z} = \Phi(\kappa)}_{\text{Step 2: Map to copula space}} \xrightarrow{\text{Inverse CDF}} \underbrace{\mathbf{Y} = \mathbf{F}^{-1}(\mathbf{Z})}_{\text{Step 3: Transform to marginals}} \tag{19}$$

For instance, if a beta inverse CDF is used, the transformed samples will follow a beta distribution while still maintaining the underlying Gaussian dependence structure. Therefore, this methodology provides an efficient approach to generate samples from any desired marginal distribution while promoting a smooth variation within the spatial domain.

3.3 Incorporating copulas in statFEM

The question that remains is how to arrive at a formulation that provides a version of statFEM that incorporates the Copula Process approach. Primarily, it is necessary to determine how it affects the evaluation of the mean approximation. Given that the diffusion coefficient follows an arbitrary marginal distribution, it is possible to evaluate the expected value over the domain $\overline{\mu}(\mathbf{x}) = \mathbb{E}[\mu(\mathbf{x})]$. Then, this value can be substituted in Eq. (12) to assemble the system matrix and subsequently evaluate the approximated mean.

Next, for the evaluation of the approximated covariance, $\mathbf{\Sigma}_\mu$ needs to be determined. Since a perturbation method is used to evaluate the first order approximation in statFEM [10], the same approach is employed to find the relation between $\mathbf{\Sigma}_\mu$ and $\mathbf{\Sigma}_\kappa$. Given that there is a mapping such that $\mu(x) = g(\kappa(x))$, a first-order Taylor expansion around $\overline{\kappa}$ (with zero mean in this case) results in:

$$g(\kappa(\mathbf{x})) \approx g(0) + g'(0)\kappa(\mathbf{x}) \tag{20}$$

where $g(0)$ is the expected value of μ when $\kappa(x) = 0$, and $g'(0)$ addresses how a small perturbation in $\kappa(\mathbf{x})$ leads to a change in μ. The perturbation of μ is:

$$\delta\mu = \mu - g(0) \xrightarrow{\text{Linearization}} \delta\mu \approx g'(0)\kappa(\mathbf{x}) \tag{21}$$

Then, the covariance of $\delta\mu$ is given by:

$$\mathbf{\Sigma}_\mu = \text{Cov}(\delta\mu, \delta\mu) \approx (g'(0))^2 \ \text{Cov}(\kappa(\mathbf{x}), \kappa(\mathbf{x})) = (g'(0))^2 \ \mathbf{\Sigma}_\kappa \tag{22}$$

Thus, the propagated uncertainty in μ can be defined in terms of the latent variable κ and the sensitivity $g'(0)$, which depends on the chosen marginal distribution function. Finally, $\mathbf{\Sigma}_\mu$ can be substituted into Eq. (12), along with the previously obtained system matrix, and be used to evaluate the output covariance $\mathbf{\Sigma}_u$.

Furthermore, once the response is obtained, the goodness of the approximation can be assessed by comparing it with MC sampling from the probability $p(\mathbf{u})$ defined in Eq. (10). In each run, a single realization from $\kappa(\mathbf{x})$ in Eq. (16) is drawn, and the procedure defined in Section 3.2 is employed to obtain samples from the desired marginal space, which are then used to assemble the system matrix.

4 APPLICATION CASE

In this section, the proposed Copula-based statFEM is applied to a one-dimensional Poisson problem, where the domain is discretized into 32 elements using linear Lagrange shape functions. In Section 4.1, the governing equations and stochastic variables are defined, and in Section 4.2 the employment of the copula process to the problem at hand is discussed. The results are presented in Section 4.3 and discussed in Section 4.4.

4.1 Case definition

The one-dimensional Poisson-Dirichlet problem is defined based on Eq. (1):

$$\begin{aligned} -\frac{d}{dx}\left(\mu(x)\frac{du}{dx}\right) &= f(x), \quad \text{in } \Omega = (0,1) \\ u(x) &= 0, \qquad \text{on } x = 0 \text{ and } x = 1 \end{aligned} \tag{23}$$

where the diffusion coefficient $\mu(x)$ is random, and the forcing term $f(x) = 1$ is deterministic. The solution $u(x)$ is evaluated using the copula-based statFEM approach, while the quality of the approximations is assessed by comparing with MC sampling results and employing the Wasserstein 2-distance (W_2) as a dissimilarity metric. This is particularly advantageous when dealing with MC samples as it can be directly applied to point clouds, not requiring the empirical distribution to be functionally described [1].

4.2 Probabilistic FE analysis

Since the proposed approach is supposed to deal with arbitrary marginal distributions, two different ones are used to represent the random diffusion coefficient. In both cases, however, μ is assumed to have a known expected value ($\overline{\mu} = \mathbb{E}[\mu(x)] = 1$) and bounds ($\pm\, 0.2\,\overline{\mu}$). For a diffusion coefficient that follows a uniform distribution:

$$\mu(x) \sim \mathcal{U}(0.8, 1.2) \implies g(\kappa) = \mu_{min} + (\mu_{max} - \mu_{\min})\,\Phi(\kappa) \tag{24}$$

The evaluation of the approximated mean and covariance from Eq. (12) requires the definition of:

$$\begin{aligned} \overline{\mu} &= \mathbb{E}[\mu(x)] = 1 \\ \mathbf{\Sigma}_\mu &\approx \left(\frac{\mu_{max} - \mu_{min}}{\sqrt{2\pi}}\right)^2 \mathbf{\Sigma}_\kappa \end{aligned} \tag{25}$$

where both μ_{min}, μ_{max} refer to the bounds. Alternatively, for the case it follows a beta distribution:

$$\begin{aligned} &X \sim \text{Beta}(5,5) \\ &\mu(x) = 0.8 + (1.2 - 0.8)X \implies g(\kappa) = \mu_{min} + (\mu_{max} - \mu_{\min}) F^{-1}_{Beta}(\Phi(\kappa); 5, 5) \end{aligned} \tag{26}$$

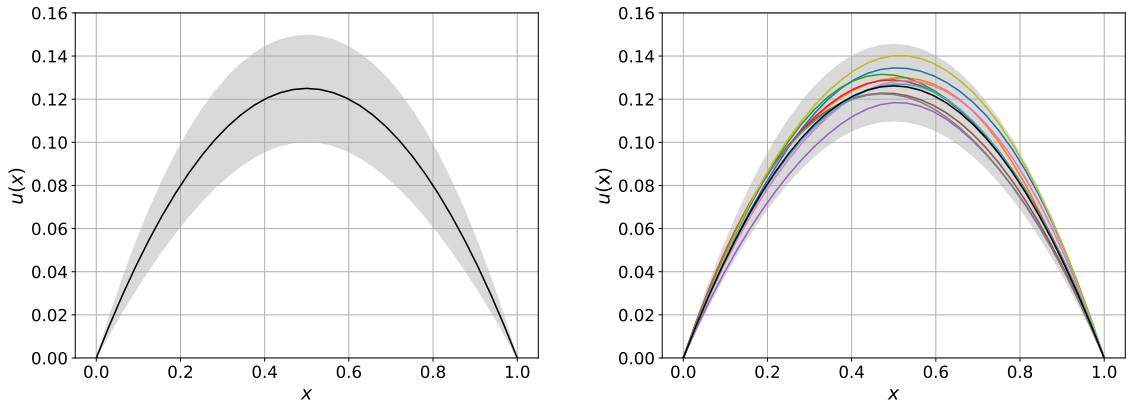
where $F^{-1}_{Beta}(\cdot; \alpha, \beta)$ is the inverse beta CDF. Thus:

$$\begin{aligned} \overline{\mu} &= \mathbb{E}[\mu(x)] = 1 \\ \mathbf{\Sigma}_\mu &\approx \left(\frac{\mu_{max} - \mu_{min}}{\sqrt{2\pi}} \frac{1}{f_{Beta}\left(F^{-1}_{Beta}(0.5; 5, 5)\right)}\right)^2 \mathbf{\Sigma}_\kappa \end{aligned} \tag{27}$$

in which f_{Beta} is the beta probability distribution function. Finally, to completely define the stochastic variables in the proposed model, the kernel function of the latent variable $\kappa(x)$ can be modeled solely in terms of the length scale, as previously defined. In both cases, $\mathbf{\Sigma}_\kappa$ is computed for $l_\kappa = 0.2$.

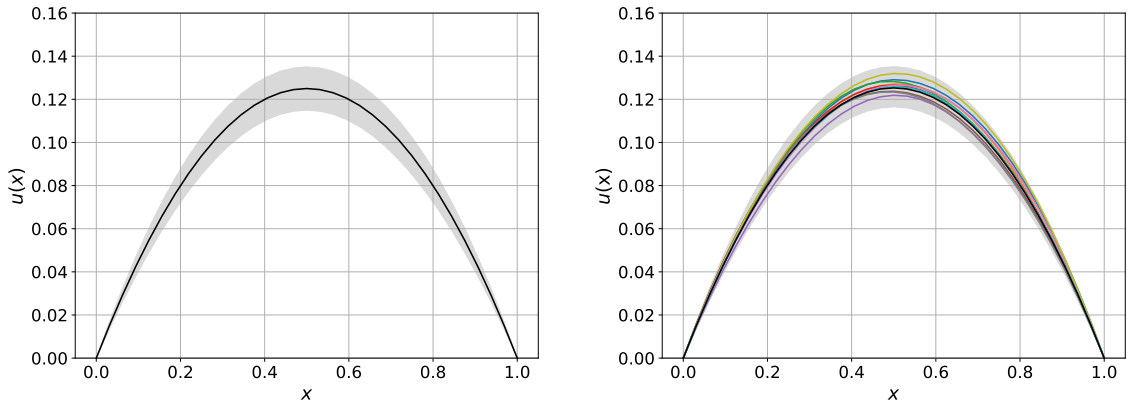
4.3 Results

Since it is now possible to evaluate second-order statistics, i.e., mean and covariance, of the approximated solution field $p(\mathbf{u}) = \mathcal{N}(\overline{\mathbf{u}}, \mathbf{\Sigma}_u)$, and also to sample from the input distribution for the MC solution, the methodology described in Section 3.3 can be employed. The displacements for the uniform and beta distributed diffusion coefficients are illustrated, respectively, in Figs. 4 and 5. Then, the moment-matched distribution is sampled and the Wasserstein 2-distances are shown in Table 1 to indicate the dissimilarity between the results and the MC samples in both cases. The colored lines are realizations of $u(x, \mu)$ from samples drawn from Eq. (5), and the shaded areas are the 95% confidence regions for each case.



(a) StatFEM solution. (b) MC sampling solution.

Figure 4: Comparison between (a) StatFEM and (b) MC sampling for uniformly distributed μ. The shaded areas depict the 95% confidence regions.



(a) StatFEM solution. (b) MC sampling solution.

Figure 5: Comparison between (a) StatFEM and (b) MC sampling for beta distributed μ. The shaded areas depict the 95% confidence regions.

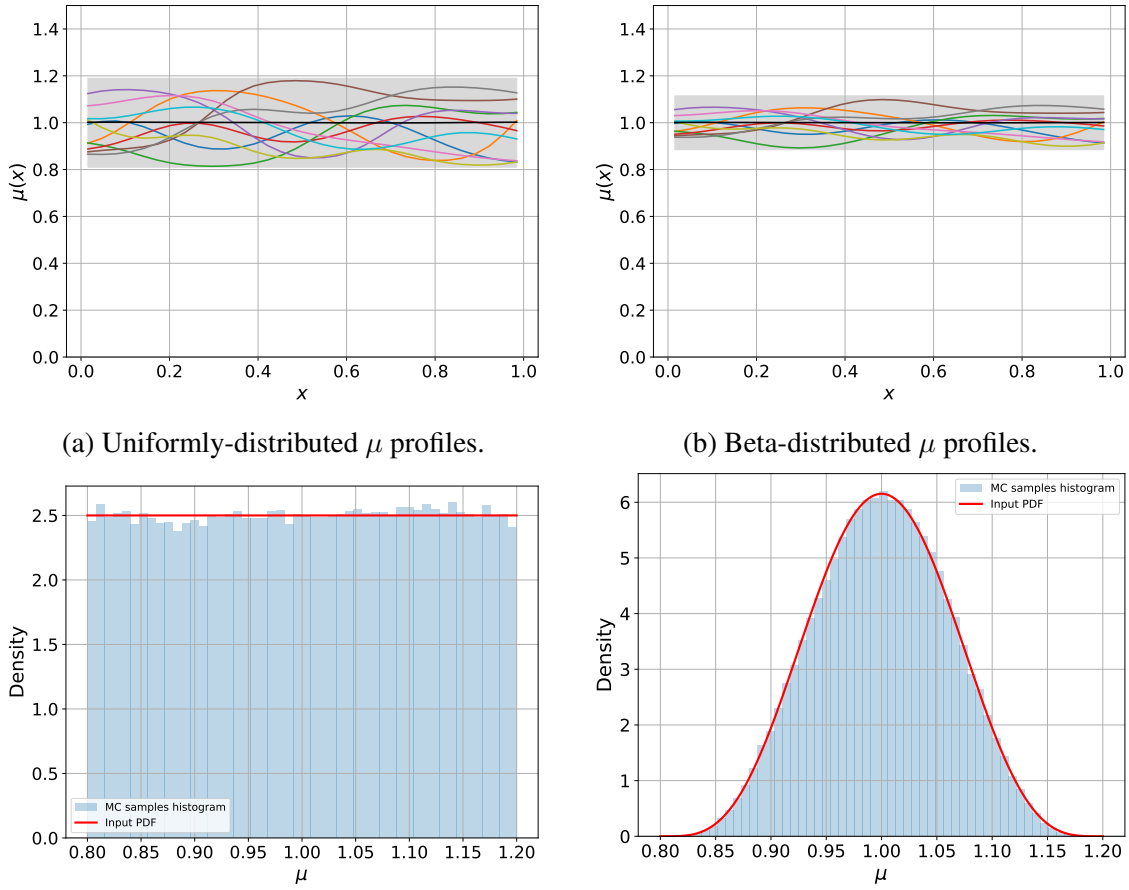
Furthermore, the diffusion coefficient profiles along the domain for the highlighted samples are shown in Fig. 6. The shaded areas are related to the aforementioned confidence interval.

4.4 Discussion

Observing both Figs. 4 and 5, the mean, due to the same expected values in both distributions, is in excellent agreement with reference results. However, the uniformly distributed

$\mu(x)$	W_2
Uniform	0.069
Beta	0.031

Table 1: Wasserstein 2-distances for different input distributions.



(a) Uniformly-distributed μ profiles.

(b) Beta-distributed μ profiles.

(c) Uniformly-distributed μ histogram.

(d) Beta-distributed μ histogram.

Figure 6: Characterization of μ samples with different marginal distributions.

diffusion coefficient propagates a higher uncertainty to the output than the beta distributed one. This behavior is expected, since the uniform probability density is higher for values farther from the mean than the beta one, resulting in more scattered realizations of μ, as seen in Fig. 6.

Moreover, the approximation in both cases results in higher uncertainty than the reference ($\boldsymbol{\Sigma}_{\mathbf{u}} > \boldsymbol{\Sigma}_{\mathrm{MC}}$). This difference is noticeably higher in the uniformly distributed case, while presenting a better agreement for the beta one. The goodness of these approximations is measured in Table 1, which shows that the Beta's W_2 is much lower. This is expected, since the beta distribution is symmetric and bell-shaped, resembling well the assumed Gaussian behavior in the output. However, it is valid to remark that the provided linear approximation for the Uniform marginal can still be used, since the reasonability of the W_2 distance absolute value is context-dependent.

5 CONCLUSIONS

In this work, a Gaussian Copula Process approach is used to generate random physical parameters with arbitrary marginal distributions and a latent, usually desired, Gaussian spatial dependence. It is embedded into the original statFEM framework to have the uncertainty propagated, being seamlessly integrated into numerical simulations. This statistical approach is assessed by solving a probabilistic forward problem for beta and uniformly distributed diffusion coefficients of a 1D Poisson toy problem, with the obtained approximations being compared to MC sampling solutions.

Although the developed copula-based statFEM approach allows for the use of arbitrary marginal distributions, the quality of the resulting Gaussian approximation of the non-Gaussian statFEM marginal over the FE solution is observed to be dependent on the input distribution. Due to the higher resemblance with a Gaussian behavior, the results for the beta distributed μ present a much better agreement with the MC sampling solution. On the other hand, although the approximation for the Uniform marginal case can still be considered reasonable, it is of substantially lower quality. Furthermore, it is important to remark that the majority of this difference is due to the approximated covariance, which was, in both cases, greater than the one obtained from MC sampling. This would, however, be desirable e.g., in the context of structural design, acting as a safety factor regarding the propagated uncertainty while still providing close approximations.

Therefore, this work highlights the suitability of the proposed copula-based statFEM framework to solve probabilistic forward FE problems for non-Gaussian distributed diffusion coefficients, as the presented results demonstrate a good agreement with reference values for different input distributions. The goodness of the linear approximations, however, was shown to be dependent on the marginal distribution of μ. Thus, an improvement on the approximation method to further enhance the framework seems feasible. Additionally, since this work focuses only on the forward problem with smoothly varying random variables, the demonstration of the copula-based statFEM in inverse problems is left for future studies. Other possible extensions are the GP characterization by a Gaussian Copula-based Bayesian Network and the consideration of variables with non-Gaussian dependence.

Acknowledgements

The authors gratefully acknowledge the European Commission for its support of the Marie Sklodowska Curie program through the Horizon Europe DN APRIORI project (GA 101073551). Views and opinions expressed are, however, those of the authors only and do not necessarily reflect those of the European Union. The European Union cannot be held responsible for them.

REFERENCES

[1] L. Riccius, I. B. C. M. Rocha, J. Bierkens, H. Kekkonen, F. P. van der Meer, Integration of Active Learning and MCMC Sampling for Efficient Bayesian Calibration of Mechanical Properties, *arXiv:2411.13361*, 2024.

[2] B. Barros, B. Conde, B. Riveiro, O. Morales-Nápoles, Gaussian Copula-based Bayesian network approach for characterizing spatial variability in aging steel bridges, *Structural Safety*, **106**, 102403, 2024.

[3] M. Girolami, E. Febrianto, G. Yin, F. Cirak, The statistical finite element method (statFEM) for coherent synthesis of observation data and model predictions, *Computer Methods in Applied Mechanics and Engineering*, **375**, 113533, 2021.

[4] E. Febrianto, L. Butler, M. Girolami, F. Cirak, Digital twinning of self-sensing structures using the statistical finite element method, *Data-Centric Engineering*, **3**, e31, 2022.

[5] A. Glyn-Davies, C. Duffin, I. Kazlauskaite, M. Girolami, Statistical Finite Elements via Interacting Particle Langevin Dynamics, *arXiv:2409.07101*, 2024.

[6] C. Czado, *Analyzing dependent data with vine copulas: A practical guide with R*, Springer, 2019.

[7] A. Sklar, Fonctions de repartition a n dimensions et leurs marges, *Publ. Inst. Statist. Univ. Paris*, **8**, 229–231, 1959.

[8] R. B. Nelsen, *An introduction to Copulas*, Springer Series in Statistics, 2006.

[9] A. G. Wilson, Z. Ghahramani, Copula Processes, *Advances in Neural Information Processing Systems*, **23**, 2010.

[10] W. K. Liu, T. Belytschko, A. Mani, Random field finite elements, *International Journal for Numerical Methods in Engineering*, **23**, 1831–1845, 1986.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

POSSIBILISTIC NEURAL NETWORKS: RELIABLE CONFIDENCE PREDICTIONS UNDER LIMITED DATA

Jan Schneider[1], **Tom Könecke**[1], **and Michael Hanss**[1]

[1]Institute of Engineering and Computational Mechanics
Pfaffenwaldring 9, 70569 Stuttgart
e-mail: {jan.schneider, tom.koenecke, michael.hanss}@itm.uni-stuttgart.de

Abstract. *Neural networks excel at capturing complex data relationships across diverse fields, yet they often lack the ability to quantify prediction uncertainties—a critical requirement for applications like medical diagnosis and autonomous driving. While existing solutions like Bayesian neural networks and interval-based methods address this issue, they typically demand substantial computational resources, large data sets, and often rely on stochastic assumptions that may not suit scenarios with both aleatoric and epistemic uncertainties. This paper introduces possibilistic neural networks (PNNs), a novel approach for providing reliable confidence predictions with limited data. Based on possibility theory, PNNs offer more flexible uncertainty modeling than traditional probabilistic methods while maintaining stochastic guarantees. The method employs a simple architecture of fully connected layers and introduces a novel three-part loss function to ensure nested confidence intervals of decreasing width. The network is trained to predict a plausibility function, which is then transformed into a possibility function using either approximate or reliable transformations. The effectiveness of the proposed approach is demonstrated through experiments with both Gaussian and non-Gaussian noise scenarios, using only* 80 *training samples. Results show that PNNs can successfully generate valid nested confidence intervals without requiring prior knowledge of the underlying distributions. The reliable transformation extends the method to provide guaranteed confidence predictions even with limited data sets, making it particularly suitable for real-world applications where data scarcity and uncertainty quantification are crucial concerns.*

Keywords: Possibility Theory, Neural Networks, Limited Data, Confidence Prediction, Uncertainty Quantification.

1 INTRODUCTION

Machine learning, and particularly neural networks, have revolutionized numerous fields by providing powerful tools for modeling complex relationships in data. From medical diagnosis to autonomous systems, these methods have demonstrated remarkable capabilities in pattern recognition and prediction tasks. However, as these applications increasingly impact critical decision-making processes, the need for reliable uncertainty quantification has become an area of major interest. Traditional neural network approaches focus primarily on minimizing prediction errors, resulting in point estimates that may appear deceptively precise. This limitation becomes particularly problematic in safety-critical applications, where understanding the confidence level of a prediction is as crucial as the prediction itself. Furthermore, real-world applications often face the dual challenge of limited training data and the presence of both aleatoric and epistemic uncertainties.

There are various approaches to capture uncertainty in machine learning tasks and neural networks using interval arithmetic, Bayesian neural networks, or robust training based on Lipschitz bounds [10, 8, 9, 3]. These neural networks can be trained on noisy data and capture underlying relations to outputs which enables quantified statements in the prediction step. However, these methods often require prior assumptions or information on the underlying noise distribution, prior guesses on appropriate basis functions or they suffer from a complex network structure.

In contrast, we propose a novel approach for neural network training with uncertain data based on possibility theory. Possibility theory, as a theory of imprecise probabilities, grants a flexible tool to describe uncertainty without the need of prior assumptions. This allows to model aleatoric and epistemic uncertainty sufficiently while still being able to provide stochastical guarantees. The proposed method, called Possibilistic Neural Networks (PNNs), is designed to provide reliable confidence predictions for regression tasks with limited data. The network is trained to predict a plausibility function which is then transformed into a possibility function by the use of either either approximate or reliable transformations. The main contribution of this paper is the derivation of a suitable cost function for the neural network training. The effectiveness of the proposed approach is demonstrated in scenarios with both Gaussian and non-Gaussian noise, using only a limited number of training samples. The experiments demonstrate the method's ability to construct valid nested confidence intervals, even without knowing the distributions of the input data beforehand. By incorporating a reliable transformation technique, this approach remains effective even when working with small data sets. This makes PNNs especially valuable for practical applications where both limited data availability and the need to accurately measure uncertainty pose significant challenges.

The remainder of this paper is organized as follows: Section 2 provides the theoretical background of possibility theory and its application as a tool for uncertainty quantification. Section 3 introduces the architecture and training methodology of PNNs and states the loss function as well as the analytical gradients for the network training. Section 4 presents experimental results on both Gaussian and non-Gaussian noise scenarios, demonstrating the method's effectiveness. Finally, Section 5 discusses the implications and potential applications of this approach, along with directions for future research.

2 POSSIBILITY THEORY

In the following paragraph, a brief introduction to possibility theory is given that only provides the essential background for understanding the proposed neural-network learning scheme and the idea of confidence predictions. This introduction does not aim to be exhaustive and for a more detailed discussion the reader is referred to [5, 7].

Consider a classical random experiment with the sample space Ω and its associated σ-algebra Σ. For an event $U \in \Sigma$ with counter event $\neg U$, the so-called elementary possibility function is any measurable function $\pi : \Sigma \to [0, 1]$ satisfying

$$\sup_{\omega \in \Omega} \pi(\omega) = 1. \tag{1}$$

Subsequently, this allows to introduce the possibility measure

$$\Pi(U) = \sup_{\omega \in U} \pi(\omega) \; \forall \; U \in \Sigma, \tag{2}$$

which is induced by the elementary possibility function and satisfies the following properties:

$$\Pi(\emptyset) = 0, \quad \Pi(\Omega) = 1, \quad \Pi(U \cup V) = \max\{\Pi(U), \Pi(V)\}. \tag{3}$$

There exist different characteristics of an elementary possibility function to quantify the information with respect to a probabilistic statement. One key concept is consistency [2, 1], which states that a probability measure P is consistent with an elementary possibility function if the probability $\mathrm{P}(U)$ is upper-bounded by the induced possibility measure $\Pi(U)$ for all events $U \in \Sigma$. This concept forms the foundation for understanding the later proposed statement relating possibility functions to probabilities and, thus, to confidence predictions. An essential concluding property arises in the context of strict superlevel sets of the elementary possibility function, given by

$$C_\pi^\alpha = \{\omega \in \Omega \mid \pi(\omega) > \alpha\}. \tag{4}$$

These sets are also called confidence sets and specify regions in the sample space where the possibility measure exceeds a given threshold $\alpha \in [0, 1]$.

Subsequently, when an elementary possibility function is consistent with a probability measure, the confidence sets can be used to derive the probabilistic guarantee from the possibility function via

$$P(C_\pi^\alpha) \geq 1 - \alpha. \tag{5}$$

In the context of inferential models [4], i.e., when the distribution provides predictive information about an unknown state, this property has to be assessed and the corresponding model is called valid if it holds for the whole sample space. This formulation is the basis for the later proposed neural-network learning scheme, which aims to predict confidence sets for input instances. The assessment of validity provides a more intuitive understanding of possibility functions and facilitates comparisons between them. In fact, there are infinite possibility functions consistent with a probability measure. This motivates the introduction of specificity, which states that a possibility function π_1 is more specific than another possibility function π_2 if $\pi_1(\omega) \leq \pi_2(\omega)$ for all $\omega \in \Omega$. Using the interpretation of Eq. (5), it is trivial to conclude that a more specific possibility function yields a more informative result since the confidence sets are tighter, i.e. $C_{\pi_1}^\alpha \subseteq C_{\pi_2}^\alpha$ for all $\alpha \in [0, 1]$.

These concepts and characteristics extend to quantitative statements about imprecise probabilities through the introduction of a subjective plausibility function $\rho : \Sigma \to [0, 1]$, which quantifies event plausibility based on available information. While ρ is not subjected to the constraints of a possibility function and, technically, does not have to map on the same metric, it must establish a relative ordering of event plausibility. Therefore, given a family $\mathcal{P}$ of probability measures, an appropriate plausibility function can be used to define the plausibility-to-possibility- or IP-Π-Transform [5]

$$\pi(\omega) = \sup_{P \in \mathcal{P}} P\{\xi \in \Omega : \rho(\xi) \leq \rho(\omega)\} \quad \forall\, \omega \in \Omega. \tag{6}$$

To this end, one of the crucial challenges is to identify an appropriate plausibility function. While existing approaches for data-driven applications rely on prior knowledge of the data-generating process, we propose a neural-network-based approach to learn plausibility functions directly from data for predicting instance-specific confidence sets. Furthermore, the vaildity of the neural-network-based predictions is of utmost importance to guarantee reliable predictions on unseen data.

3 NEURAL NETWORKS AS PLAUSIBILITY MEASURES

Several approaches have been proposed to capture uncertainty in training data for reliable predictions of neural networks [10, 8]. However, these methods often suffer from complex structures or require prior assumptions. The approach proposed in this paper is a quantitative learning scheme using a classical neural network that predicts plausibility

functions, which are then used to derive confidence sets and, subsequently, possibility functions.

The general neural network architecture is intentionally kept simple. It consists of an input layer with one neuron per input dimension, followed by a series of l hidden layers, each containing k_l neurons. Each hidden neuron $h^j_{k_j}$ is fully connected to either the successor layer or the output layer. Finally, the output layer contains $2n$ neurons, with corresponding outputs $\boldsymbol{y} = y_1, \ldots, y_{2n}$, where $n \in \mathbb{N}$ represents the desired number of confidence intervals. Each neuron is structured classically as a linear combination of weighted inputs and bias terms, which are fed to a specified activation function. The network is trained on N pairs of data points, consisting of input instances $x \in \mathbb{R}^d$ and noisy output values $\hat{y} \in \mathbb{R}$, which can be interpreted as realizations of a random process. The general structure is visualized in Figure 1. The key idea lies in the interpretation of

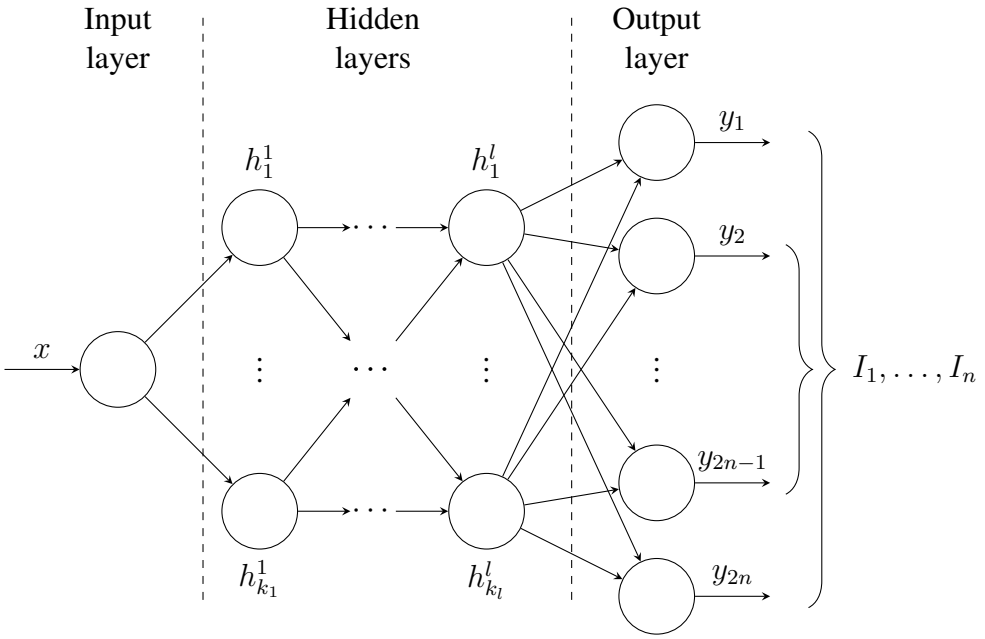


Figure 1: General structure of the neural network with interval outputs.

the outputs: every i th and $(2n - i + 1)$ th neuron in the output layer define one prediction interval I_i. The challenge consists in developing an appropriate loss function that ensures these intervals to provide a nested plausibility ordering while enabling maximal specificity and validity of the transformed possibility functions.

For the proposed method, consider a three-part loss

$$L(\boldsymbol{y}, \hat{y}) = \boldsymbol{\lambda} \begin{bmatrix} L_{\mathrm{w}}(\boldsymbol{y}) & L_{\mathrm{o}}(\boldsymbol{y}) & L_{\mathrm{n}}(\boldsymbol{y}, \hat{y}) \end{bmatrix}^\top, \tag{7}$$

where $L_\mathrm{w}, L_\mathrm{o}, L_\mathrm{n}$ are individual loss terms for different optimization goals weighted by respective hyperparameters $\boldsymbol{\lambda} = [\lambda_\mathrm{w}, \lambda_\mathrm{o}, \lambda_\mathrm{n}]$.

The first term L_w penalizes the width of the prediction interval. As stated earlier, the goal is to be as specific as possible since boundaries over the whole state space would still be valid but contain no information. Since confidence intervals with high α-values should be smaller than lower ones, the partial loss of the corresponding interval $I_i = [y_i, y_{2n-i+1}]$ is scaled with an interval dependent factor. The complete loss term is given by

$$L_\mathrm{w}(\boldsymbol{y}) = \sum_{i=1}^{n} \left(\frac{i}{2n} + 1 \right) |y_{2n-i+1} - y_i|. \tag{8}$$

For backpropagation, the gradient of this partial loss function has to be calculated with respect to each lower interval bound y_i and upper interval bound y_{2n-i+1} independently. The gradients for the successor layers are then calculated by the chain rule. Analytically, this can be derived as

$$\frac{\delta L_\mathrm{w}}{\delta y_i} = \left(\frac{i}{2n} + 1 \right) \operatorname{sgn}(y_{2n-i+1} - y_i), \quad \frac{\delta L_\mathrm{w}}{\delta y_{2n-i+1}} = -\frac{\delta L_\mathrm{w}}{\delta y_i}. \tag{9}$$

Furthermore, the second term L_o is responsible for ordering the intervals. It is trivial to see that the inequality $y_i \leq y_{2n-i+1}$ has to hold for all I_i with $i \in \{1, \ldots, n\}$. Besides, it is important to ensure a hierarchical ordering between the intervals, i.e., the lower bound of the $(i+1)$ th interval has to be greater than (or equal to) the upper bound of the i th interval. These relations can be enforced by defining the second loss term as

$$L_\mathrm{o}(\boldsymbol{y}) = \frac{1}{2n} \sum_{j=1}^{2n-1} \max(0, y_{j+1} - y_j). \tag{10}$$

Note that it is sufficient to enforce $y_{j+1} \leq y_j$ for all j since this implies $y_{j+2} \leq y_j$. Analogously to the width loss, the gradient of this partial loss function with respect to the outputs has to be calculated independently for each node with

$$\frac{\delta L_\mathrm{o}}{\delta y_1} = -\frac{\mathbb{1}_{\mathbb{R}^+}(y_2 - y_1)}{2n}, \quad \frac{\delta L_\mathrm{o}}{\delta y_{2n}} = \frac{\mathbb{1}_{\mathbb{R}^+}(y_{2n} - y_{2n-1})}{2n}, \tag{11}$$

$$\frac{\delta L_\mathrm{o}}{\delta y_j} = \frac{-\mathbb{1}_{\mathbb{R}^+}(y_{j+1} - y_j) + \mathbb{1}_{\mathbb{R}^+}(y_j - y_{j-1})}{2n}, \quad j \in \{2, \ldots, 2n-1\}, \tag{12}$$

where $\mathbb{1}_{\mathbb{R}^+}$ is the indicator function for positive real numbers.

Lastly, the third term L_n handles prediction errors by penalizing intervals that do not contain the true value $\hat{y}$. When the true value falls outside an interval, this loss term encourages two corrective actions: widening the interval and shifting its center towards $\hat{y}$.

However, not all intervals need to provide the same level of confidence. For instance, a 99% confidence interval should almost always contain the true value, while a 50% confidence interval can reasonably exclude it about half the time.

This motivates the introduction of the loss term

$$L_{\mathrm{n}}(\boldsymbol{y}, \hat{y}) = \frac{1}{2N} \sum_{i=1}^{n} \left(\frac{2n}{i} \right)^2 \left([\max(0, \hat{y} - y_i)]^2 + [\max(0, y_{2n-i+1} - \hat{y})]^2 \right), \tag{13}$$

where the hierarchical confidence structure is implemented through the quadratic scaling factor $((2n)/i)^2$. For intervals meant to provide high confidence (small i relative to n), this factor creates a large penalty when the true value falls outside the interval. Conversely, for intervals meant to provide lower confidence (large i relative to n), the penalty is reduced, allowing these intervals to remain narrower even if they occasionally exclude the true value.

The gradient for the currently considered interval can again be derived independently as

$$\frac{\delta L_{\mathrm{n}}}{\delta y_i} = -\frac{1}{N} \left(\frac{2n}{i} \right)^2 \max(0, \hat{y} - y_i), \tag{14}$$

$$\frac{\delta L_{\mathrm{n}}}{\delta y_{2n-i+1}} = \frac{1}{N} \left(\frac{2n}{i} \right)^2 \max(0, y_{2n-i+1} - \hat{y}). \tag{15}$$

Finally, the total gradient

$$\frac{\delta L}{\delta y_i} = \lambda_{\mathrm{w}} \frac{\delta L_{\mathrm{w}}}{\delta y_i} + \lambda_{\mathrm{o}} \frac{\delta L_{\mathrm{o}}}{\delta y_i} + \lambda_{\mathrm{n}} \frac{\delta L_{\mathrm{n}}}{\delta y_i} \tag{16}$$

can be calculated for each output neuron allowing to determine the gradients of each hidden neuron accordingly. After training, the outputs of the network define a plausibility function $\rho_Y(y)$. For n total confidence intervals, the plausibility of each interval is defined by $\rho_{Y,I_i} = i/n$ with $i \in \{1, \ldots, n\}$.

To obtain a possibility function consistent with the plausibility function, the plausibility-to-possibility transform must be applied. However, since only limited data are available for training and transformation, different approaches can be considered. The first approach is the approximate transformation [5], which defines the unbiased estimator

$$\bar{\pi}_Y(y) = \frac{K(y)}{N}, \quad K(y) = |\{Y_k : \rho_Y(Y_k) \leq \rho_Y(y); k = 1, \ldots, N\}|, \quad \forall y \in Y, \tag{17}$$

where $K(y)$ counts the number of samples with plausibility less than or equal to $\rho_Y(y)$. As shown in [6], this counting mechanism provides a sufficient statistic for the transformation. However, while this approach maintains validity on the training data set,

it cannot provide guaranteed coverage for unseen data. Hence, as stated in [6], a reliable transformation can be applied using the inverse of the beta cumulative distribution function beta^{-1}, resulting in

$$\hat{\pi}_Y(y) = \text{beta}^{-1}(\gamma, K(y)+1, N-K(y)), \quad \forall y \in Y, \tag{18}$$

where the reliability level $\gamma \in [0,1]$ states that $\hat{\pi}_Y(y)$ is point-wise less specific than $\pi_Y(y)$ with a probability of

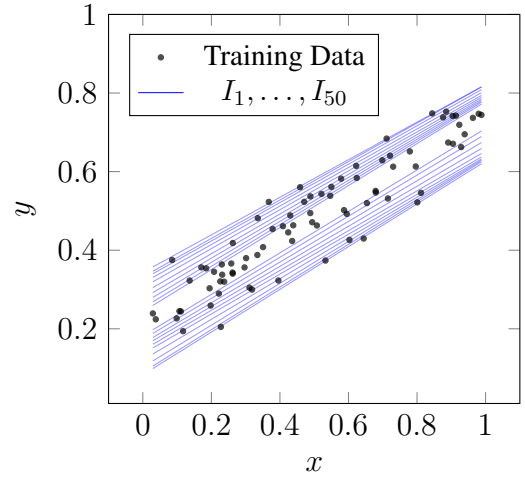
$$P\left(\hat{\pi}_Y(y) \geq \pi_Y(y)\right) \geq \gamma. \tag{19}$$

With the reliable extension, it is possible to loosen the specificity learned by the network on the training data set and generalize the prediction to unseen data while still being able to give guarantees on the predictions. The γ-level defines a trade-off parameter between specificity and conservatism and can be tuned to the desired level of reliability.

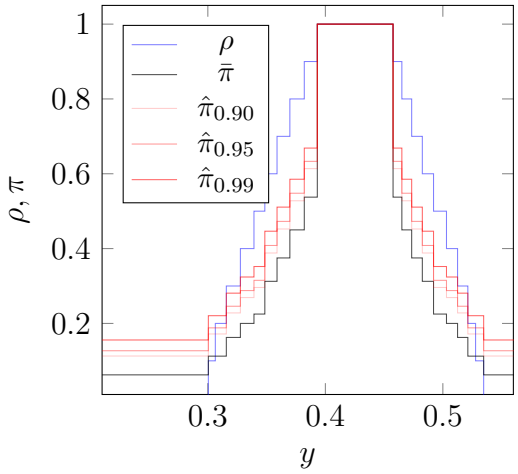
4 RESULTS

To verify the proposed method, two neural networks are trained for different test cases. As a first example, consider a data set drawn from a linear function $\hat{y} = 0.2x+1+w$ with a Gaussian distributed noise term $w \sim \mathcal{N}(0, 0.1)$ to simulate aleatory measurement noise. The training set comprises $N = 80$ randomly sampled values of $x \in [0,5]$ and the corresponding noisy measurements $\hat{y}$ which are normalized to the interval $[0,1]$. Both networks share the same architecture of two hidden layers with $k_1 = 2$ and $k_2 = 6$ neurons using the ReLu activation function, but differ in their output layers: the first network has $m = 20$ neurons to predict $n = 10$ confidence intervals, while the second has $m = 100$ outputs to predict $n = 50$ intervals. In their output layers, both networks employ linear activation functions and undergo training for 25.000 epochs with a learning rate of 0.0008. Since the networks performance is highly sensitive to hyperparameter selection, a grid-based optimization procedure is employed, yielding optimal hyperparameters $\boldsymbol{\lambda} = [0.0005, 0.001, 1.5]$ for the first network and $\boldsymbol{\lambda} = [0.001, 0.003, 1.5]$ for the second network. As shown in Figure 2(a) and Figure 3(a), the networks are able to produce confidence intervals which are nested, have a decreasing width and bound the data set tightly. In order to achieve a possibilistically valid result, $K(y)$ is determined through a post-processing analysis that quantifies the number of training data points contained within each interval. This allows to derive the approximate and reliable possibility function, which use the same interval boundaries but with respective possibilistic values. The differences can be observed when considering the prediction on a specific x-value. The results for the approximate and reliable transformation with different γ-levels at $x = 0.4$ are shown in Figure 2(b) and Figure 3(b).

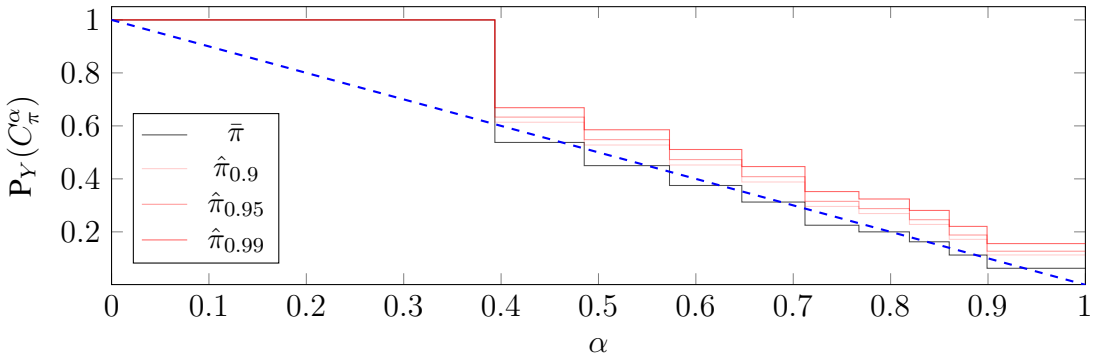
After verifying the nominal performance on the training data set, the next step is to analyze the network's predictive capabilities on unseen data and compare the reliable and approximate transformations. To do so, the consistency of confidence predictions



(a) Network prediction ρ.



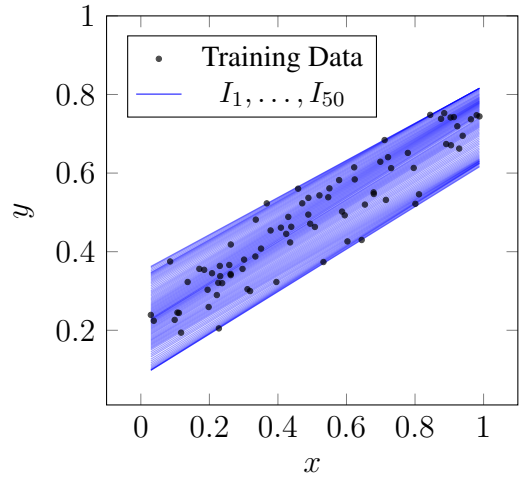
(b) Cut at $x = 0.4$.



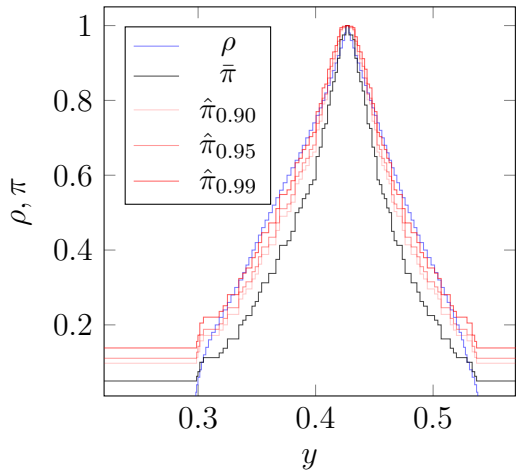
(c) Evaluation of $P_Y(C_\pi^\alpha)$.

Figure 2: Prediction and validity results of network with ten confidence intervals.

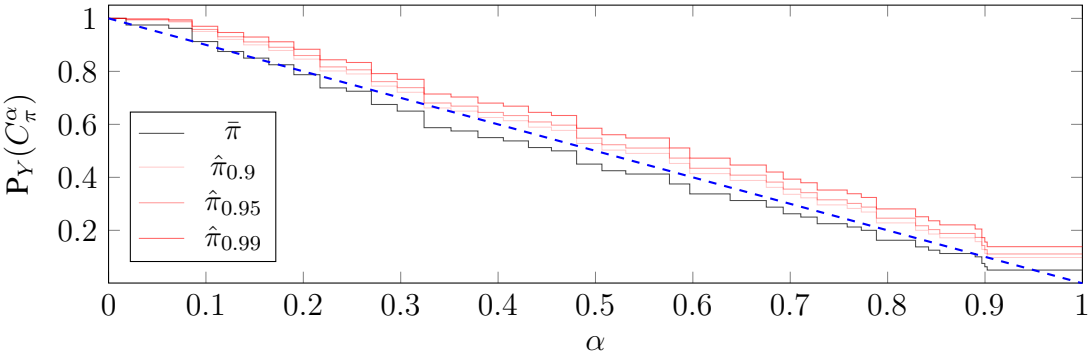
is evaluated on an unseen data set by checking Eq. (5) for different transformations. When assessing the number of correct predictions per interval, the corresponding graphs should remain above the $(1 - \alpha)$-line. This indicates that for any arbitrary α-level, the network is overly conservative but does not produce false predictions. The results for the same networks are shown in Figure 2(c) and Figure 3(c). As expected, the reliable transformation with increasing γ-levels is more conservative than the approximate transformation and allows to provide confident predictions for unseen data. Note that with an increasing number of intervals, the possibility functions start to converge to the result of the Optimal Transform [4] of a normal distribution, which implies that the network is able to capture the underlying relationships between the probabilistic and possibilistic space.



(a) Network prediction ρ.



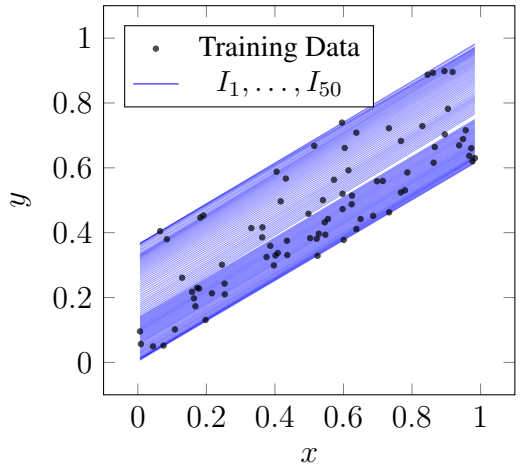
(b) Cut at $x = 0.4$.



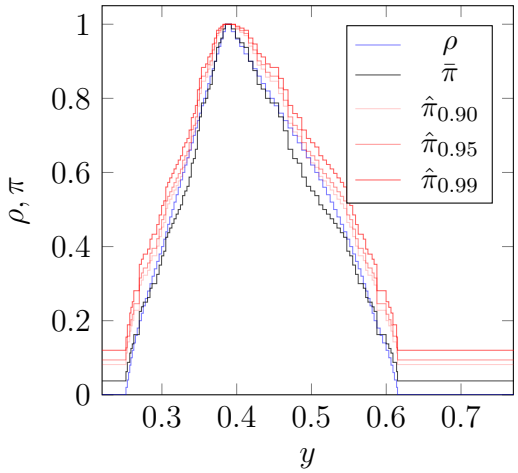
(c) Evaluation of $P_Y(C_\pi^\alpha)$.

Figure 3: Prediction and validity results of network with fifty confidence intervals.

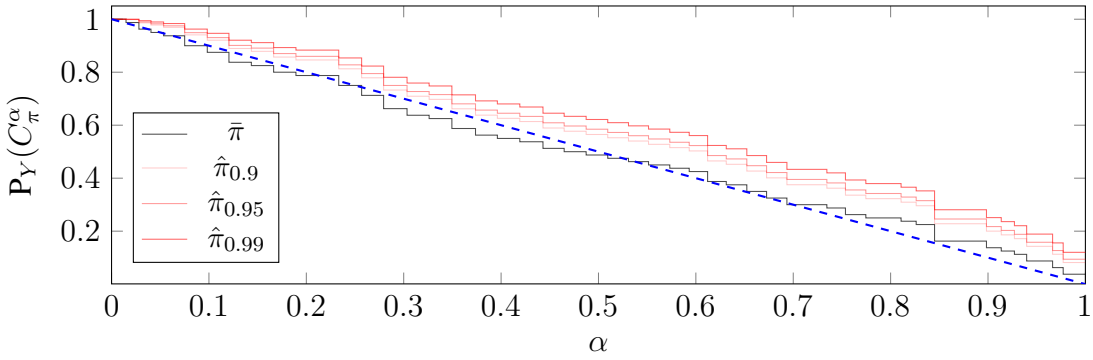
Furthermore, the great benefit of using this neural network structure under limited data is shown in a second example, where the network is trained on a data set which is influenced by two combined uniform noise terms $W_1 \sim \mathcal{U}(-0.2, 0)$ and $W_2 \sim \mathcal{U}(0, 0.4)$. The random variable w is drawn from W_1 for one half of the total samples and from W_2 for the remaining half. As in the previous example, $N = 80$ randomly sampled input values x with their corresponding noisy measurements $\hat{y}$ are used. The same network architecture is maintained, but the hyperparameters are now optimized to $\boldsymbol{\lambda} = [0.0021, 0.003, 1.5]$. The results for the network prediction and the reliable and approximate transformation are shown in Figure 4. The network demonstrates that comparable performance to the Gaussian noise case can be achieved. This shows that the proposed method is able to provide reliable confidence predictions under limited data and without any prior information about the underlying distribution.



(a) Network prediction ρ.



(b) Cut at $x = 0.4$.



(c) Evaluation of $P_Y(C_\pi^\alpha)$.

Figure 4: Prediction and validity results of network with fifty confidence intervals with uniform noise distribution.

5 CONCLUSION

In this work, a novel framework for uncertainty-aware neural-networks using possibility theory is introduced. The proposed methodology successfully quantifies uncertainty in training data by giving nested prediction intervals for possibilistic interpretations. While the main contribution is from the derivation of a suitable loss function for such specific desires, the trained output can be extended by using the approximate and reliable transformation principles found in possibility theory. Especially the latter is meant to give reliable confidence predictions on an unseen data set.

Empirical results demonstrate constant performance across diverse data sets, even for non-Gaussian noise distributions. It is especially remarkable that the network does not need any prior information or assumptions about the underlying distribution. While such characteristics show promise for applications in safety-critical environments like autonomous driving, challenges remain in capturing highly-nonlinear and/or multidimensional data distributions. Future research could be directed towards improving the convergence behavior and extending the proposed methodology, especially the sensitivity and optimization of the hyperparameters, which, so far, is the most challenging part for more complex data sets. One promising approach is to formalize different loss terms explicitly in the network architecture to reduce the sensitivity and number of hyperparameters while keeping the same guarantees. It is especially challenging to guarantee nested intervals over the whole domain rather then only on the training set, which could be resolved by this enhancement.

In general, possibilistic neural networks provide a promising alternative to traditional probabilistic methods, offering a principled approach to uncertainty representation that supports the development of more reliable AI systems.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support provided by the German Research Foundation (DFG) in the framework of the research project no. 514188140 (HA 2798/16-1). The authors would furthermore like to express their sincere appreciation of the work contributed by Niklas Abraham in the framework of his student thesis under supervision of the authors.

References

[1] Miguel Delgado and Serafın Moral. “On the Concept of Possibility-Probability Consistency”. In: *Fuzzy Sets and Systems* 21.3 (1987), pp. 311–318.

[2] Didier Dubois and Henri Prade. “When Upper Probabilities Are Possibility Measures”. In: *Fuzzy Sets and Systems* 49.1 (1992), pp. 65–74.

[3] David Heckerman. *A Tutorial on Learning With Bayesian Networks*. 2022. arXiv: 2002.00269 [cs].

[4] Dominik Hose. “Possibilistic Reasoning with Imprecise Probabilities: Statistical Inference and Dynamic Filtering”. In: *Schriften Aus Dem Institut Für Technische Und Numerische Mechanik Der Universität Stuttgart*. Dissertation, Band 74. Shaker Verlag, 2022.

[5] Dominik Hose and Michael Hanss. “A Universal Approach to Imprecise Probabilities in Possibility Theory”. In: *International Journal of Approximate Reasoning* 133 (2021), pp. 133–158.

[6] Dominik Hose and Michael Hanss. “On Data-Based Estimation of Possibility Distributions”. In: *Fuzzy Sets and Systems* 399 (2020), pp. 77–94.

[7] George Klir and Mark Wierman. *Uncertainty-Based Information: Elements of Generalized Information Theory*. Vol. 15. Springer Science & Business Media, 1999.

[8] Luis Oala et al. *Interval Neural Networks: Uncertainty Scores*. 2020. arXiv: 2003.11566.

[9] Patricia Pauli et al. “Training Robust Neural Networks Using Lipschitz Bounds”. In: *IEEE Control Systems Letters* 6 (2022), pp. 121–126.

[10] Krasymyr Tretiak, Georg Schollmeyer, and Scott Ferson. “Neural Network Model for Imprecise Regression with Interval Dependent Variables”. In: *Neural Networks* 161 (2023), pp. 550–564.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

BAYESIAN ROBUST DESIGN OPTIMIZATION: AN EXTREME VALUE APPROACH

Augustin Persoons[1], Pascal Lafon[1], and David Moens[2,3]

[1]University of Technology of Troyes
12 Rue marie Curie, Troyes 10010, France
e-mail: augustin.persoons@utt.fr

[2] KU Leuven, Department of Mechanical Engineering, LMSD Div.
Jan De Nayerlaan 5, 2860, Sint-Katelijne-Waver, Belgium

[3] Flandersmake@KULeuven

Abstract. *This work deals with the problem of robust design optimization under interval uncertainty. The proposed approach falls in the framework of Bayesian optimization techniques relying on Gaussian process regression models as a surrogate of the expensive performance function. The main difficulty resides in the problem at hand being two nested optimization problems. A direct, nested implementation of Bayesian optimization methods results in sub-optimal constraints in the selected calibration data points. The proposed method consists in approximating the stochasticity of the optimal of the inner problem using extreme value distributions. This approach allows us to consider an alternative stochastic process associated directly with the outer optimization problem and on which Bayesian optimization can be applied. While this approach does not circumvent the nested nature of the problem, the Bayesian optimization is limited to the outer problem and the generated calibration dataset is therefore not locked to any specific nested shape. The methods exhibits promising performances on the simple examples it has been tested on.*

Keywords: Bayesian optimization, Robust design optimization, Interval uncertainty, Extreme value distribution.

---

1 Introduction and statement of the problem

Consider an engineering design optimization problem involving a system which depends on two sets on variables. First a set of design parameters $x \in \mathbb{R}^{n_x}$ that can be chosen in a deterministic manner within a design space I_x . A second set of uncertain parameters $u \in \mathbb{R}^{n_u}$ that will take an arbitrary value in an interval I_u. The performance y of the system can be evaluated by the means of numerical model g such that $\forall (x, u) \in (I_x, I_u),\ y = g(x, u)$. Such a problem can for instance occur when limited data of the uncertainty variables is available and no distribution can be reasonably inferred from them.

This work focuses on design optimization in the context of an objective function formulated as a robustness metric. The notion of robustness, often credited to Taguchi [1], refers to a system's ability to maintain stable performance despite variations in operating conditions. Over time, various definitions of robustness metrics have been developed to suit different contexts, depending largely on the type of uncertainty being addressed. Some approaches rely on stochastic formulations, such as those based on statistical moments [2, 3, 4], while others adopt deterministic perspectives [5, 6, 7, 8]. The deterministic approach, which is particularly appropriate for handling interval uncertainties, is the one adopted in this study.

This work will focus on only one aspect of the robustness optimization problem, which is the minimization of the variation of performance. The second problem that will not be considered is the optimization of the nominal performance, in a similar fashion as deterministic design optimization. The first one is considered the most difficult and the proposed approach is a first step toward a weighted multi-objective optimization method.

The goal will therefore be to find the optimal design x^* minimizing the variation in performance propagated from the uncertain variables. The optimal design can then be defined as:

$$x^* = \underset{x \in I_x}{\operatorname{argmin}} \left(\max_{u \in I_u} g(x, u) - \min_{u \in I_u} g(x, u) \right) \tag{1}$$

A robustness indicator Δy can be conveniently defined as:

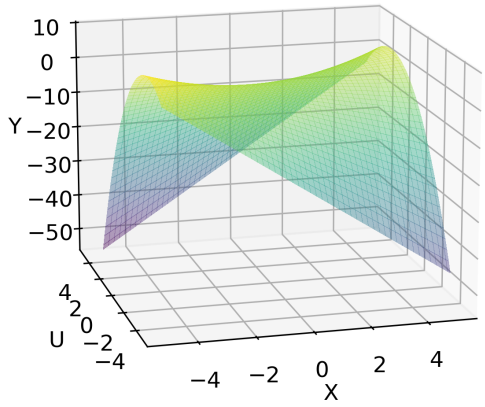
$$\forall x \in I_x,\ \Delta y(x) = \max_{u \in I_u} g(x, u) - \min_{u \in I_u} g(x, u) \tag{2}$$

The problem is illustrated on a saddleback-like 2-D function in Figure1. Figure1a illustrates the performance surface as a function of the design x and uncertain u variables. Figure1b illustrates the projection of the surface on the x space, highlighting the performance amplitude associated with each design as the vertical amplitude of the blue area. Figure1c finally illustrates the robustness indicator Δ_y associated with each design as well as the optimal design x^*.

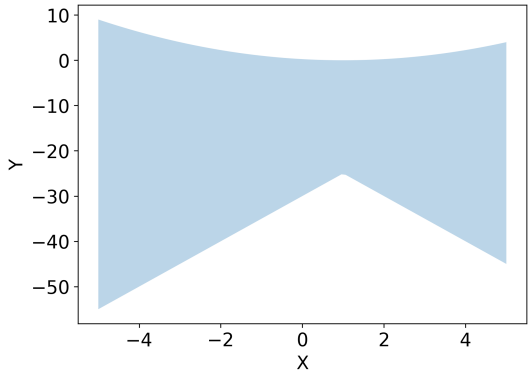
In this work, the Bayesian optimization framework is first briefly introduced and the difficulties in applying it to the problem at hand is highlighted. The heuristic approach based on extreme value distribution that is proposed in this work is then be detailed as well as the algorithm associated with it. The performance of the method is then illustrated on two analytical functions and a finite element model and the results are discussed. The paper finally concludes with a discussion, perspectives for future developments currently lacking and a link to the GitLab repository of the method is provided with a brief description of its content.

2 Bayesian optimization approach

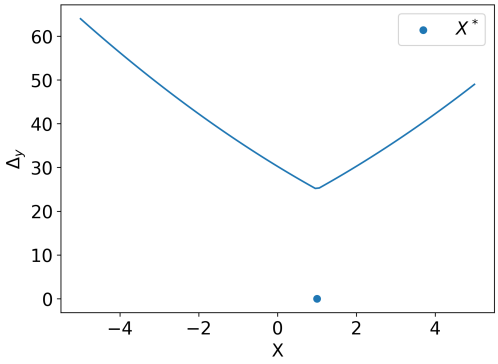
Bayesian optimization refers to a class of numerical optimization methods aimed tackling optimization problems where the objective function is computationally costly to evaluate (e.g.



(a) Illustrative behavior surface



(b) Projection of the output performance on the design axis



(c) Robustness metric Δy as a function of the design x

Figure 1: Illustration of a dummy robust design optimization problem and its solution

a numerical model). Their principle consists in replacing the objective function with a surrogate $\hat{g}$ that is less computationally expensive to evaluate, see e.g. [9]. The tradeoff of such surrogate-based methods is the extra approximation introduced by the prediction error of the surrogate. This error is related to the quality of the surrogate calibration and, in turn, to the size of the calibration dataset. The goal of efficient surrogate methods is to define a semi-optimal calibration dataset as a compromise between the the number of evaluation of the objective function and the error induced to the quantity of interest. Popular methods in the literature use a so-called active-learning approach aiming a building a near-optimal design of experiment iteratively [10, 11, 12]. They take advantage of the stochastic nature of the prediction provided by Gaussian process regression models (GPR), using the variance of the surrogate as an indicator of the local prediction error. The next optimal location to add to the dataset is the one identified (usually through a heuristic) as the location where the local error contributes the most to the error of the quantity of interest.

One of the most popular Bayesian optimization method is the efficient global optimization methods (EGO) [13]. Which proposed a heuristic for selecting the coordinates to refine in the shape of the expected improvement. Let us define y_{min}, the scalar being the current best

estimate of the minimum of the objective function.

$$y_{min} = \min_{x} \hat{g}(x) \tag{3}$$

The expected improvement is defined as:

$$\begin{aligned} \forall x \in I_x, EI(x) &= E\left[max(y_{min} - \hat{g}(x)\,,\,0)\right] \\ &= F(y_{min} \mid x) E(y_{min} - \hat{g}(x) \mid \hat{y}_y(x) \leq \hat{g}_{y\,min}) \\ &= \int_{-\infty}^{y_{min}} (y_{min} - \tau)\, f(\tau \mid x)\, d\tau \end{aligned} \tag{4}$$

with F and f respectively the cumulative and probability density functions of $\hat{g}(x)$. Following the EGO algorithm, the optimal design coordinate of the next point to add the calibration dataset x^* is defined as:

$$x^* = \underset{x \in I_x}{\operatorname{argmax}}\, EI(x) \tag{5}$$

Several approaches are possible to apply Bayesian optimization in solving the nested problem defined in equation1. A first approach could be to solve the outer problem as a deterministic one and the inner problems with Bayesian optimization. The main limitation is that the choice of outer location investigated by the optimization algorithm locks the method into significant calibration effort to turn this location into a deterministically known value, before jumping to another one. This imposes a grid-like shape to the generated calibration data and significant inefficiency.

Some work have been proposed in the Bayesian optimization literature for related min-max problems where two nested optimization problems need to be solved. Authors have proposed a straightforward approach consisting in implementing two nested EGO algorithms considering two distinct GPRs [14]. A more promising framework is to solve directly the outer problem with Bayesian optimization, the difficulty then becomes to propagate the inner uncertainty to this outer problem. Authors have proposed approaches to do so through a worst-case scenario based on the GPR statistical moments [15] or through confidence intervals [16, 17]. This work proposes an new approach falling into this category.

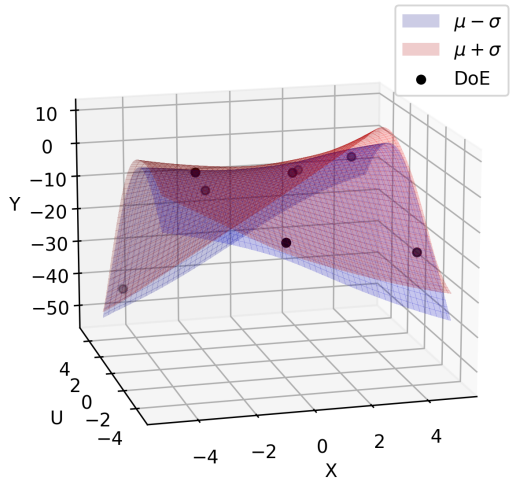
Lets us define the optimization problem considering a GPR surrogate $\hat{g}$ of the performance. The surrogate counterpart of the robustness indicator Δy defined in equation 2 then becomes:

$$\forall x \in I_x,\ \Delta\hat{y}(x) = \max_{u \in I_u} \hat{g}(x,u) - \min_{u \in I_u} \hat{g}(x,u) \tag{6}$$

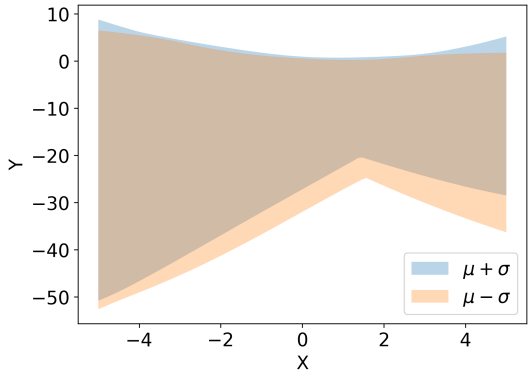
Let us then define $\bar{\hat{g}}$ and $\underline{\hat{g}}$ the two stochastic processes defined as:

$$\forall x \in I_x, \begin{cases} \bar{\hat{g}}(x) = \max\limits_{u \in I_u} \hat{g}(x,u) \\ \underline{\hat{g}}(x) = \min\limits_{u \in I_u} \hat{g}(x,u) \end{cases} \tag{7}$$

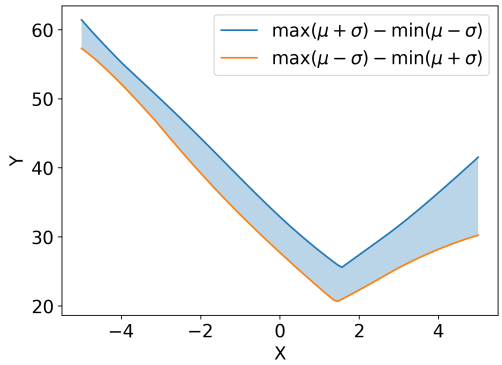
The robustness indicator associated with any design $\Delta\hat{y}(x)$ is a random variable, defined as the difference between the two extrema of the Gaussian stochastic process $\hat{g}(x,u)$, $u \in I_u$. Its stochastic nature is illustrated in Figure 2 using the same dummy function as presented in Figure 1. Figure 2a illustrates the two performance surfaces associated with the mean prediction of the GPR plus and minus its standard deviation. When projected on the design space as in Figure 2b these surfaces highlight the stochastic nature of the performance amplitude. Finally,



(a) Kriging prediction surface $\mu_{x,u}$ and the confidence interval surfaces $\mu_{x,u} \pm \sigma_{x,u}$



(b) Projection of Kriging estimated performances on the design axis



(c) Stochastic robustness metric $\hat{\Delta}y$ and its confidence interval as a function of the design

Figure 2: Illustration of a dummy robust design optimization problem associated with a GPR surrogate of the system

the variance of the robustness metric $\Delta\hat{y}$ is illustrated in Figure 2c by representing its upper and lower bounds associated with the two surfaces.

A major difficulty of implementing a Bayesian optimization procedure to minimize Δy resides in estimating the statistics of $\Delta\hat{y}(x)$ with an affordable computation burden. As previously mentioned, several methods have been proposed to do so, through worst-case scenario or confidence intervals approaches. This work does not aim at estimating precisely the statistics of the robustness indicator, instead a heuristic is developed based on the statistics of extreme value distributions.

3 A heuristic based on extreme value distributions

It should be stressed that estimating the statistics of $\bar{\hat{g}}(x)$ and $\underline{\hat{g}}(x)$ is a very challenging task. It consists in estimating, for any design x, the statistics of both extrema of a non-stationary Gaussian process (the GPR predictive posterior) over the domain I_u. This difficulty and the proposed way of addressing it are further discussed in section 3.2. Another difficulty resides

in estimating the local contribution of the GP variance to these statistics. They result from an aggregated contribution from all Gaussian random variables $g(x,u)$ and differentiating the contribution of each is not trivial. In essence we are interested in a quantity related to the maximum of the partial derivative of the expected improvement with respect to the variance of $\hat{g}(x,u)$, i.e. $\forall x,\ \max_u \frac{\partial EI(x|\hat{g})}{\partial var(\hat{g}(x,u))}$.

Therefore, estimating the statistics of the extrema of the of GP isn't enough, we also need to estimate their sensitivity to the local variance of the GP. In the general case of non-stationary GPs, one has to resort to numerical estimation for those statistics, which would be computationally intractable for the derivation and optimization at hand. This work proposes a heuristic of the local contribution based on the extreme value distributions associated with the Gaussian random variables $\hat{g}(x,u)$. This heuristic is aimed at being informative enough to semi-optimally guide the calibration of the surrogate while introducing a manageable computation burden.

3.1 Identifying semi-optimal coordinates in the uncertain sub-space

Let us consider, for a given design x, a Monte-Carlo population of n trajectories $\{\hat{g}_i\}_{i=1..n}$ of the GP $\hat{g}(x,u),\ u \in I_u$. We can define two Monte-Carlo populations of scalars $\{\underline{\hat{y}}_i\}_{i=1..n}$ and $\{\bar{\hat{y}}_i\}_{i=1..n}$ of the two extrema random variables $\bar{\hat{g}}(x)$ and $\underline{\hat{g}}(x)$ in the shape:

$$\forall i \in [1,n],\ \begin{cases} \bar{\hat{y}}_i = \max\limits_u \hat{g}_i(x,u) \\ \underline{\hat{y}}_i = \min\limits_u \hat{g}_i(x,u) \end{cases} \tag{8}$$

The goal is then to identify the locations $\bar{u}^*$ and $\underline{u}^*$ for which the local random variables $\hat{g}(x,\bar{u}^*)$ and $\hat{g}(x,\underline{u}^*)$ contribute the most to the statistics of $\bar{\hat{g}}(x)$ and $\underline{\hat{g}}(x)$ respectively. It is proposed to consider the random variables $\bar{u}$ and $\underline{u}$ defined as the arguments of the Gaussian process extrema:

$$\forall x \in I_x,\ \begin{cases} \bar{u}(x) = \operatorname{argmax}\limits_u \hat{g}(x,u) \\ \underline{u}(x) = \operatorname{argmin}\limits_u \hat{g}(x,u) \end{cases} \tag{9}$$

an intuitive way of defining $\bar{u}^*$ and $\underline{u}^*$ could be to consider the modes of the distributions of $\bar{u}$ and $\underline{u}$ respectively. This work does not aim at providing an analytical or numerical estimation of those modes, it is instead proposed to define a simple heuristic to providing to a satisfactory near-optimal and guiding the calibration of the surrogate efficiently. To introduce it, let us consider, for any u the Monte-Carlo population of performance values $\{\hat{g}_i(x,u)\}_{i=1..n}$. The extrema of this population are realizations of the random variables $\bar{\hat{g}}(x,u)$ and $\underline{\hat{g}}(x,u)$ defined respectively as the maximum and minimum of n independent identically distributed Gaussian random variables $\hat{g}(x,u)$. From the extreme value theory we know that $\bar{\hat{g}}(x,u)$ and $\underline{\hat{g}}(x,u)$ follow Gumbel distributions which parameters relate directly to mean μ and standard deviation σ of $\hat{g}(x,u)$:

$$\begin{aligned} \forall \{x,u\} \in I_x * I_u\,,\ \bar{\hat{g}}(x,u) = \max\{\hat{g}_i(x,u)\}_{i\in[1,n]} &\sim \text{Gumbel}(\bar{\nu},\bar{\beta}) \\ \bar{\nu}\,(x,u,n) &= \Phi^{-1}\left(1-\frac{1}{n}\right)\sigma + \mu \\ \bar{\beta}\,(x,u,n) &= \Phi^{-1}\left(1-\frac{1}{n}e^{-1}\right) - \bar{\nu} \end{aligned} \tag{10}$$

and:

$$\begin{aligned}
\forall\{x,u\} \in I_x * I_u\,,\ \underline{\hat{g}}(x,u) &= \min\{\hat{g}_i(x,u)\}_{i\in[1,n]} \sim \text{Gumbel}(\underline{\nu},\underline{\beta}) \\
\underline{\nu}\,(x,u,n) &= \Phi^{-1}\left(\frac{1}{n}\right)\sigma + \mu \\
\underline{\beta}\,(x,u,n) &= -\Phi^{-1}\left(\frac{1}{n}e^{-1}\right) + \underline{\nu}
\end{aligned} \tag{11}$$

with Φ^{-1} the inverse of the standard Gaussian cumulative distribution function. It is then proposed to use the expected value of $\bar{\hat{g}}(x,u)$ and $\underline{\hat{g}}(x,u)$ as heuristics for the value of the density of $\bar{u}$ and $\underline{u}$ at the location u. This choice can be seen as considering the expected value of the extremum of the population of trajectories, evaluated at the location u, as indicative of the likelihood of u being the argument of the extremum a trajectory over the domain I_u. The definitions of $\bar{u}^*$ and $\underline{u}^*$ then naturally become:

$$\forall x \in I_x \begin{cases} \bar{u}^* = \underset{u\in I_u}{\text{argmax}}\, E\left(\bar{\hat{g}}(x,u)\right) \\ \underline{u}^* = \underset{u\in I_u}{\text{argmin}}\, E\left(\underline{\hat{g}}(x,u)\right) \end{cases} \tag{12}$$

or

$$\forall x \in I_x \begin{cases} \bar{u}^*(x) = \underset{u\in I_u}{\text{argmax}}\left(\bar{\nu}\,(x,u,n) + \gamma\bar{\beta}\,(x,u,n)\right) \\ \underline{u}^*(x) = \underset{u\in I_u}{\text{argmin}}\left(\underline{\nu}\,(x,u,n) - \gamma\underline{\beta}\,(x,u,n)\right) \end{cases} \tag{13}$$

with γ the Euler's constant. The two semi-optimal locations can then estimated by solving two optimization problems following equation 14 and involving one evaluation of the GPR per iteration each. It is finally proposed to only evaluate the performance function at one of those locations, i.e. the one associated with the highest GPR variance.

3.2 Identifying semi-optimal coordinates in the design sub-space

As introduced in section 2, the goal is to solve the outer optimization problem (i.e. the one defined over the design sub-space) with the EGO algorithm. The challenge then resides in estimating the expected improvement associated with any design coordinate x. The outer-optimization problem can be framed as the minimization of the stochastic process $\Delta\hat{y}$ defined in equation 6. Evaluating the expected improvement requires the distribution of $\Delta\hat{y}(x)$, which implies knowing the distribution of $\bar{\hat{g}}(x)$ and $\underline{\hat{g}}(x)$. Unfortunately estimating the distribution of the extrema of non-stationary Gaussian processes is notoriously difficult and require numerous numerical estimations [18], the most notorious approach being based on the Rice series [19]. It is therefore proposed to define another heuristic for this distribution, deriving from the one introduced in section 3.1.

The principle is, for any design x, to reduce the Gaussian process $\hat{g}(x)$ to two independent random variables, i.e. the Gaussian random variables $\hat{g}(x,\bar{u}^*(x))$ and $\hat{g}(x,\underline{u}^*(x))$. It consists in reducing the continuous Gaussian process to the two random variables that contribute the most to the process extrema distributions. Doing so provides us with a simple indicator of $\Delta\hat{y}(x)$ as the difference of the two Gumbel distributions described in equations eqs. (10) and (11) at the locations $\bar{u}^*(x)$ and $\underline{u}^*(x)$ respectively:

$$\forall x \in I_x,\ \Delta\hat{y}(x) = \max\{\hat{g}_i(x, \bar{u}^*(x))\}_{i\in[1,n]} - \min\{\hat{g}_i(x, \underline{u}^*(x))\}_{i\in[1,n]} \tag{14}$$

Since the two Gumbel distributions are known, the distribution $f_{\Delta_{\hat{y}}}$ of $\Delta\hat{y}(x)$ can be estimated numerically by convolution. The expected improvement, as introduced in equation 4 can finally be estimated by numerical integration using the distribution of the the heuristic defined above:

$$\forall x \in I_x, EI^{EV}(x) = \int_{-\infty}^{\Delta_{min}} (\Delta_{min} - \tau)\, f_{\hat{\Delta}_y^{EV}}(\tau \mid x)\, d\tau \tag{15}$$

With $f_{\hat{\Delta}_y}$ the distribution of the difference of the two Gumbel-distributed random variables at the locations $\bar{u}^*(x)$ and $\underline{u}^*(x)$, and Δ_{min} the current best estimate of the minimum of $\Delta_{\hat{y}}$

3.3 The active learning algorithm

The active learning algorithm employed in this work is based on the EGO algorithm, with the primary difference being that EGO is used here to identify the minimum of the (non-Gaussian) stochastic process $\hat{\Delta}_y$, as defined in Equation 6. While the expected improvement criterion admits a closed-form expression in the case of a Gaussian process, an additional step is required when adapting it to robust design optimization. In this context, the additional step involves solving two optimization problems to determine the coordinates $\bar{u}(x, n)$ and $\underline{u}(x, n)$ in the uncertainty space, as outlined in Section 3.1. The full algorithm, including this extension, is summarized in the flowchart shown in Figure 3, where the modifications to the standard EGO approach are highlighted in blue.

Several components of the algorithm are treated as hyperparameters. These include the choice of kernel for the Gaussian Process Regression (GPR), the number of virtual Monte Carlo samples n introduced in Section 3.1, the strategy and size of the initial design of experiments, and the optimization methods used to identify $\bar{u}(x, n)$, $\underline{u}(x, n)$, and x^*. The present study reports preliminary results from ongoing research, and the sensitivity of the algorithm's performance to these parameters has not yet been explored. As an initial step, these choices were made arbitrarily to examine convergence behavior and performance. Specifically, a squared exponential kernel was selected, the virtual Monte Carlo population size was set to $n = 100$, and the initial design of experiments was generated using a Latin Hypercube sampling strategy with a size equal to three times the problem dimension. The optimization problems were solved using an interior point algorithm, complemented by a 10-starts distributed search strategy to improve the chances of reaching global optima.

The stopping criterion for the algorithm, in this early stage, is limited to a fixed simulation budget: the number of simulations is not allowed to exceed a predefined threshold. Future work will explore more adaptive and efficient stopping criteria.

4 Examples and results

4.1 A benchmark function with local minima

The performance of the proposed approach is first demonstrated using a two-dimensional test function characterized by the presence of local minima. This example is defined through the following performance function g:

$$\forall x, u \in I_x, I_u, ; g(x) = xu - \sin(x)u^2 + x^2 \tag{16}$$

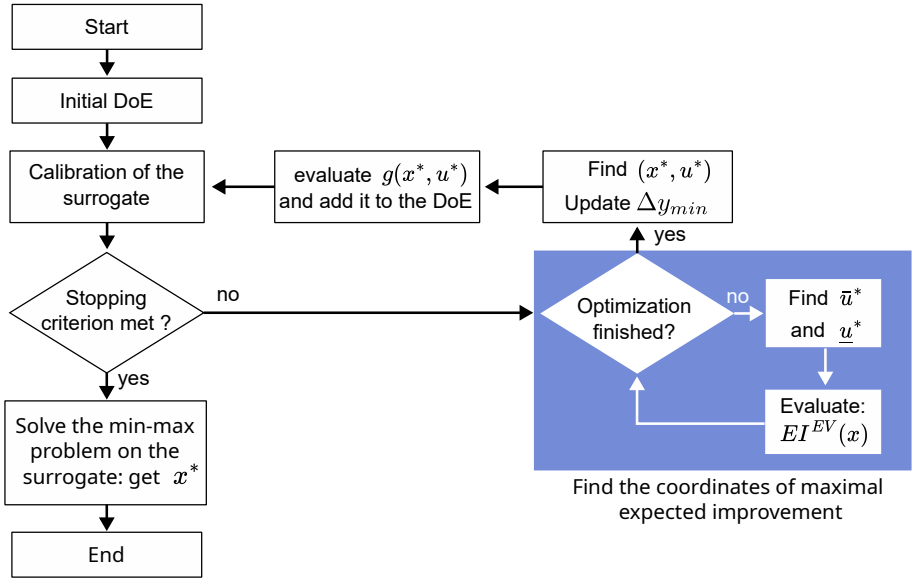
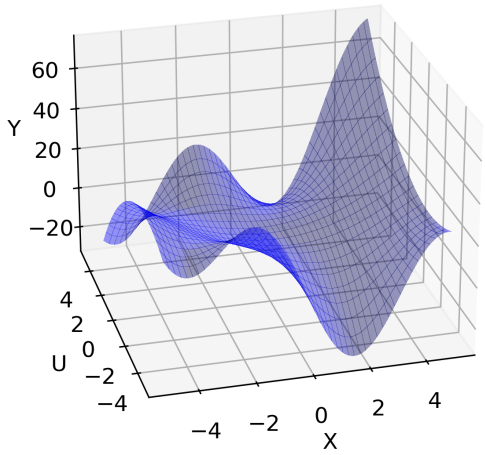
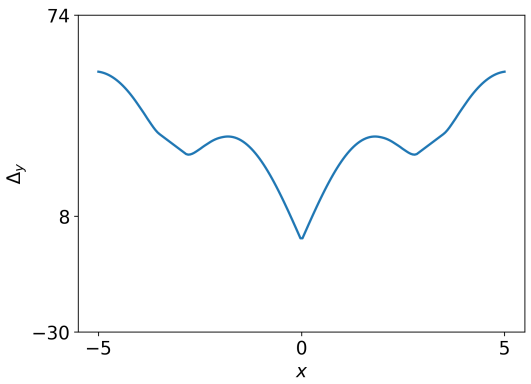


Figure 3: Flowchart of the proposed robust optimization algorithm.

The design variable x belongs to the interval $I_x = [-5, , 5]$, and the uncertain variable u is defined over $I_u = [-5, 5]$. The known optimal design is located at $x^* = 0$. Although the function is relatively simple, it yields a highly non-linear robustness function with multiple local minima and non-differentiability at the optimum. This test case has previously been used in the same context to evaluate a method based on confidence intervals, referred to as AK-RULOK, which was introduced by the authors in [16]. In that study, AK-RULOK converged to the optimal design after 30 iterations, resulting in an error of approximately 0.02 in the design coordinate. It will serve here as a baseline for comparison with the current method.



(a) Behavior surface of the first example function



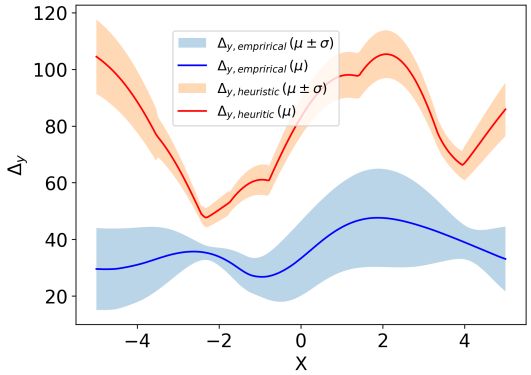
(b) Surface of the robustness metric Δ_y for the first example function

Figure 4: Illustration of the behavior and optimal solution of the first example function

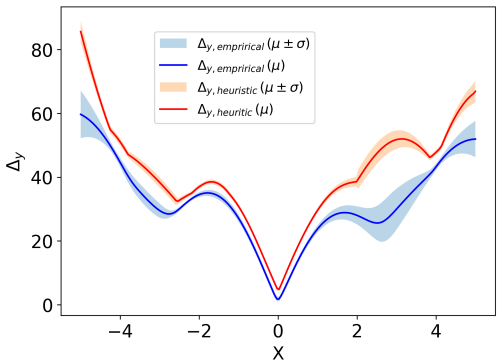
The results are first presented in Figure 5c, which displays the convergence of the optimal

design point over the course of the optimization and the convergence amplitude over 15 repetitions of the method. The plot indicates that the algorithm successfully converges to the correct optimal value, with convergence observed after approximately 21 iterations, or 27 evaluations of the performance function, with an error interval of $[-7.4e^{-2}, 4.0e^{-2}]$. Those results suggest that the proposed approach can achieve similar or better performance than existing methods in the literature.

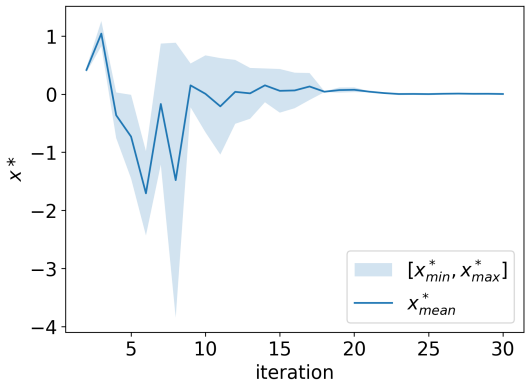
Further insight into the convergence behavior is provided in Figures 5a, 5b, which depict the evolution of the heuristic robustness metrics $\hat{\Delta}_y^{EV}$ introduced in Section 3.2. The heuristic is plotted with its mean surface and its plus/minus standard deviation envelope and is compared to empirical counterparts obtained with a Monte-Carlo population of GPR realizations. These figures correspond to iterations 2, and 20, respectively, representing different stages of convergence. Although the indicator appears to be biased relative to the robustness function estimated via the empirical GPR model, it successfully captures the overall trends of the system behavior. It should also be mentioned that a constant bias does not affect the efficiency of the method since the heuristic is only used comparatively to other heuristic values. Importantly, both the heuristic $\hat{\Delta}_y^{EV}$ and the GPR-estimated robustness function $\hat{\Delta}_y$ ultimately converge toward the same deterministic profile in the region of interest.



(a) Robustness surface metric estimated by the GP at iteration 3



(b) Robustness surface metric estimated by the GP at iteration 20



(c) Convergence of the optimal robust design coordinates compared to the true optimal

Figure 5: Illustration of the convergence of the algorithm in terms of optimal solution and robustness function approximation on the first example

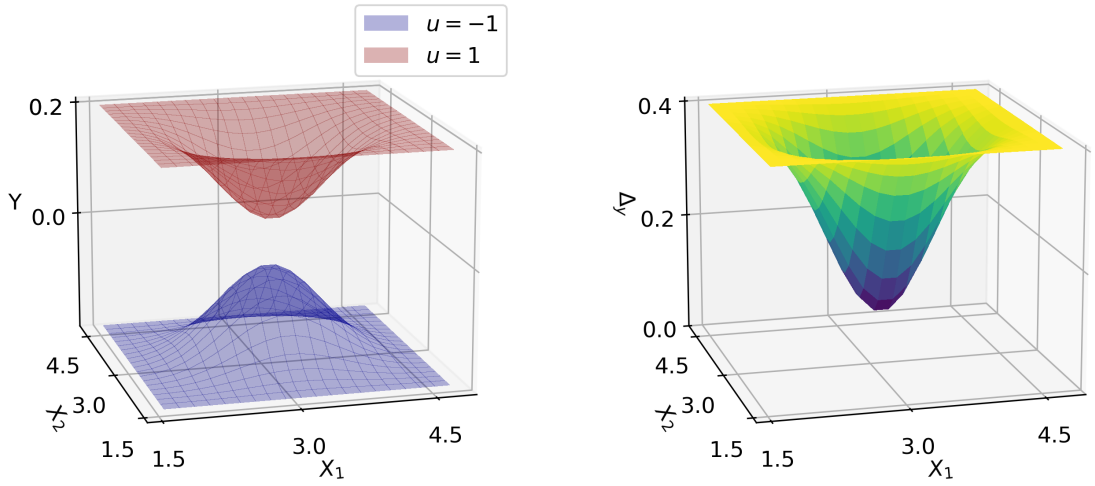
4.2 A benchmark function in dimension three

The second example involves a three-dimensional function derived from the Easom function, as defined in Equation 17. In this case, the design space is two-dimensional, with the boundaries for both variables given by $I_{x_1} = I_{x_2} = [\frac{\pi}{2}, \frac{3\pi}{2}]$. The uncertain variable u is defined over the interval $I_u = [-1, 1]$. Within this domain, the function is monotonic with respect to the variable u, which simplifies the interpretation of the resulting behavior surfaces. The function's shape and the corresponding optimal design coordinates are shown in Figures 6a and 6b, respectively.

$$\begin{aligned} &\forall\{x, u\} \in \{I_x.^2, I_u\}, \\ &g(x) = -cos(x_1)cos(x_2)sin\left(\frac{u}{2\pi}\right) e^{-(x_1-x_1^*)^2+(x_2-x_2^*)^2} + \frac{u}{5} \end{aligned} \tag{17}$$

with $x^* = \{x_1^*, x_2^*\} = \{\pi, \pi\}$ the optimal design.

In order to illustrate the repeatability of the convergence, the proposed method has been repeated fifteen times over forty iterations each.



(a) Behavior surface of the second example function considering two values of u

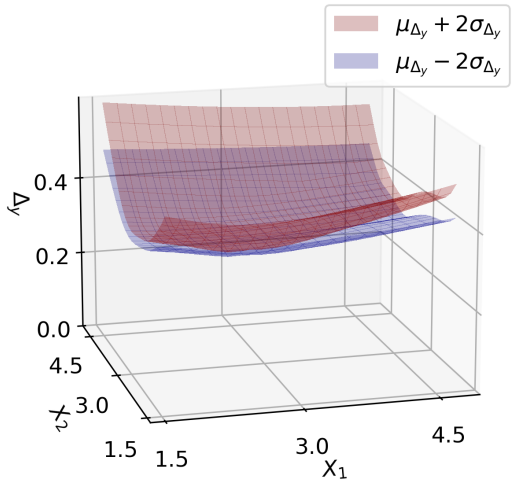
(b) Surface of the robustness metric Δ_y for the second example function

Figure 6: Illustration of the behavior and optimal solution of the second example function

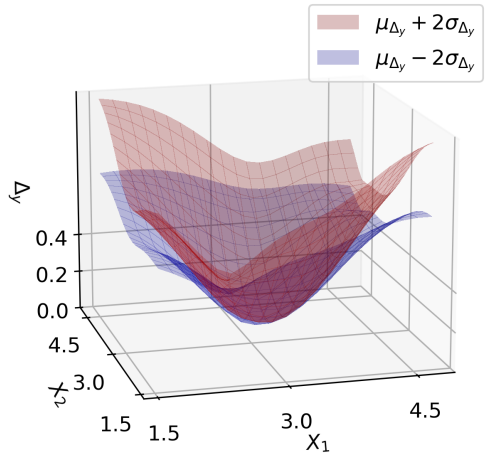
The first results presented in Figures 7a and 7b represent the heuristic $\hat{\Delta}_y$ surface at an early iteration and a late one. The second figure show the convergence of the heuristic metric to the actual Δ_y surface (see Figure 6b) around the optimum. The second result presented in Figure 7c is the convergence of the method over forty iterations. Both the convergence of x_1^* and x_2^* are represented in blue and orange respectively. The color surfaces represent the min and max of both coordinates over fifteen repetitions of the method. The figure highlight the fast convergence of the method with the variance over fifteen applications being visibly negligible after thirty iterations (i.e. 39 evaluations of the performance function).

4.3 A forming process finite element example

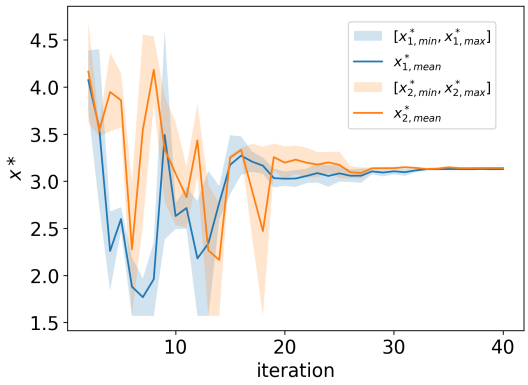
The last example is a 2D finite element model of a forming process simulating the plastic deformation of a metal sheet (DP780 steel with 1.4 mm thickness) held in contact with the die



(a) Robustness surface metric estimated by the GP at iteration 3



(b) Robustness surface metric estimated by the GP at iteration 20



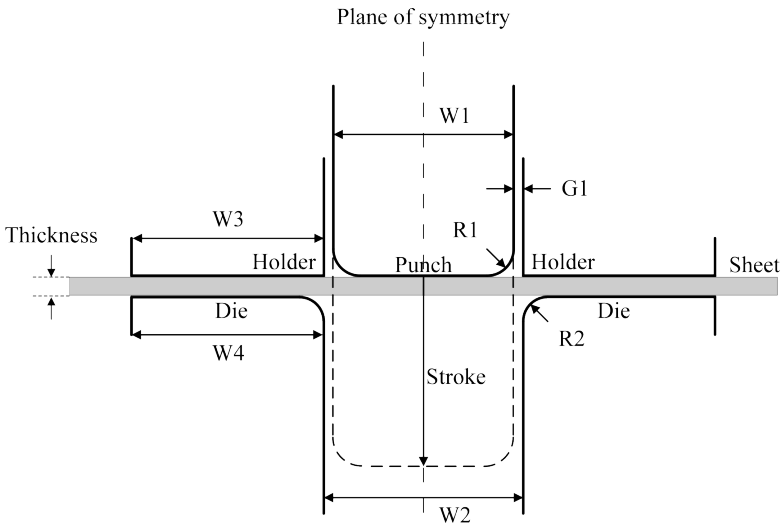
(c) Convergence of the optimal robust design coordinates compared to the true optimal

Figure 7: Illustration of the convergence of the algorithm in terms of optimal solution and robustness surface approximation on the second example

with a holder and deformed in a "U" shape by the punch, see [20]. The simulation includes the spring back after release of the elastic deformation and we are interested in the shape of the profile after spring back. The formed profile is considered dependent on only to variables R_2 and R_1, the radius of the die and punch edge rounding respectively. The Punch radius is considered as an interval uncertainty with the interval $I_{R1} = [2, 5]$. The Die radius is considered as the design parameter with a design space $I_{R1} = [1, 10]$.

The performance metric is defined at the root mean square error between the nodal coordinates of the profiles before and after spring back (see example in Figure 9).

The RDO problem is then to find the die radius resulting in the most consistent profile after spring back, despite variation of the punch radius. The convergence of the algorithm is presented in Figure 10a with a fast convergence to the final solution. The convergence plot exhibits a noisy behavior with the algorithm alternating between two solutions indicating the presence of another local minimum. Figure 10b illustrates ten profiles after spring back taken form ten linearly spaced values of the die radius between $2mm$ and $5mm$, considering the optimal punch



Parameters	W1	W2	W3	W4	R1	R2	G1	Stroke
Dimensions	50.0	54.0	89.0	89.0	[2, 5]		2.0	71.8

(Unit: mm)

Figure 8: Schematic representation and dimensions of the forming process

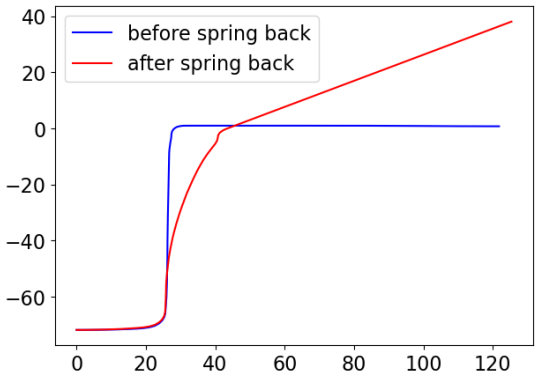
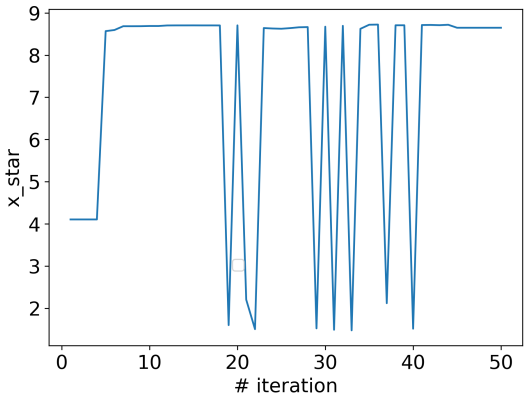


Figure 9: Illustration of the formed sheet profile before and after spring back

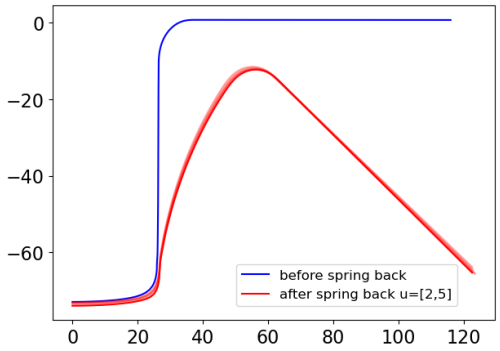
radius found by the algorithm. The proposed solution exhibits a very high spring back and its compliance with the desired shape is quite poor. However the profiles are very consistent with an extremely limited influence form the variations of the die radius. This results highlights the importance of extending this method in a multi-objective setting to find a tradeoff between the value of the error and the amplitude of ts variations. Nevertheless, the identified solution is very satisfactory with respect to the assigned objective, which is only to generate a profile which is robust to the variations of R_1.

5 Conclusions

This paper introduces a novel learning function along with an adaptive calibration algorithm specifically designed for robust design optimization under interval uncertainties. The core objective is to define a heuristic distribution that guides the calibration of the Gaussian Process Regression (GPR) model through the estimation of the expected improvement function. This heuristic is intended to be computationally efficient, reflective of the GPR-based robustness metric, and capable of highlighting the local influence of each design variable.



(a) conergence of the algorithm over the finite element example



(b) Illustration of the solution found by the algorithm with (red) ten profiles after spring back corresponding to ten values of $R_1 \in [2, 5]$

Figure 10: Illustration of results from the finite element example

The proposed approach involves generating a virtual Monte Carlo population and, for each design point, constructing the heuristic by taking the convolution of the two extreme value distributions derived from the most extreme expected values within the Monte Carlo sample.

To validate the method, it is applied to two analytical benchmark problems and a finite element example as a proof of concept. In both analytical cases, the algorithm converges efficiently and accurately to the correct optimal design, demonstrating performance that is comparable to or slightly better than existing methods. On the finite element example, the algorithm identified a solution which is poor in terms of industrial compliance but which does satisfies very well the robustness criterion. These initial results are promising and form a strong foundation for further investigation. Future work will include a detailed sensitivity analysis of key hyperparameters such as the size of the virtual Monte Carlo population as well as performance evaluations on a more comprehensive and reproducible benchmark set.

6 A GitLab repository

The recent work on this method focused on proposing a python version of the method shared on a GitLab repository under the address https://git.utt.fr/persoons/RDO. The project has two branches using the SMT [21] and GPJax [22] toolboxes for Gaussian process calibration. The project includes:

- a module python file *module.py* including all the functions of the method, including the estimation of the expected improvement as well as eight predefined benchmark functions
- a jupyter notebook *demo.ipynb* illustrating step by step the problem at hand and the convergence of the method.
- a jupyter notebook *demo_EI.ipynb* illustrating the calculation of the expected improvement heuristic with fast Fourier transform and graphically comparing it to empirical histograms drawn from a Monte-Carlo population.
- a python file *results_generation.py* allowing to run repeated applications of the method and used to generate the results presented in this paper.

REFERENCES

[1] G. Taguchi. *Quality engineering (Taguchi methods) for the development of electronic circuit technology*, IEEE Transactions on Reliability, vol. 44, no. 2, pp. 225–229, Jun. 1995.

[2] R. Jin, X. Du, W. Chen, and W. Chen, *The use of metamodeling techniques for optimization under uncertainty*, Structural and Multidisciplinary Optimization, vol. 25, no. 2, pp. 99–116, Jun. 2003

[3] J. Janusevskis and R. Le Riche. *"Simultaneous kriging-based estimation and optimization of mean response*, Journal of Global Optimization, vol. 55, no. 2, pp. 313–33, Feb. 2013.

[4] B. J. Williams, T. J. Santner, and W. I. Notz, *Sequential design of computer experiments to minimize integrated response functions*, Statistica Sinica, vol. 10, pp. 1133–1152, 2000.

[5] H.-G. Beyer and B. Sendhoff, *Robust optimization – A comprehensive survey*, Computer Methods in Applied Mechanics and Engineering, vol. 196, no. 33-34, pp. 3190–3218, Jul. 2007.

[6] C. Zang, M. Friswell, and J. Mottershead, *A review of robust optimal design and its application in dynamics*, Computers & Structures, vol. 83, no. 4-5, pp. 315–326, Jan. 2005.

[7] D. Bertsimas, D. B. Brown, and C. Caramanis, *Theory and Applications of Robust Optimization*, IAM Review, vol. 53, no. 3, pp. 464–501, Jan. 2011.

[8] B. Rustem and M. Howe, *Algorithms for worst-case design and applications to risk management*, Princeton, N.J. ; Oxford: Princeton University Press, 2002

[9] Greenhill, Stewart, Santu Rana, Sunil Gupta, Pratibha Vellanki, et Svetha Venkatesh, *Bayesian Optimization for Adaptive Experimental Design: A Review* , IEEE Access 8 (2020): 13937-48. https://doi.org/10.1109/ACCESS.2020.2966228.

[10] Gramacy, Robert B., *Surrogates: Gaussian Process Modeling, Design and Optimization for the Applied Sciences*, Boca Raton, Florida: Chapman Hall/CRC, 2020.

[11] Jiang, Ping, Qi Zhou, et Xinyu Shao, *Surrogate Model-Based Engineering Design and Optimization. Springer Tracts in Mechanical Engineering*, Singapore: Springer Singapore, 2020. https://doi.org/10.1007/978-981-15-0731-1.

[12] Moustapha, M., S. Marelli, et B. Sudret, *A Generalized Framework for Active Learning Reliability: Survey and Benchmark*, arXiv:2106.01713 , 3 juin 2021.

[13] Jones, Donald R, et Matthias Schonlau, *Efficient Global Optimization of Expensive Black-Box Functions*, Journal of Global Optimization 13 (1998): 455 92

[14] Marzat, Julien, Eric Walter, et Hélène Piet-Lahanier, *A New Expected-Improvement Algorithm for Continuous Minimax Optimization*, Journal of Global Optimization 64, no 4 (avril 2016): 785 802. https://doi.org/10.1007/s10898-015-0344-x.

[15] Rehman, Samee ur, Matthijs Langelaar, et Fred van Keulen, *Efficient Kriging-Based Robust Optimization of Unconstrained Problems* , Journal of Computational Science 5, no 6 (novembre 2014): 872 81. https://doi.org/10.1016/j.jocs.2014.04.005.

[16] Mierlo, Conradus van, Augustin Persoons, Matthias G. R. Faes, et David Moen, *Robust Design Optimisation under Lack-of-Knowledge Uncertainty* , Computers & Structures 275 (15 janvier 2023): 106910. https://doi.org/10.1016/j.compstruc.2022.106910.

[17] Mierlo, Conradus van, Augustin Persoons, Matthias G. R. Faes, et David Moen, *Robust Design Optimization of Expensive Stochastic Simulators Under Lack-of-Knowledge*, ASCE-ASME J Risk and Uncert in Engrg Sys Part B Mech Engrg 9, no 2 (1 juin 2023): 021205. https://doi.org/10.1115/1.4056950.

[18] Azaïs, Jean-Marc, et Li-Vang Lozada-Chang, *A Toolbox on the Distribution of the Maximum of Gaussian Process*, 2013. hal-00784874

[19] Lindgren, Georg., *Gaussian Integrals and Rice Series in Crossing Distributions—to Compute the Distribution of Maxima and Other Features of Gaussian Processes*, Statistical Science 34, no 1 (1 février 2019). https://doi.org/10.1214/18-STS662.

[20] H. Huh, K. Chung, SS. Han, et WJ. Chung, *Benchmark Study of the 8th International Conference and Workshop on Numerical Simulation of 3D Sheet Metal Forming Processes*, Proceedings of Numisheet 2011

[21] P. Saves and R. Lafage and N. Bartoli and Y. Diouane and J. Bussemaker and T. Lefebvre and J. T. Hwang and J. Morlier and J. R. R. A. Martins, *SMT 2.0: A Surrogate Modeling Toolbox with a focus on Hierarchical and Mixed Variables Gaussian Processes*, Advances in Engineering Sofware, 2024, vol. 188, https://doi.org/10.1016/j.advengsoft.2023.103571.

[22] Thomas Pinder and Daniel Dodd, *GPJax: A Gaussian Process Framework in JAX*, Journal of Open Source Software, 2022, vol. 7, https://doi.org/10.21105/joss.04455.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

EFFICIENT CONFIDENCE BOUND ESTIMATION FOR A DYNAMIC COMPOSITE PLATE WITH EPISTEMIC UNCERTAINTIES USING NON-INTRUSIVE IMPRECISE STOCHASTIC SIMULATION

Ewerton Grotti[1], Matteo Broggi[2], Herbert M. Gomes[1], and Michael Beer[2]

[1]Federal University of Rio Grande do Sul
Department of Mechanical Engineering, Av. Sarmento Leite, 425, Porto Alegre, RS, Brazil
e-mail: ewerton.grotti@ufrgs.br, herbert@mecanica.ufrgs.br

[2]Leibniz University Hannover
Institute for Risk and Reliability, Callinstr. 34, 30167, Hannover, Germany
broggi@irz.uni-hannover.de, beer@irz.uni-hannover.de

Abstract

This study investigates the dynamic response of composite plates under epistemic uncertainties, focusing on confidence bounds for vertical displacements. A composite plate with a clamped-free-free-free boundary condition is modeled using the Finite Element Method (FEM) based on Mindlin plate theory. The material properties and loadings are characterized using probabilistic distributions and contain epistemic uncertainty. To approximate the confidence bounds (envelopes) for output variables, Non-Intrusive Imprecise Stochastic Simulation (NISS) method is applied, and the resulting High-Dimensional Model Representation (HDMR) equations are optimized to obtain the confidence limits for CDFs. This procedure provides a computationally efficient approach compared to the resource-intensive nested double-loop Monte Carlo Simulation (MCS) for epistemic variables. The results provide a comparative analysis of confidence bounds generated by both methods, highlighting the efficiency and the associated error of the proposed NISS based framework.

Keywords: Epistemic Uncertainty, Non-Intrusive Imprecise Stochastic Simulation, Double-Loop Monte Carlo, Composite Plate, Vibration.

1 INTRODUCTION

In the fields of engineering and material sciences, understanding the behavior of composite materials under uncertain conditions is a growing concern. Composite materials are known for their lightweight combined with high-strength properties and are widely used in applications such as aerospace, automotive, and civil engineering. However, their performance is influenced by a multitude of factors, ranging from material properties to excitation forces, many of which are inherently uncertain [1, 2]. These uncertainties stem from complex manufacturing processes, variations in constituent material properties, inherent defects, and diverse excitation loads and boundary conditions.

Probabilistic descriptions of uncertainties in composite materials are often hindered by insufficient data [3]. As a result, epistemic uncertainty presents a significant challenge, potentially compromising the safety and efficiency of critical designs. Direct approaches to propagating epistemic uncertainty, such as the nested double-loop Monte Carlo Simulation (MCS) method, require an exceptionally large number of deterministic simulations, which often lead to computationally intractable procedures [4]. This issue is made worse in engineering applications where the deterministic simulations are computationally demanding, such as those based on the finite element method (FEM). To reduce computational costs, metamodels are frequently employed, such as Kriging, Radial Basis Function, Polynomial Chaos Expansion, and Neural Networks, as detailed in [1].

Approaches for problems involving epistemic uncertainties include the interval method [5-8], fuzzy theory [9-13], among others. These are called non-probabilistic methods and have been popularly adopted in structural dynamics [14]. The approach named imprecise probabilistic method, also called P-box, has also gained attention [15, 16], where a distribution with uncertain statistics is assumed. Among the recent advancements in this category, the Non-Intrusive Imprecise Stochastic Simulation (NISS) has emerged as a promising technique, as demonstrated in [17-20]. According to [21], the P-box model is still rarely applied to analyze the uncertainty of composites. This study aims at this gap by proposing a novel framework to efficiently tackle dynamic problems involving epistemic uncertainties. This work represents the first attempt in the literature to apply NISS to composite vibration problems solved in the time domain to the best of the author's knowledge.

This study is organized into five sections, beginning with this introduction as the first section. The second section provides an overview of the theoretical framework related to the simulation of the composite plate used in this work. The third section introduces the methods for addressing epistemic uncertainty, specifically NISS and MCS. An illustrative example with the corresponding solution is described in the fourth section. Finally, the study concludes with a summary of findings and insights in the fifth section.

2 COMPOSITE PLATES THEORY

Composite materials are engineered by combining two or more constituent substances with significantly different mechanical properties. This combination results in a material with characteristics distinct from those of its individual components. While composites exhibit exceptional mechanical properties, their complex heterogeneous structures and anisotropic nature pose unique challenges for traditional analytical methods. The FEM overcomes these challenges by discretizing the material into smaller, manageable elements [22], allowing for detailed simulations of stress distribution, deformation and failure mechanisms.

2.1 Finite Element Method for Mindlin Plates

The composite plate in this study is simulated using FEM by means of Mindlin plate theory (first-order shear deformation theory, FSDT). According to [23], the displacement field, in this case, is

$$u(x,y,z) = u_0(x,y) + z\theta_x(x,y), \tag{1}$$

$$v(x,y,z) = v_0(x,y) + z\theta_y(x,y), \tag{2}$$

$$w(x,y,z) = w_0(x,y), \tag{3}$$

where u, v, and w are the displacements in x, y, and z, respectively, θ_x is the rotation on the x axis, and θ_y is the rotation on the y axis.

$$\begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \\ \gamma_{xz} \\ \gamma_{yz} \end{Bmatrix} = \left\{ \frac{\partial u}{\partial x} \quad \frac{\partial v}{\partial y} \quad \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \quad \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \quad \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) \right\}^T \tag{4}$$

To define the stiffness matrix, one should start from the strain energy and integrate it throughout each layer over the thickness direction. The membrane part of the stiffness matrix $\boldsymbol{K}_{mb}^{(e)}$, the membrane-bending coupling components $\boldsymbol{K}_{mb}^{(e)}$ and $\boldsymbol{K}_{bm}^{(e)}$, the bending $\boldsymbol{K}_{bb}^{(e)}$ and the shear component $\boldsymbol{K}_{ss}^{(e)}$ are obtained as follows

$$\boldsymbol{K}_{mm}^{(e)} = \sum_{k=1}^{nc} \int_A \boldsymbol{B}_m^T \boldsymbol{D_k} \boldsymbol{B_m} \, (z_{k+1} - z_k) dA \tag{5}$$

$$\boldsymbol{K}_{mb}^{(e)} = \sum_{k=1}^{nc} \int_A \boldsymbol{B}_m^T \boldsymbol{D_k} \boldsymbol{B_b} \frac{1}{2} \, (z_{k+1}^2 - z_k^2) dA \tag{6}$$

$$\boldsymbol{K}_{bm}^{(e)} = \sum_{k=1}^{nc} \int_A \boldsymbol{B_b}^T \boldsymbol{D_k} \boldsymbol{B_m} \frac{1}{2} \, (z_{k+1}^2 - z_k^2) dA \tag{7}$$

$$\boldsymbol{K}_{bb}^{(e)} = \sum_{k=1}^{nc} \int_A \boldsymbol{B_b}^T \boldsymbol{D_k} \boldsymbol{B_b} \frac{1}{3} \, (z_{k+1}^3 - z_k^3) dA \tag{8}$$

$$\boldsymbol{K}_{ss}^{(e)} = \sum_{k=1}^{nc} \int_A \boldsymbol{B_s}^T \boldsymbol{D_k} \boldsymbol{B_s} \, (z_{k+1} - z_k) dA \tag{9}$$

where nc is the number of layers, $\boldsymbol{D}$ is the rotated constitutive matrix, z_k is the coordinate for the k^{th} layer in the thickness direction with the reference being the middle plane of the composite plate as shown in Figure 1 (starting at $k = 1$ from the bottom). $\boldsymbol{B_m}$, $\boldsymbol{B_b}$ and $\boldsymbol{B_s}$ are the strain-displacement matrices for membrane, bending and shear, respectively, which hold information regarding the shape function. For this study, the shape function is assumed as a bilinear four-noded Q4 element as shown in [23].

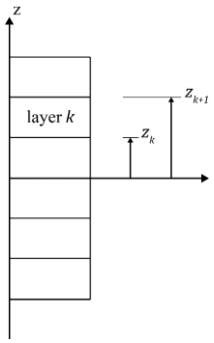


Figure 1: Laminated plate layer diagram. Based on Ferreira, 2014 [23].

Thus, the element stiffness matrix results in

$$\boldsymbol{K}^{(e)} = \boldsymbol{K}_{mm}^{(e)} + \boldsymbol{K}_{mb}^{(e)} + \boldsymbol{K}_{bm}^{(e)} + \boldsymbol{K}_{bb}^{(e)} + \boldsymbol{K}_{ss}^{(e)}. \tag{10}$$

The global stiffness matrix $\boldsymbol{K}$ is then assembled through superposition. The corresponding global mass matrix $\boldsymbol{M}$ is obtained in an analogous way.

2.2 Micromechanics for Elastic Properties

Using the constituents' properties it is possible to derive the mechanical properties of the composite plate. The elastic properties can be estimated by means of the Rule of Mixtures [24], defined as a function of the volume fraction of fibers and matrix.

$$E_1 = E_f V_f + E_m V_m \tag{11}$$

$$1/E_2 = V_f/E_f + V_m/E_m \tag{12}$$

$$V_m = 1 - V_f \tag{13}$$

where E_1 and E_2 are the longitudinal and transversal elastic modulus, V_f and V_m are the volume fraction of fiber and matrix, and E_f and E_m are the elastic modulus of fiber and matrix, respectively.

$$\nu_{12} = \nu_f V_f + \nu_m V_m \tag{14}$$

$$\nu_{21} = \nu_{12} E_2/E_1 \tag{15}$$

$$1/G_{12} = V_f/G_f + V_m/G_m \tag{16}$$

with ν_{21} and ν_{12} being the major and minor Poisson's ratio. G_f and G_m are the shear modulus of the fiber and matrix, and G_{12} is the in-plane shear modulus.

3 SIMULATION METHODS FOR DEALING WITH EPISTEMIC UNCERTAINTY

Epistemic uncertainty arises from a lack of knowledge or incomplete information in modeling complex systems. This chapter focuses on NISS and MCS approaches, providing a foundation for their application in subsequent sections of this study.

3.1 Nested Double-Loop Monte Carlo Simulation

A nested double-loop Monte Carlo simulation is the brute force approach when dealing with epistemic uncertainties. The following pseudocode in Table 1 portrays the code implemented in Matlab, used to generate the results in this study.

Define the number of samples for the inner loop, $\boldsymbol{N_{in}}$, and outer loop, $\boldsymbol{N_{out}}$.
Outer loop: for $\boldsymbol{i = 1}$ to $\boldsymbol{N_{out}}$
  Sample the epistemic interval of each epistemic parameter.
  Inner loop: for $\boldsymbol{j = 1}$ to $\boldsymbol{N_{in}}$
    Sample the uncertain variables using the epistemic parameters $\boldsymbol{i}$.
  end
  Apply the $\boldsymbol{N_{in}}$ sampled uncertain variables in the desired function.
end
Generate one CDF for each set of $\boldsymbol{N_{in}}$ samples of the desired function.
Find the minimum and the maximum from all CDFs in each PQ (probability quantile).
Output: Build the graphical representation of the p-box using the set of minima and maxima extracted from the CDFs in each PQ.

Table 1: Pseudocode for double-loop Monte Carlo.

3.2 Non-intrusive Imprecise Stochastic Simulation

To deal with epistemic uncertainty while avoiding the high cost of the double-loop MCS, the NISS framework may be used. This method was proposed by Wei et al. [17] and uses a combination of mathematical techniques like HDMR decomposition (High Dimensional Model Representation), and importance sampling to approximate quantities of interest such as mean, standard deviation, and failure probability. In Wei et al. [17], local (cut-HDMR) and global (RS-HDMR) approaches are presented, where the local approach is more precise around the chosen combination of epistemic parameters, called $\boldsymbol{\theta}^*$ (i.e. the mean point of the interval).

For either cut-HDMR or RS-HDMR, the final equation for the quantity of interest should be built as

$$E_y(\boldsymbol{\theta}) = E_{y0} + \sum_{i=1}^{d} E_{yi}\,(\theta_i) + \sum_{1\le i<j\le d} E_{yij}\left(\boldsymbol{\theta}_{ij}\right) + \cdots + E_{y12..d}(\boldsymbol{\theta}), \tag{17}$$

which, for cut-HDMR, follows

$$\begin{cases} \hat{\mathrm{E}}_{cut\ y0}(\boldsymbol{\theta}^*) = \dfrac{1}{N}\displaystyle\sum_{k=1}^{N} y^{(k)} \\ \hat{\mathrm{E}}_{cut\ yi}(\theta_i) = \dfrac{1}{N}\displaystyle\sum_{k=1}^{N} y^{(k)}\, r_{cut\ i}\left(\boldsymbol{x}^{(k)}\middle|\theta_i, \boldsymbol{\theta}^*\right) \\ \hat{\mathrm{E}}_{cut\ yij}\left(\theta_{ij}\right) = \dfrac{1}{N}\displaystyle\sum_{k=1}^{N} y^{(k)}\, r_{cut\ ij}\left(\boldsymbol{x}^{(k)}\middle|\theta_{ij}, \boldsymbol{\theta}^*\right) \end{cases}, \tag{18}$$

$$\begin{cases} r_{cut\,i}\left(\boldsymbol{x}^{(k)}\middle|\theta_i,\boldsymbol{\theta}^*\right) = \dfrac{f_X\left(\boldsymbol{x}^{(k)}\middle|\theta_i,\boldsymbol{\theta}^*_{-i}\right)}{f_X\left(\boldsymbol{x}^{(k)}\middle|\boldsymbol{\theta}^*\right)} \\ r_{cut\,ij}\left(\boldsymbol{x}^{(k)}\middle|\theta_{ij},\boldsymbol{\theta}^*\right) = \dfrac{f_X\left(\boldsymbol{x}^{(k)}\middle|\boldsymbol{\theta}_{ij},\boldsymbol{\theta}^*_{-ij}\right)}{f_X\left(\boldsymbol{x}^{(k)}\middle|\boldsymbol{\theta}^*\right)} - r_{cut\,i}\left(\boldsymbol{x}^{(k)}\middle|\theta_i,\boldsymbol{\theta}^*_{-i}\right) - r_{cut\,j}\left(\boldsymbol{x}^{(k)}\middle|\theta_j,\boldsymbol{\theta}^*_{-j}\right) \end{cases} \quad (19)$$

here, the indexes i and j may assume values between 1 and the total number of epistemic parameters. $\theta_i^{(k)}$ denotes the k-th sample point of θ_i, drawn from the joint samples. Similarly, $\boldsymbol{\theta}_{ij}^{(k)}$ represents the k-th sample point of $\boldsymbol{\theta}_{ij}$, also drawn from the joint samples. The term $\boldsymbol{\theta}_{ij}$ refers to the 2-dimensional vector composed of θ_i and θ_j, $\boldsymbol{\theta}$ is the vector of epistemic parameters, while $\boldsymbol{\theta}_{-i}$ denotes this same vector containing the epistemic parameters except for θ_i. $\hat{E}$ is the approximation for the quantity of interest, which in this case is the mean.

Other quantities of interests such as probability of failure and variance may be defined, and then, Equations 25 to 27 can be rewritten following Equation 28

$$\begin{cases} \hat{E}_y(\boldsymbol{\theta}) = \dfrac{1}{N}\sum_{k=1}^{N} \mathrm{CkT}^{(k)}\dfrac{f_X\left(\boldsymbol{x}^{(k)}\middle|\boldsymbol{\theta}\right)}{f_X\left(\boldsymbol{x}^{(k)}\middle|\boldsymbol{\theta}^*\right)} \\ \widehat{\mathbb{V}}_y(\boldsymbol{\theta}) = \dfrac{1}{N}\sum_{k=1}^{N} \mathrm{kCkT}^{(k)}\big) - E_y\big)^2\dfrac{f_X\left(\boldsymbol{x}^{(k)}\middle|\boldsymbol{\theta}\right)}{f_X\left(\boldsymbol{x}^{(k)}\middle|\boldsymbol{\theta}^*\right)} \\ \hat{P}(\boldsymbol{\theta}) = \dfrac{1}{N}\sum_{k=1}^{N} I_F\left(x^{(k)}\right)\dfrac{f_X\left(\boldsymbol{x}^{(k)}\middle|\boldsymbol{\theta}\right)}{f_X\left(\boldsymbol{x}^{(k)}\middle|\boldsymbol{\theta}^*\right)} \end{cases} \quad (20)$$

where I_F is the indicator function of failure expressed as $I_F(x) = 1$ if $x \in F$ (failure), and $I_F(x) = 0$ if $x \notin F$ (safety), $\hat{P}$ is the approximated failure probability, and $\widehat{S^2}_y$ is the approximation of the variance (second moment about the mean). For more details about cut-HDMR or complete equations for RS-HDMR, please refer to [17].

3.3 Confidence bounds and failure probability in dynamic problems using NISS and MCS

The application of NISS and MCS to static problems is straightforward. However, certain adjustments are required for dynamic problems. The proposed framework for calculating the confidence bounds involves the following steps:

1. Generate Time Signals: For each sample of the input variables, generate a time signal using deterministic simulation.
2. Group Solutions by Time Step: Organize the solutions from all time signals into groups corresponding to each time step, resulting in as many sets as there are time steps.
3. Process the data: Use each set of solutions as input for NISS (cut-HDMR or RS-HDMR) or the subsequent steps of Double-Loop MCS.

In the case of Double-Loop MCS, confidence bands (also called envelopes) can be constructed by determining the desired Probability Quantile (PQ) from the data collected at each time step. For the NISS-based approach, an additional hypothesis of normal distribution output is made. With that in mind, an optimization algorithm is needed to solve the following problems

$$Minimize\ f_{lb}(\boldsymbol{\theta}),$$
$$Subjected\ to\ \theta_{i\,min} \leq \theta_i \leq \theta_{i\,max}, i = 1, \ldots, p, \tag{21}$$

$$Maximize\ f_{ub}(\boldsymbol{\theta}),$$
$$Subjected\ to\ \theta_{i\,min} \leq \theta_i \leq \theta_{i\,max}, i = 1, \ldots, p, \tag{22}$$

here, p denotes the total number of epistemic parameters. For each epistemic parameter i, $\theta_{i\,min}$ and $\theta_{i\,max}$ represent the lower and upper limits, respectively. The functions $f_{lb}(\boldsymbol{\theta})$ and $f_{ub}(\boldsymbol{\theta})$ are defined as follows

$$f_{lb}(\boldsymbol{\theta}) = E_y(\boldsymbol{\theta}) - XS_y(\boldsymbol{\theta}), \tag{23}$$

$$f_{ub}(\boldsymbol{\theta}) = E_y(\boldsymbol{\theta}) + XS_y(\boldsymbol{\theta}). \tag{24}$$

In this context, E_y and S_y represent the output equations of the NISS method that approximate the mean and standard deviation, respectively, as detailed in Equations 17 to 20. These equations are computed in this study using the second-order cut-HDMR approach. For $X = 2$, the resulting bounds (envelopes) are expected to achieve approximately 95.45% confidence. The solutions to these minimization and maximization problems at each time step form the sets that constitute the final confidence bounds.

4 NUMERIC EXAMPLE: DISPLACEMENT CONFIDENCE BOUNDS

The dynamic structure simulated in this work consists of a Graphite AS4/Epoxy 3501-6 plate with a volume fraction of fibers $V_f = 0.6$. The laminate has a stacking sequence of $[30, -30, -30, 30]$ and is subjected to clamped-free-free-free boundary conditions.

The excitation force F is a sustained load applied at the tip of the plate described as normally distributed $F \sim \mathcal{N}(\mu_F, \sigma_F)$ with mean value of $\mu_F = F'[0.8, 1.2]\ N$ and standard deviation of $\sigma_F = F'[0.1, 0.2]\ N$, where $F' = -0.1\ N$. The longitudinal elastic modulus E_1 is also defined as normally distributed $E_1 \sim \mathcal{N}(\mu_{E_1}, \sigma_{E_1})$ with epistemic parameters mean $\mu_{E_1} = {E_1}'[0.8, 1.2]\ GPa$ and standard deviation $\sigma_{E_1} = {E_1}'[0.1, 0.2]\ GPa$. The value for ${E_1}'$ is derived using the rule of mixtures and properties from Table 2, resulting in $E_1' = 138.2\ GPa$. Thus, the vector of epistemic parameters is defined as $\boldsymbol{\theta} = [\mu_{E_1}, \mu_F, \sigma_{E_1}, \sigma_F]$.

The quantity of interest, that is the vertical displacement z, along with a schematic representation of the composite plate is illustrated in Figure 2.

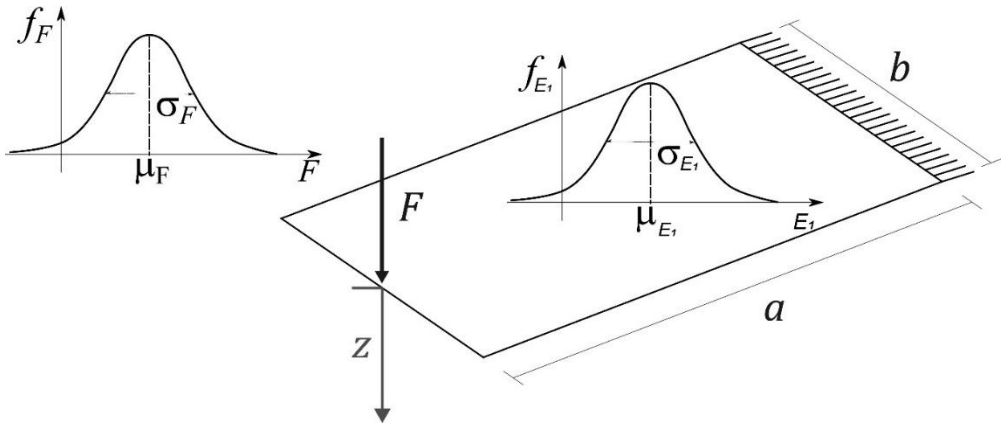


Figure 2: Composite plate diagram.

	$E\ [GPa]$	$\nu_{12}\ [-]$	$G_{12}\ [GPa]$	$\rho\ [kg/m^3]$
Fiber (Graphite AS4)	227.527	0.2	24.8211	1800
Matrix (Epoxy 3501-6)	4.2	0.340	1.567	1300

Table 2: Material Elastic properties extracted from MechG-Comp [25].

Newmark integration algorithm is applied to solve the finite element problem in time domain. The optimization algorithm used in the results presented in this work is the PSO [26] with the following parameters: number of particles, $n = 64$; momentum, $\omega = 0.95$; cognitive component 1 and 2, $C_1 = C_2 = 2.01$; tolerance, $tol = 10^{-6}$; maximum number of iterations, $nmax = 1{,}000$; random decay $\propto = 0.9$; and random perturbation, $\propto_t = 0.01$.

The final confidence bounds (envelopes) for the displacement can be seen in Figure 3, where the second order cut-HDMR was able to successfully capture the general behavior of the bound resulted by the Double Loop MCS. The total number of samples for the double-loop MCS was 800×800, while cut-HDMR was set to 2,000. Increasing the number of samples of NISS did not improve the solution, and the number of MCS samples was limited by the available computational resources.

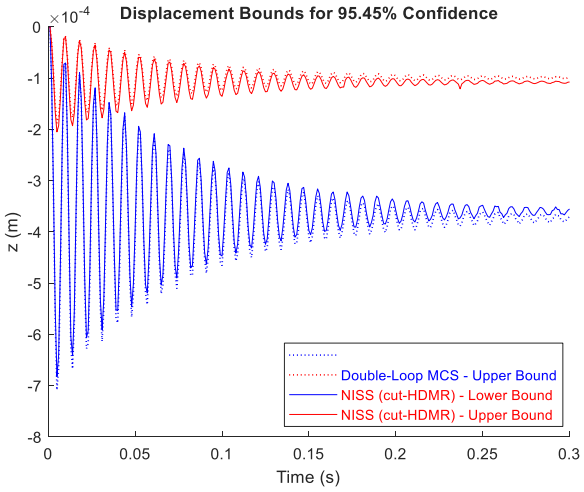


Figure 3: Displacement bounds for 95.45% confidence.

The RMS (root mean square) values of the time signals presented in Figure 3 can be checked in table 3, where the higher associated error found was 9.26% ($RMS_{Error} = (RMS_{\text{cut-HDMR}} - RMS_{MCS})/RMS_{MCS}$). This difference is attributed mainly to NISS precision and the normality hypothesis.

	RMS_{MCS} (m)	$RMS_{\text{cut-HDMR}}$ (m)	RMS_{Error} (%)
Upper Bound	0.1017×10^{-3}	0.1111×10^{-3}	9.26
Lower Bound	0.3838×10^{-3}	0.3695×10^{-3}	-3.72

Table 3: RMS values of the 95.45% confidence bounds found by Double-Loop Monte Carlo and NISS (second order cut-HDMR).

Figure 4 shows the design variables found by the optimization algorithm in each time step, revealing that some values of μ_{E_1} did not adhere to the maximum and minimum of the interval bound. While this behavior is atypical for static problems of the same nature, it can be attributed to the dynamic characteristics of the system. Variations in stiffness lead to shifts in the natural frequency of the system, resulting in intermediate values of μ_{E_1} as solutions to the optimization problem at specific time steps. Lastly, the confidence and behavior of each cut-HDMR in time domain for the mean $E(\boldsymbol{\theta})$ can be checked in Figure 5.

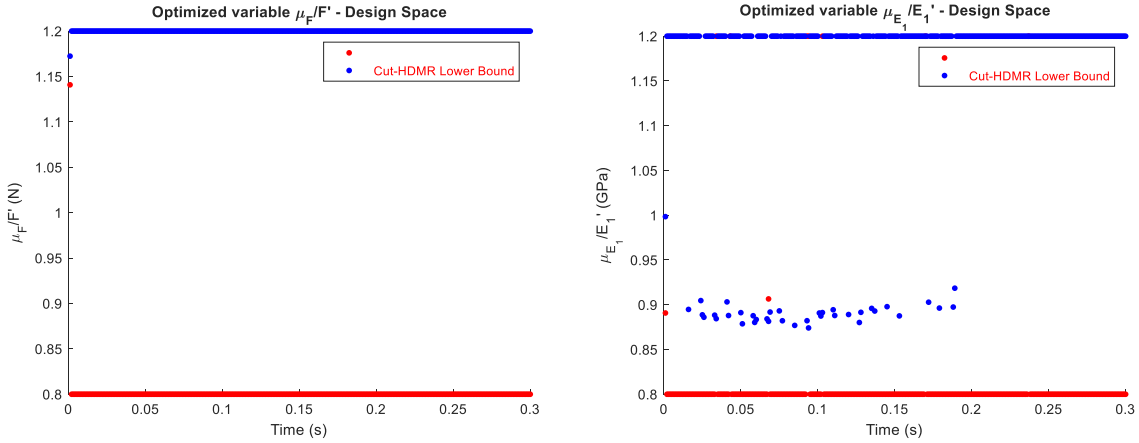


Figure 4: Optimized variables μ_{E_1} and μ_F in design space.

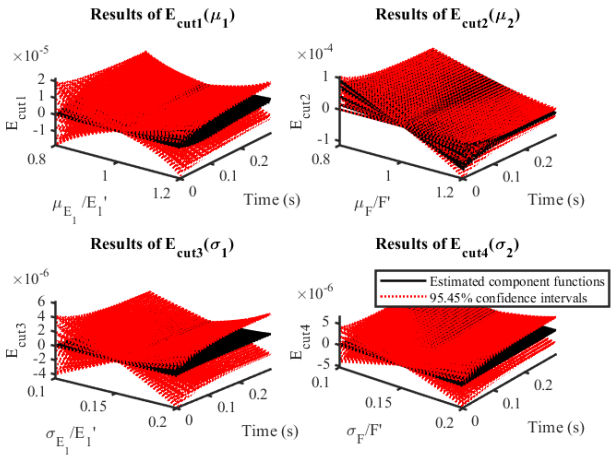


Figure 5: Confidence of each cut-HDMR component in time domain.

5 CONCLUSION

This study explores the confidence bounds and reliability problems in dynamic composite plates under epistemic uncertainties using NISS (cut-HDMR). The numerical example demonstrates the effectiveness of the presented framework employing cut-HDMR to estimate confidence bounds for vertical displacements in dynamic problems involving composite plates. Cut-HDMR successfully captures the overall behavior of the confidence bounds with great computational efficiency, requiring only 2,000 deterministic evaluations compared to 640,000 for double-loop MCS.

Future works may explore extending this framework to more complex structures and loading scenarios, incorporating advanced uncertainty modeling techniques. Additionally, further research could use the presented tools to investigate failure probability in composite plates.

FUNDING

This study is funded by CAPES/PROBRAL Project No.88887.895021/2023-00.

REFERENCES

[1] S. Dey, T. Mukhopadhyay, and S. Adhikari, *Uncertainty Quantification in Laminated Composites*. CRC Press, 2018.

[2] X. Li, Z. Lv, and Z. Qiu, A novel univariate method for mixed reliability evaluation of composite laminate with random and interval parameters, *Composite Structures*, vol. 203, pp. 153–163, Nov. 2018, doi: https://doi.org/10.1016/j.compstruct.2018.06.097.

[3] X.-Y. Zhou *et al.*, Bounds of mechanical properties of fibre reinforced polymer composites with hybrid random and interval uncertainties, *Thin-Walled Structures*, vol. 182, pp. 110158–110158, Oct. 2022, doi: https://doi.org/10.1016/j.tws.2022.110158.

[4] Matthias G.R. Faes, M. A. Valdebenito, D. Moens, and M. Beer, Operator norm theory as an efficient tool to propagate hybrid uncertainties and calculate imprecise probabilities, *Mechanical Systems and Signal Processing*, vol. 152, pp. 107482–107482, May 2021, doi: https://doi.org/10.1016/j.ymssp.2020.107482.

[5] D. Wu, A. Liu, Y. Huang, Y. Huang, Y. Pi, and W. Gao, Mathematical programming approach for uncertain linear elastic analysis of functionally graded porous structures with interval parameters, *Composites Part B Engineering*, vol. 152, pp. 282–291, Jun. 2018, doi: https://doi.org/10.1016/j.compositesb.2018.06.032.

[6] J. Zhu, Z. Lv, and H. Liu, A novel iterative algorithm for natural frequency analysis of FG thin plates under interval uncertainty, *Structural and Multidisciplinary Optimization*, vol. 60, no. 4, pp. 1389–1405, May 2019, doi: https://doi.org/10.1007/s00158-019-02267-x.

[7] L. Wang, Y. Liu, and Y. Liu, An inverse method for distributed dynamic load identification of structures with interval uncertainties, *Advances in Engineering Software*, vol. 131, pp. 77–89, May 2019, doi: https://doi.org/10.1016/j.advengsoft.2019.02.003.

[8] C. Xiong, L. Wang, G. Liu, and Q. Shi, An iterative dimension-by-dimension method for structural interval response prediction with multidimensional uncertain variables, *Aerospace Science and Technology*, vol. 86, pp. 572–581, Mar. 2019, doi: https://doi.org/10.1016/j.ast.2019.01.032.

[9] S. Dey, T. Mukhopadhyay, H. H. Khodaparast, and S. Adhikari, Fuzzy uncertainty propagation in composites using Gram–Schmidt polynomial chaos expansion, *Applied Mathematical Modelling*, vol. 40, no. 7–8, pp. 4412–4428, Apr. 2016, doi: https://doi.org/10.1016/j.apm.2015.11.038.

[10] M. D. F. Awruch and H. M. Gomes, A fuzzy α-cut optimization analysis for vibration control of laminated composite smart structures under uncertainties, *Applied Mathematical*

Modelling, vol. 54, pp. 551–566, Feb. 2018, doi: https://doi.org/10.1016/j.apm.2017.10.002.

[11] H. Lü, Wen-Bin Shangguan, and D. Yu, Uncertainty quantification of squeal instability under two fuzzy-interval cases, *Fuzzy Sets and Systems*, vol. 328, pp. 70–82, Jul. 2017, doi: https://doi.org/10.1016/j.fss.2017.07.006.

[12] H. Lü, Z. Cai, Q. Feng, Wen-Bin Shangguan, and D. Yu, An improved method for fuzzy–interval uncertainty analysis and its application in brake instability study, *Computer Methods in Applied Mechanics and Engineering*, vol. 342, pp. 142–160, Dec. 2018, doi: https://doi.org/10.1016/j.cma.2018.07.028.

[13] P. B. Santana, Ewerton Grotti, and H. M. Gomes, An Efficient Anti-Optimization Approach for Uncertainty Analysis in Composite Laminates, *Materials Research*, vol. 24, no. suppl 2, Jan. 2021, doi: https://doi.org/10.1590/1980-5373-mr-2021-0334.

[14] K. Zhou, Z. Wang, Q. Gao, S. Yuan, and J. Tang, Recent advances in uncertainty quantification in structural response characterization and system identification, *Probabilistic Engineering Mechanics*, vol. 74, pp. 103507–103507, Aug. 2023, doi: https://doi.org/10.1016/j.probengmech.2023.103507.

[15] R. Schöbi and B. Sudret, Global sensitivity analysis in the context of imprecise probabilities (p-boxes) using sparse polynomial chaos expansions, *Reliability Engineering & System Safety*, vol. 187, pp. 129–141, Jul. 2019, doi: https://doi.org/10.1016/j.ress.2018.11.021.

[16] H. Xie, J. Li, and D. Liao, A new structural reliability analysis method under non-parameterized probability box variables, *Structural and Multidisciplinary Optimization*, vol. 65, no. 11, Nov. 2022, doi: https://doi.org/10.1007/s00158-022-03408-5.

[17] P. Wei *et al.*, Non-intrusive stochastic analysis with parameterized imprecise probability models: I. Performance estimation, *Mechanical Systems and Signal Processing*, vol. 124, pp. 349–368, Jun. 2019, doi: https://doi.org/10.1016/j.ymssp.2019.01.058.

[18] P. Wei *et al.*, Non-intrusive stochastic analysis with parameterized imprecise probability models: II. Reliability and rare events analysis, vol. 126, pp. 227–247, Jul. 2019, doi: https://doi.org/10.1016/j.ymssp.2019.02.015.

[19] J. Song *et al.*, Generalization of non-intrusive imprecise stochastic simulation for mixed uncertain variables, *Mechanical Systems and Signal Processing*, vol. 134, pp. 106316–106316, Dec. 2019, doi: https://doi.org/10.1016/j.ymssp.2019.106316.

[20] J. Song, M. Valdebenito, P. Wei, M. Beer, and Z. Lu, Non-intrusive imprecise stochastic simulation by line sampling, *Structural Safety*, vol. 84, pp. 101936–101936, Feb. 2020, doi: https://doi.org/10.1016/j.strusafe.2020.101936.

[21] H. Zhao and C. Zhou, An imprecise multiscale uncertainty quantification framework for fiber reinforced composites, *Probabilistic Engineering Mechanics*, vol. 78, pp. 103686–103686, Sep. 2024, doi: https://doi.org/10.1016/j.probengmech.2024.103686.

[22] O.C. Zienkiewicz, R.C. Taylor, *The finite element method, Vol. I, 4th Edition*. McGraw Hill, 1989.

[23] A J M Ferreira and G M L Gladwell, *MATLAB Codes for Finite Element Analysis : Solids and Structures*. Dordrecht: Springer Netherlands, 2009.

[24] A. K. Kaw, *Mechanics of Composite Materials*. Boca Raton Crc Press, 2005.

[25] Grupo de Materiais Compósitos e Nano Compósitos, 2025. Mech G-comp, www.ufrgs.br/mechg.

[26] J. Kennedy and R. Eberhart, Particle swarm optimization, *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995, doi: https://doi.org/10.1109/icnn.1995.488968.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

CAPACITY BOUNDS OF UNREINFORCED MASONRY WALLS WITH INTERVAL MECHANICAL PROPERTIES

Alba Sofi[1], **Elyas Bayat**[1], **and Federica Tubino**[2]

[1] Department of Architecture and Design, University "Mediterranea" of Reggio Calabria, via dell'Università 25, 89124 Reggio Calabria, Italy
e-mail: {alba.sofi, elyas.bayat@unirc.it}

[2] Department of Civil, Chemical and Environmental Engineering Polytechnic School, University of Genoa, via Montallegro 1, 16145 Genoa, Italy
federica.tubino@unige.it

Abstract

The analysis of unreinforced masonry walls with uncertain mechanical properties is addressed. The interval model of uncertainty is adopted to deal with the limited availability of experimental data on masonry. Accordingly, the uncertain properties are modelled as interval variables with assigned lower bound and upper bound. Focusing on the in-plane behaviour of unreinforced masonry walls, a finite element model based on a simplified micro-modelling approach is developed in ABAQUS. The masonry wall shear capacity bounds are efficiently predicted by applying a ratio-of-polynomials response surface. A benchmark wall under in-plane monotonic loading is analysed to assess the accuracy and efficiency of the proposed approach.

Keywords: Unreinforced Masonry Walls, In-Plane Resistance, Simplified-Micro Modelling, ABAQUS, Interval Analysis, Response Surface Method.

1 INTRODUCTION

Extensive experimental, analytical, and numerical studies have been devoted to the prediction of the load-bearing capacity of unreinforced masonry (URM) walls (see e.g., [1],[2]). Finite Element (FE) approaches for analysing masonry structures are commonly classified into three main modelling strategies based on the treatment of units and mortar joints [3]-[6]: detailed micro-modelling, macro-modelling, and simplified micro-modelling. Detailed micro-modelling offers high accuracy but is computationally demanding and often impractical for engineering applications. In contrast, macro-modelling reduces computational effort but fails to accurately predict cracking patterns and failure modes. Simplified micro-modelling provides a good balance between computational efficiency and accuracy, making it a commonly used approach.

Reliable predictions from numerical simulations require an accurate definition of the mechanical properties of masonry. Such properties can vary in a wide range due to several sources of uncertainty relating to workmanship techniques, block shape and texture, properties of single components (mortar and bricks/stones), environmental conditions, past damage, deterioration, etc. The aim of this paper is to study the capacity of URM walls with uncertain mechanical properties under in-plane loading.

The traditional probabilistic model has been extensively adopted in the literature to predict the influence of uncertainties on the behaviour of masonry structures (see e.g., [7]-[11]). A major drawback is the limited availability of experimental data for the accurate probabilistic characterization of the uncertain mechanical properties of masonry, especially for historical buildings. To cope with this issue, the present study represents uncertainties as interval variables [12] with assigned Lower Bound (LB) and Upper Bound (UB), relying on the *Improved Interval Analysis* (*IIA*) via *Extra Unitary Interval* (*EUI*) [13]. The interval model requires only range information on the uncertain parameters, in line with the prescriptions provided by Building Codes and Guidelines (see e.g., [14]) as well as visual approaches for the mechanical characterization of masonry, such as the Masonry Quality Index (MQI) method [15].

Once the interval mechanical properties are incorporated into a FE model, the in-plane resistance of the masonry wall also exhibits an interval nature. The evaluation of the capacity bounds is a challenging task due to the nonlinear behaviour of the masonry wall and the high computational cost of deterministic FE analyses. Surrogate models (see e.g., [7],[11]) have been applied in the literature to efficiently analyse the ultimate capacity of masonry structures with random mechanical properties. The proposed approach for the propagation of interval uncertainties relies on the use of a ratio-of-polynomials Response Surface (RS), which requires only a few deterministic analyses at selected sampling points. The assumed RS was originally developed within the probabilistic framework [16] and later extended to linear problems with interval parameters [17],[18]. This study investigates the extension of the ratio-of-polynomials RS to the propagation of interval uncertainties in nonlinear problems, along with the influence of higher-order terms.

Focusing on URM walls under in-plane loading, a 3D FE model is developed in ABAQUS [19] based on a simplified micro-modelling approach and the Extended Finite Element Method (XFEM). A benchmark URM wall subjected to monotonic in-plane loading [20] is selected as a case study to assess the accuracy and efficiency of the proposed approach in predicting the bounds of the in-plane resistance and the associated failure modes in the presence of interval mechanical properties.

2 MASONRY WALL 3D FINITE ELEMENT MODEL

The 3D behaviour of unreinforced masonry (URM) walls is simulated using a simplified micro-modelling approach [4] implemented in the commercial Finite Element (FE) software ABAQUS [19]. This approach employs the Surface-Based Cohesive Behaviour (SBCB) and the Drucker-Prager (DP) plasticity models. Crack propagation within masonry units is simulated using the Extended Finite Element Method (XFEM), without the need to predefine crack locations.

The SBCB model captures failure mechanisms, such as tensile cracking and shear sliding of joints, based on surface interactions. The initial joint behaviour is described by a linear traction-separation relationship, characterized by stiffness constants depending on the elastic properties of mortar and brick units. Tensile cracking and shear sliding of joints are governed by their tensile and shear strengths, respectively. The calculation of joint shear strength relies on the Mohr-Coulomb failure criterion, which involves the cohesion, c, and coefficient of friction, μ. Damage initiation is assumed to occur when contact stresses satisfy the quadratic stress failure criterion [21]. After damage initiation, the reduction of the interface stiffness is assumed to follow a linear softening law.

The nonlinear compressive response of masonry units is simulated using the DP model. Specifically, a linear yield criterion is adopted, which requires the definition of the friction angle and flow stress ratio as well as the uniaxial compressive stress–strain curve (see, e.g., [22]) of masonry units.

For further details on the implementation and parameters of the models described above, readers are referred to the relevant literature (see, e.g., [3], [4], [19]).

3 UNCERTAIN MASONRY MECHANICAL PROPERTIES

A reliable FE analysis of masonry walls requires the accurate selection of the key mechanical properties, considering the various sources of uncertainty. Since only a limited amount of experimental data is typically available for existing masonry buildings, uncertainties can be appropriately described using the interval model [12], which relies solely on range information. The uncertain parameters are described as interval variables with assigned Lower Bound (LB) and Upper Bound (UB). To limit conservatism, typically affecting the *Classical Interval Analysis* (*CIA*) [12], the so-called *Improved Interval Analysis* via *Extra Unitary Interval* (*IIA* via *EUI*) [13] is adopted. Based on this approach, the $i-$th interval mechanical property is expressed in the following affine form:

$$d_i^I = \left[\underline{d}_i, \overline{d}_i\right] = d_{0,i}\left(1+\alpha_i^I\right) = d_{0,i}\left(1+\Delta\alpha_i \hat{e}_i^I\right) \tag{1}$$

where the superscript I indicates interval variables; $\underline{d}_i$ and $\overline{d}_i$ denote the LB and UB of the interval variable d_i^I, respectively; $d_{0,i}$ represents the nominal value; $\alpha_i^I = \left[\underline{\alpha}_i, \overline{\alpha}_i\right] \in \mathbb{IR}$ is the fluctuation around the nominal value, described by a dimensionless symmetric interval variable ($\overline{\alpha}_i = -\underline{\alpha}_i$) with deviation amplitude $\Delta\alpha_i$; $\mathbb{IR}$ is the set of all real interval numbers; $\hat{e}_i^I = \left[-1,1\right]$ is the *EUI* associated with the $i-$th interval variable. Based on Eq. (1), the interval variable d_i^I can be defined by its LB and UB or by the midpoint and normalized deviation amplitude:

$$d_{0,i} = \mathrm{mid}\left\{d_i^I\right\} = \frac{\underline{d}_i + \overline{d}_i}{2}; \qquad \Delta\alpha_i = \frac{\Delta d_i}{d_{0,i}} = \frac{\overline{d}_i - \underline{d}_i}{2d_{0,i}} \tag{2a,b}$$

where $\mathrm{mid}\{\bullet\}$ denotes the midpoint operator; Δd_i is the deviation amplitude of d_i^I, and $\Delta\alpha_i < 1$ to guarantee positive values of the uncertain mechanical properties.

The definition of these quantities is quite challenging as the values of masonry mechanical properties may vary significantly due to various factors, such as masonry typology, workmanship techniques, environmental conditions, etc. The analyst can rely on ranges of the mechanical properties recommended by Building Codes and Guidelines (see, e.g., [14]), visual approaches such as the Masonry Quality Index (MQI) method [15], or experimental data, if available.

4 CAPACITY BOUNDS VIA RESPONSE SURFACE APPROACH

Let us assume that N mechanical properties of the FE modelled URM wall are described as interval variables d_i^I, $i = 1,2,\ldots,N$, whose fluctuations are collected into the interval vector $\boldsymbol{\alpha}^I = \left[\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}\right] \in \mathbb{IR}^N$. All possible responses obtained as the uncertain parameters vary within their interval define a solution set, which is typically approximated by the smallest conservative hypercube. Focusing on the generic response quantity, $y^I = f(\boldsymbol{\alpha}^I)$, with $f : \mathbb{R}^N \to \mathbb{R}$ denoting a deterministic mapping, the LB and UB are formally defined as:

$$\underline{y} = \min_{\boldsymbol{\alpha}\in\boldsymbol{\alpha}^I} \{f(\boldsymbol{\alpha})\}; \qquad \overline{y} = \max_{\boldsymbol{\alpha}\in\boldsymbol{\alpha}^I} \{f(\boldsymbol{\alpha})\}. \tag{3a,b}$$

Such bounds can be evaluated by solving a global optimization problem which requires repeated deterministic FE analyses of the masonry wall, resulting in a high computational burden. Surrogate models are often used in conjunction with global optimization to reduce the computational cost associated with the evaluation of the interval response bounds.

The uncertainty propagation strategy proposed in this study relies on the use of a ratio-of-polynomials RS, which can be defined by performing a certain number of deterministic FE analyses.

Since the deterministic nonlinear static analysis of the URM wall is performed using an incremental procedure associated with the Newton-Raphson method, the RS is determined at each load increment. As a first step, the following expression for the generic interval response quantity at each load increment, $y^I(\boldsymbol{\alpha})$, is assumed:

$$y^I \equiv y(\boldsymbol{\alpha}^I) = y_0 + \sum_{i=1}^{N} y_i(\alpha_i^I) \tag{4}$$

where y_0 denotes the nominal value, i.e., the response quantity estimated for $\boldsymbol{\alpha} = \mathbf{0}$ and $y_i(\alpha_i^I)$ is the deviation due to the $i-$th uncertain parameter, evaluated setting $\alpha_j^I = 0$ for $j \neq i$. To simplify notation, the load step is omitted. By interval extension of the RS proposed in Ref. [16], the deviation $y_i(\alpha_i^I)$ associated with the $i-$th uncertain parameter is approximated as a rational function of the fluctuation α_i^I, so that Eq. (4) reads:

$$y^I(\boldsymbol{\alpha}) = y_0 + \sum_{i=1}^{N}\sum_{k=1}^{s} \frac{\alpha_i^I}{A_i^{(k)} + B_i^{(k)}\alpha_i^I} = y_0 + \sum_{i=1}^{N}\sum_{k=1}^{s} \frac{\Delta\alpha_i \hat{e}_i^I}{A_i^{(k)} + B_i^{(k)}\Delta\alpha_i \hat{e}_i^I} \tag{5}$$

where s is the number of natural deformation modes of the relevant FE affected by the $i-$th fluctuation α_i^I; $A_i^{(k)}$ and $B_i^{(k)}$ are $2N \times s$ unknown coefficients, which can be evaluated by fitting the approximate solution to the actual response at selected sampling points. By applying a *saturated design*, $1 + 2s \times N$ deterministic analyses are needed, one to compute the nom-

inal solution y_0 and $2s \times N$ to evaluate the "exact" implicit response, $y(\boldsymbol{\alpha}^I)$ for $2N \times s$ arbitrary realizations of the interval parameters $\alpha_i^{(j)} \in \alpha_i^I$ ($i = 1,2,\ldots,N$; $j = 1,2,\ldots,2s$). It is worth mentioning that the coefficients of the ratio-of-polynomials RS are almost insensitive to the selection of the sampling points [16]. Once the coefficients $A_i^{(k)}$ and $B_i^{(k)}$ are evaluated, Eq. (5) can replace the time-consuming numerical model of the masonry wall, allowing the efficient determination of the LB and UB of the response quantity of interest (Eqs. (3a,b)).

5 NUMERICAL APPLICATION

A well-known experimental benchmark masonry wall [20] is selected as a case study to validate the proposed procedure (see Fig. 1). The URM wall comprised 18 courses of solid clay bricks bonded with a 10 mm-thick mortar. The upper and lower courses were secured to steel beams using high-performance mortar. The effective dimensions of the wall were 990 mm in length, 1000 mm in height, and 100 mm in thickness. Two identical specimens, labeled J4D and J5D, were tested under a uniform vertical pressure of 0.3 MPa. Before applying a 4 mm horizontal displacement, the vertical movement and rotation of the upper steel beam were constrained. To simulate the behaviour of the benchmark masonry wall, a FE model is built in ABAQUS using the simplified micro-modelling approach described in Section 2. 3D hexahedral elements with eight nodes, reduced integration, and hourglass control (C3D8R) [19] are used to model the expanded units. A mesh, consisting of 7×3×2 elements per masonry unit, resulting in 4520 elements, is adopted. The mechanical properties used to perform numerical simulations are taken from the experimental findings [20], as well as the data available in the literature [4]. Nonlinear static analysis is carried out by applying displacement control incrementally, using the Newton-Raphson algorithm. Automatic load stepping with specified increments up to a maximum of 10^5 is employed. A viscosity parameter of 0.002 is implemented for regularization to ensure convergence during stiffness degradation in the masonry joints.

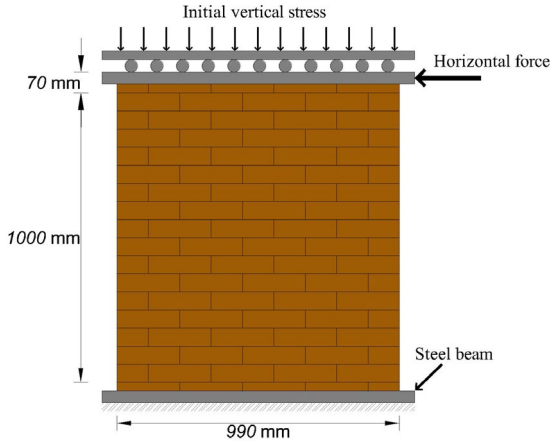


Figure 1: Test setup for the URM wall under confined shear loading (based on [20]).

As a first step, the FE model built in ABAQUS is validated by comparison with experimental results and numerical simulations available in the literature [4]. The load-displacement curves plotted in Fig. 2 demonstrate the ability of the implemented FE model to accurately capture the in-plane behaviour of the masonry wall.

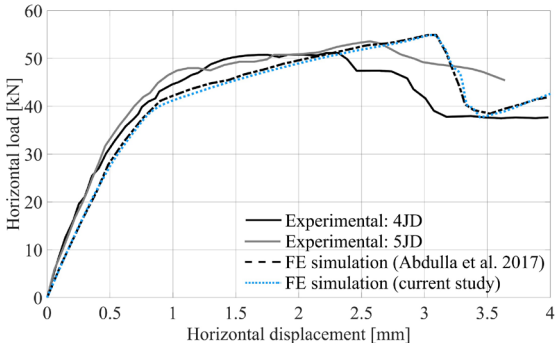
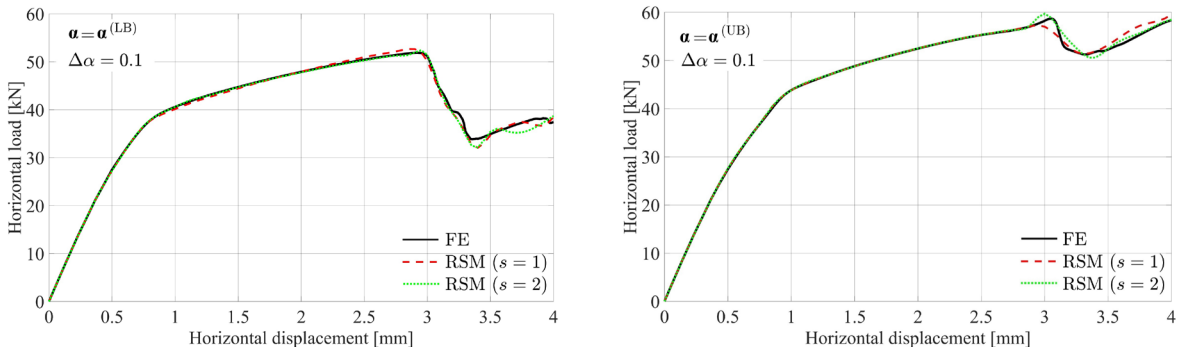


Figure 2: Load-displacement curves from FE simulations and experimental results for the benchmark URM wall under in-plane loading.

Given their significant impact on the in-plane behaviour of URM walls [5], the cohesion, c, and the coefficient of friction, μ, of the masonry joint interfaces are assumed uncertain. Both parameters are allowed to vary independently across two identical horizontal portions of the wall, each having a height of 500 mm. As a result, four interval parameters are involved, c_b^I, μ_b^I, c_t^I, and μ_t^I, representing the cohesion and coefficient of friction in the bottom (b) and top portions (t) of the URM wall. The nominal values of the uncertain parameters are assumed to be the same for the two portions of the wall and equal to those of the benchmark wall. The dimensionless fluctuations of the uncertain parameters around their nominal value, α_i^I, $i = c_b, \mu_b, c_t, \mu_t$, characterized by the same deviation amplitude $\Delta\alpha$, are collected into the interval vector $\boldsymbol{\alpha}^I = \left[\alpha_{c_b}^I \quad \alpha_{\mu_b}^I \quad \alpha_{c_t}^I \quad \alpha_{\mu_t}^I\right]^{\mathrm{T}}$.

Approximate estimates of the LB and UB of the masonry wall in-plane resistance are determined by applying the ratio-of-polynomials RS in conjunction with a combinatorial procedure. As a first step, the problem is discretized by subdividing the range of the interval parameters into smaller subintervals. Then, deterministic analyses are performed for all possible combinations of the associated endpoints, and the minimum and maximum among the obtained capacity samples are identified. For instance, subdividing the range of each interval parameter into ten subintervals and applying the ratio-of-polynomials RS (Eq. (5)) with $s = 2$, the bounds of the in-plane resistance of the masonry wall with $\Delta\alpha = 0.1$ are achieved for the following combinations of the uncertain parameters: $\boldsymbol{\alpha}^{(\mathrm{LB})} = \left[-0.1 \quad -0.1 \quad 0.1 \quad -0.1\right]^{\mathrm{T}}$ and $\boldsymbol{\alpha}^{(\mathrm{UB})} = \left[0.1 \quad 0.1 \quad 0.1 \quad 0.1\right]^{\mathrm{T}}$.

For validation purposes, Fig. 3 shows the comparison between the load-displacement curves obtained by FE analysis and the proposed Response Surface Method (RSM) with $s = 1$ and $s = 2$ for the realizations of the interval parameters $\boldsymbol{\alpha}^{(\mathrm{LB})}$ and $\boldsymbol{\alpha}^{(\mathrm{UB})}$. A very good agreement can be observed, even when $s = 1$.



(a) (b)

Figure 3: Load-displacement curves provided by FE analysis and RSM for the realizations of the uncertain parameters yielding the (a) LB and (b) UB of the shear capacity ($\Delta\alpha = 0.1$).

Table 1 lists the LB and UB of the masonry wall in-plane resistance predicted by FE analysis and RSM (with $s = 1$ and $s = 2$), R_{FE} and R_{RSM}, for two different values of the normalized deviation amplitude $\Delta\alpha$, along with the associated percentage error $\varepsilon(\%) = ((R_{RSM} - R_{FE}) / R_{FE}) \times 100$. It can be observed that the RS yields very accurate estimates of the bounds of the capacity, even when $s = 1$ and larger degrees of uncertainty ($\Delta\alpha = 0.2$) are considered. The percentage errors for $\Delta\alpha = 0.2$ are less than 3.5% and 2.5% for $s = 1$ and $s = 2$, respectively.

	LB					UB				
$\Delta\alpha$	R_{FE} [kN]	R_{RSM} [kN] s=1	ε(%) s=1	R_{RSM} [kN] s=2	ε(%) s=2	R_{FE} [kN]	R_{RSM} [kN] s=1	ε(%) s=1	R_{RSM} [kN] s=2	ε(%) s=2
0.10	51.88	52.58	1.35	52.38	0.96	58.71	57.19	-2.59	59.70	1.69
0.20	48.62	49.54	1.90	49.23	1.27	62.61	60.49	-3.38	61.18	-2.27

Table 1: LB and UB of the masonry wall in-plane resistance provided by FE analysis and RSM, and associated percentage errors.

6 CONCLUSIONS

An efficient approach for analysing the in-plane behaviour of unreinforced masonry walls with uncertain mechanical properties has been presented. Given the limited availability of experimental data on masonry, uncertainties have been described as interval variables, fully characterized by their lower bound and upper bound. The masonry wall behaviour has been simulated by a 3D Finite Element (FE) model built in ABAQUS, based on a simplified micro-modelling approach. Uncertainty propagation has been efficiently performed by applying a ratio-of-polynomials Response Surface (RS), which requires only a few deterministic FE analyses at selected sampling points. Different orders of the RS and degrees of uncertainty have been investigated. Numerical results on a benchmark masonry wall under monotonic in-plane loading have demonstrated the accuracy of the proposed approach in predicting the bounds of the interval capacity.

Acknowledgements

The support of the following research grant is gratefully acknowledged
NGEU PNRR– MUR - Call for tender for the presentation of proposals for the Strengthening of research structures and creation of R&D "innovation ecosystems", set up of "territorial leaders in R&D" — to be funded under the National Recovery and Resilience Plan (NRRP), Mission 4, Component 2 Investment 1.4, funded from the European Union - NextGenerationEU.
PROPOSAL: Tech4You - Technologies for climate change adaptation and quality of life improvement.
NRRP THEMATIC AREA: CLIMATE, ENERGY, SUSTAINABLE MOBILITY TERRITORY OF REFERENCE: Calabrian and Lucanian territorial areas.
Spoke 4 - Technologies for resilient and accessible cultural and natural heritage: Unibas Affiliates: Unical, Unirc, CNR - Goal 4.7: Protection and enhancement of natural and cultural heritage and identity of the inner territories.
Pilot Project 4.7.1 - Open platform "phigital space" (physical and digital) of the type "user profiling" for the advanced and dynamic codesign of interventions on the built and ex novo.

REFERENCES

[1] T. Celano, C. Modena, A. Bolognese, E. Lippiello, E. Rizzo, A. Sorrentino, S. Vassallo, Literature review of the in-plane behavior of masonry walls: Theoretical vs. experimental results. *Materials*, **14**(11), 3063, 2021.
[2] B. Doran, M.C. Griffith, R. Santos, S. Varga, Y. Zhang, J. Wang, Z. Chen, Experimental and numerical investigation of unreinforced masonry walls with and without opening. *International Journal of Architectural Heritage*, **17**(11), 1833–1854, 2023.
[3] P.J.B. Lourenço, *Computational strategies for masonry structures*. Doctoral thesis, Civil Engineering and Geosciences (CEG), TU Delft, 1997.
[4] K.F. Abdulla, L.S. Cunningham, M. Gillie, Simulating masonry wall behaviour using a simplified micro-model approach. *Engineering Structures*, **151**, 349–365, 2017.
[5] B. Zeng, Y. Li, C. Cruz Noguez, Modeling and parameter importance investigation for simulating in-plane and out-of-plane behaviors of un-reinforced masonry walls. *Engineering Structures*, **248**, 113233, 2021.
[6] K. Naciri, I. Aalil, A. Chaaba, M. Al-Mukhtar, Detailed micromodeling and multiscale modeling of masonry under confined shear and compressive loading. *Practice Periodical on Structural Design and Construction*, **26**(1), 04020056, 2021.
[7] D. Benasciutti, G. Milani, Limit analysis of masonry structures accounting for uncertainties in constituent materials mechanical properties. *The Open Civil Engineering Journal*, **2**, 51-57, 2008.
[8] J. Vaculik, M.C. Griffith, Probabilistic analysis of unreinforced brick masonry walls subjected to horizontal bending. *Journal of Engineering Mechanics*, **143**(8), 04017056, 2017.
[9] L. Iannacone, M. Andreini, P. Gardoni, M. Sassu, Probabilistic models and fragility estimates for unreinforced masonry walls subject to in-plane horizontal forces. *Journal of Structural Engineering*, **147**(6), 04021074, 2021.
[10] H. Tariq, M.A. Najafgholipour, V. Sarhosis, G. Milani, In-plane strength of masonry wall panels: A comparison between design codes and high-fidelity models. *Structures*, **47**, 1869-1899, 2023.

[11] B. Zeng, C. Cruz-Noguez, Y. Li, Enhanced prediction and uncertainty analysis for in-plane and out-of-plane resistance of unreinforced masonry walls: a multifidelity approach. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, **10**(4), 04024053, 2024.
[12] R.E. Moore, R.B. Kearfott, M.J. Cloud, *Introduction to Interval Analysis*. Society for Industrial and Applied Mathematics: Philadelphia, PA, USA; 2009.
[13] G. Muscolino, A. Sofi, Stochastic analysis of structures with uncertain-but-bounded parameters via improved interval analysis. *Probabilistic Engineering Mechanics*, **28**, 152-163, 2012.
[14] Ministero delle Infrastrutture e dei Trasporti, *Istruzioni per l'applicazione dell'aggiornamento delle 'Norme Tecniche per le Costruzioni' di cui al DM 17/01/2018*. Ministero delle Infrastrutture e dei Trasporti, Rome, Italy, 2019. Circolare 21 gennaio 2019, pubblicata nel Supplemento Ordinario alla Gazzetta Ufficiale n. 35 dell'11 febbraio 2019. (In Italian).
[15] A. Borri, M. Corradi, G. Castori, A. De Maria, A method for the analysis and classification of historic masonry. *Bulletin of Earthquake Engineering*, **13**(9), 2647-2665, 2015.
[16] N. Impollonia, A. Sofi, A response surface approach for the static analysis of stochastic structures with geometrical nonlinearities. *Computer Methods in Applied Mechanics and Engineering*, **192**(37), 4109-4129, 2003.
[17] A. Sofi, E. Romeo, A unified response surface framework for the interval and stochastic finite element analysis of structures with uncertain parameters. *Probabilistic Engineering Mechanics*, **54**, 25-36, 2018.
[18] A. Sofi, E. Romeo, O. Barrera, A. Cocks, An interval finite element method for the analysis of structures with spatially varying uncertainties. *Advances in Engineering Software*, **128**, 1-19, 2019.
[19] *Abaqus Analysis User's Manual, Version 2019*. Dassault Systèmes Simulia Corp., Providence, RI, USA, 2019. Available from: https://www.3ds.com/support/documentation.
[20] T.M.J. Raijmakers, A.T. Vermeltfoort, *Deformation controlled meso-shear tests on masonry*. Centre for Building Research, TNO-TUE, 1992. (In Dutch).
[21] M.L. Benzeggagh, M. Kenane, Measurement of mixed-mode delamination fracture toughness of unidirectional glass/epoxy composites with mixed-mode bending apparatus. *Composites Science and Technology*, **56**(4), 439-449, 1996.
[22] P.B. Shing, J.L. Noland, E. Klamerus, H. Spaeh, Inelastic behavior of concrete masonry shear walls. *Journal of Structural Engineering*, **115**(9), 2204-2225, 1989.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

LEVERAGING PHYSICS-INFORMED MODELS AND PRODUCTION DATA FOR QUALITY ASSESSMENT IN RESISTANCE SPOT WELDING

Miriam B. Dodt[1,2], Bouwe Verkens[1,2], Augustin Persoons[3], Matthias G.R. Faes[4], David Moens[1,2]

[1]KU Leuven, LMSD Division, Department of Mechanical Engineering
Jan Pieter de Nayerlaan 5, 2860 Sint-Katelijne-Waver, Belgium
e-mail: {miriam.dodt,bouwe.verkens,david.moens}@kuleuven.be

[2] Flanders Make @ KU Leuven, Belgium

[3] Université de Technologie Troyes
12 rue Marie Curie, CS 42060, 10004 Troyes CEDEX , France
e-mail: augustin.persoons@utt.fr

[4] TU Dortmund University, Chair for Reliability Engineering
Leonhard-Euler-Strasse 5, 44227 Dortmund, Germany
e-mail: matthias.faes@tu-dortmund.de

Abstract. *Ensuring high-quality and reliable production is a central challenge in industrial manufacturing. Within a production line, three main sources of knowledge are typically available: the general process settings, measurements taken during production, and post-process quality assessments. While quality assessments are essential for building accurate models and enabling process control, they are often costly and limited in availability. It is therefore important to choose a modelling approach that can handle limited data availability while also accounting for uncertainty and measurement noise. Some manufacturing processes are also subject to degradation, which must be properly accounted for in any predictive or control-oriented modelling approach.*
This paper investigates how modelling electrode degradation affects the quality prediction of resistance spot welds. Using an industrial dataset, various single- and multi-fidelity approaches, with and without an explicit drift model were developed and compared to each other. Special focus was placed on whether explicitly modelling process drift improves predictive performance. Results show that multi-fidelity models outperform single-fidelity ones, especially when degradation is explicitly modelled. Including a separate degradation model leads to a significantly better fit than models that rely only on electrode metadata or ignore drift altogether.

Keywords: Multi-fidelity Modelling, Resistance Spot Welding, Process Monitoring, Degradation Model, Surrogate Modelling.

1 INTRODUCTION

Resistance spot welding is a crucial production process in the manufacturing of cars and batteries [1], which is strongly influenced by both random variation and the gradual degradation of consumables, such as electrode caps. [2] To iterate, a modern vehicle consists of 2000 to 5000 individual spot welds, with a single production line producing more than seven million welds a day [1]. Reliable process control and monitoring require careful consideration of subtle changes in material properties, electrode conditions, environmental factors, and the wear state of consumables. This paper explores the capabilities of utilising a degradation model on the predictability of an output quantity. In this context, we develop and compare data-driven single- and multi-fidelity models using an industrial dataset. It is further explored, how using a multi-fidelity model improves predictability by combining different types of information. Multi-fidelity modelling has already been explored in related techniques, such as laser-bed welding [3, 4] but not yet in resistance spot welding. Multi-fidelity techniques as well as digital twin approaches, relevant for reliable process control, have been frequently tested on analytical cases or small-scale experimental set-ups. [5, 6] Evaluations on industrial datasets from a running production line, like in this paper, are seldom conducted. Information considered are the weld settings, which give a general expected weld quality, as well as measurements during the process. These measurements are cheap to acquire and provide information about each individual weld. To that front, the current and voltage is measured while welding, which allows to compute the dynamic resistance, one of the key parameters of resistance welding. In an industrial context, a weld is required to fulfil various quality criterion, such as a sufficient nugget diameter and indentation as well as passing visual inspection. Further detail about the criteria are found in the next section. In this paper, the indentation or rather is inverse, the remaining sheet thickness, are used as a quality assessor.

Additional to the multi-fidelity approach, a separate model is used to model the degradation. This allows us to improve our predictability and also estimate the effect of the degradation already before welding.

1.1 Resistance spot welding

During the resistance spot welding process, two to three thin metal sheets (often steel or aluminium) are joined through a localised spot. As a first step, the sheets are pressed together by two electrodes on opposing ends with a large force (range in kilonewton) during which a high electrical current (range in kiloampere) is sent through for a short amount of time (tens to hundreds of milliseconds). The thermal energy Q is generated following the equation of Joule heating $Q = I^2Rt$, with current I, resistance R and current flow duration t. [7] As heat is generated proportional to the local resistance, the resistance should be high where the weld should be formed and as small as possible elsewhere. Fig. 1 shows a schematic of the different local resistances, that affect the heat generation. [8] The total resistance is split in two types, bulk resistance of the metal R_m and contact resistance between two surfaces R_c. To minimise energy loss, the contact resistance between the two sheets $R_{c,3}$ should be by far the largest, whereas the resistance between the electrodes and the metal sheets $R_{c,1}$ and $R_{c,2}$ should be as small as possible. Copper, which has good thermal and electrical conductivity, is therefore usually chosen as the electrode material.

When sufficient heat is generated between the sheets, the material begins to melt, forming a pool of molten material. After re-solidification this pool creates the welding nugget. During

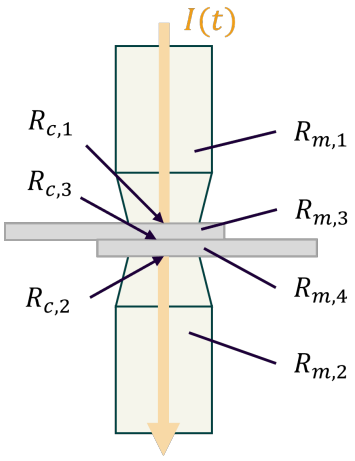


Figure 1: Schematic of contact resistance $R_{c,1-3}$ and material resistance $R_{m,1-4}$ of a two-sheet RSW-process. Also visible are the electrodes, metal sheets and current flow.

welding, both the current $I(t)$ and voltage $V(t)$ flow can be measured, allowing calculating the dynamic resistance $R(t)$ using Ohm's law $R(t) = V(t)/I(t)$. Fig. 2 [8] shows a typical resistance curve observed during the welding process. In phase 1, the sheets are pressed together by the electrodes to establish optimal contact. The resistance in that phase is concentrated at the micro-contacts and is initially high due to debris, dirt and oil on the workpieces surface acting as an insulator. The sharp drop in resistance occurs as the current begins to burn off these contaminants. In phase 2, most of the contaminants are removed, establishing metal-to-metal contact. As temperature rises, the material softens, increasing the contact area. While higher temperatures tend to increase resistance, the growing contact area has the opposite effect. In this phase, the effect of increasing contact area dominates, causing a further reduction of resistance. In phase 3, the resistance due to increased temperature becomes dominant. The temperature continues to raise until the onset of first melting near the end of this phase. In phase 4, melting progresses, expanding the molten zone, which increases the cross-sectional area, reducing the resistance. The peak resistance R_{max} corresponds to the point where temperature stabilises and the growth of the weld nugget becomes the predominant cause of resistance. [8] When current flow is continued after this point, the nugget will continue growing, until the surrounding solid material, under the large compressive electrode force, is insufficient to contain the molten material, it is being spat out, which is called expulsion. This expulsion can also be observed in the resistance curve by a sharp drop. [9]
To conclude, the dynamic resistance curve is a key feature of the welding process and its maximal value R_{max} and the respective time $t_{R_{max}}$ are two key parameters.

Fig. 3 shows the cross-section of a welding nugget, formed between two sheets with thickness t_1 and t_2 respectively. The nugget diameter w_n is visible, as well as the indentations $t_{i,1}$ and $t_{i,2}$ left by the electrode cap on both sides of the sheet. The weld quality is determined by its yield strength, which can only be tested destructively. It however directly corresponds to the size of the welding nugget w_n, which can also be estimated in a non-intrusive manner. This parameter alone is however not sufficient alone to determine the quality. Other factors such as the visual performance or indentation are relevant also. Visual performance refers to welds that appear "burnt" or show other visible defects. These defects do not directly affect the weld's

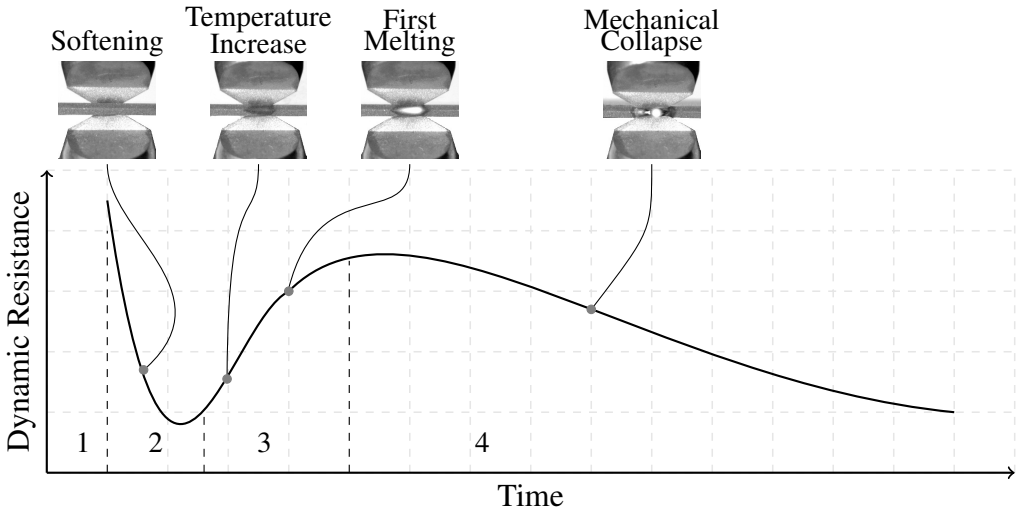


Figure 2: Theoretical dynamic resistance curve interpretation and characterisation. The images on top were created with a Phantom VEO 640 ultra high speed camera and illustrate the evolution of the weld nugget at the given phase. [8]

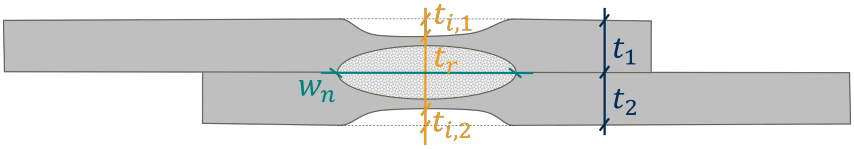


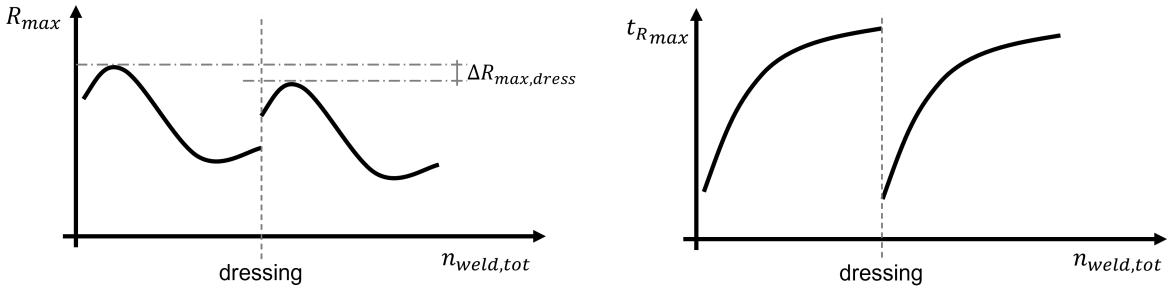
Figure 3: Schematic of the cross section of a weld. Indicated are sheet thickness ($t = t_1 + t_2 = t_{i,1} + t_r + t_{i,2}$), indentation ($t_i = t_{i,1} + t_{i,2}$) and remaining sheet thickness t_r as well as the welding nugget with width w_n.

strength but rather the surface finish and are visually unappealing to the costumer. During welding, the sheets melt and their contact point and the surrounding material softens. As the sheets are constantly pressed together with large force, the material deforms and the electrode leaves an indentation after cooling. The indentation is an indicator for the amount of material molten during welding and the corresponding yield strength. If the indentation is too small ($< 2-3\%$), there is the danger of a stick weld: the two sheets are not unified, rather just glued together, the weld's strength will not be sufficient. If the indentation is too large ($> 60\%$ of the thinnest sheet), the strength of the boundary between weld and sheet is low and can tear easily. The listed values are based on the standards DVS 2960 [10] and EN ISO 6520 [11] and contain a safety factor.

While the electrodes are manufactured to reduce their resistance as much as possible, heat generation inside the electrodes and especially at their contact to the metal sheets cannot be avoided completely. Over time, this leads to change of the electrode shape and thus the contact area and contact resistance. Additionally, dirt, oil and dust on the metal sheet can adhered to the electrode, further increasing the resistance and heat generation at the wrong contact surface. To decrease costs and increase electrode lifetime, the electrode is equipped with an electrode cap, made out of the same material. Instead of exchanging the full electrode after degradation, it is sufficient to exchange only the cap. Additionally, it is possible to dress the electrode a number of times before exchanging it. This entails taking away a small layer of material to restore the shape and surface quality.

1.2 Effect of consumables on the process

The degradation of the surface quality and the change of the contact area affect the dynamic resistance observed during the welding process. Based on expert knowledge and physical considerations, one can expect the maximal resistance R_{max} to increase the more often an electrode cap has welded. The time at which R_{max} is observed, $t_{R_{max}}$, also increases with each additional weld. The dynamic resistance describes the total resistance, including all seven resistances shown in fig. 1. As mentioned before, the dressing resets the surface finish and shape of the electrode, therefore resetting $t_{R_{max}}$. However, the thickness of the electrode cap reduces each time material is taken away. Although slowly, this introduces an additional facette of the drift, leading to a reduction in maximal observed resistance in each new dress cycle. In other words, while the time the maximal resistance is observed returns to its initial value after dressing, the maximal resistance is lower than in the previous dress cycle. Fig. 4 shows a schematic of the expected drift behaviour.



(a) Maximal observed dynamic resistance.

(b) Time of maximal resistance.

Figure 4: Schematic of the effect of electrode degradation on the observed values of R_{max} (a) and $t_{R_{max}}$ (b). Includes degradation within one dress cycle and between dresses. $n_{weld,tot}$ is the number of total welds performed.

It is important to note that welding is a highly sensitive and noisy process, influenced by the individual characteristics of each weld, local material variations, and other factors. Additionally, process control is not perfectly repeatable, and measurement errors in current and voltage are to be expected. Since dynamic resistance is computed from these measurements, it is inherently subject to the same uncertainties and variations.
Another important consideration is the condition of the electrode. Each weld affects the electrode's state, and electrode degradation is influenced not only by the weld history but also by the specific machine in use. As a result, the drift behaviour can vary significantly between different electrodes.

1.3 Motivation

To summarise, resistance spot welding is a fast manufacturing process, which is affected by large uncertainties as well as a periodic degradation. In this paper, it is explored how to model such a process to enable reliable and robust predictions for use in process control, monitoring, and, eventually, in building a digital twin. One important aspect to consider is, that the measured output quantity is subject to measurement error and is therefore noisy. The chosen modelling approach needs to take this into account. In cases where the noise is unbiased, regression methods are well suited to filter out the noise.

To address the challenges of noise, reliability and fast evaluation, grey-box and multi-fidelity modelling are used. Grey-box modelling unifies fast-to-evaluate data-based regression models, the so-called black-box models, as well as physics-informed white-box models. [12, 13] White-box models are usually numerical or analytical models, which makes them slow to evaluate for complex systems but more accurate in areas without sufficient data. [13] For processes with large variability, they are often not sufficiently accurate, because it is difficult to adjust them to each process individually due to lack of knowledge and cost restraints. To mitigate this, a multi-fidelity approach is used, which consists of a number of low- and high-fidelity models combined through a model management strategy. [14] Low-fidelity refers to models, that are less accurate, e.g. due to their simplicity, lack of complexity but also non-converged models. This makes them more cost efficient. High-fidelity models have a good accuracy but also significant costs attached. Combining both types allows for a resulting model with larger accuracy while limiting the costs. [6]
In our case, the low-fidelity model uses weld settings to predict the expected output, specifically the remaining sheet thickness t_r. However, it lacks any information about individual welds and real-time process variations, so its predictions can only reflect the average expectation. The high-fidelity model on the other hand incorporates process data, such as the dynamic resistance, voltage and current curve, which gives access to the actual conditions of individual welds. This makes the underlying model more precise and reliable but also requires a black-box model instead of a numerical model. To train this black-box model, measurements of the remaining sheet thickness are required, which are costly and resource-intensive to collect in large quantities, as pointed out earlier.

Traditionally, a parallel approach is used when designing a model management strategy, meaning that at each step, it is decided anew, which model to evaluate. [14] High-fidelity evaluations are only performed when necessary and seen as the ground truth. The low-fidelity model gives a general fit, which is then enhanced by the more reliable data. In the case of a noisy high-fidelity model, special caution is necessary to not overfit the model. Alternatively, a nested multi-fidelity scheme can be utilised. Giannoukou et al. [15] have shown, that it is very well suited to address even large noise floors, allowing to significantly reduce the number of necessary high-fidelity data points without curbing the accuracy. The idea behind is, that the low-fidelity model captures the general relationship between inputs and outputs and the high-fidelity model can then be used to focus on the discrepancy between the latter's prediction and the observed output. This simplifies the relationship the high-fidelity model needs to capture accurately, leading to better predictions. Furthermore, the approach is simple and allows for any number of sub-models, as long as there is a clear hierarchy. It is not necessary for the sub-models to be of the same type nor be identical in their inputs. The next section gives a more detailed introduction to the scheme used.

Additional to the multi-fidelity model, a degradation model is trained based on observed change in the resistance curve. This degradation model is additionally included in the low-fidelity model, as it is possible to evaluate it before welding. Section 3.1 details how it was built.

1.4 General setup of a nested multi-fidelity model

As shown in previous work [16], this approach is well suited for process control of fast production processes and is able to deal well with noisy high-fidelity data [15, 16]. Here, as the name suggests, the next lower-fidelity (LF) model $\mathcal{M}_{LF}$ is included in the high-fidelity (HF) model $\mathcal{M}_{HF}$. The high-fidelity model $\mathcal{M}_{HF}$ can then be expressed as the sum of the scaled low-fidelity model $\mathcal{M}_{LF}$ and a discrepancy term δ

$$\mathcal{M}_{HF} = \rho \cdot \mathcal{M}_{LF} + \delta. \tag{1}$$

The constant scaling coefficient ρ can be estimated through

$$\hat{\rho} = \mathrm{E}\left[\frac{\mathcal{Y}_{HF}(\mathcal{X}_{HF})}{\mathcal{M}_{LF}(\mathcal{X}_{HF})}\right] \tag{2}$$

the expected scale of the high-fidelity output $\mathcal{Y}_{HF}$ to the low-fidelity model, evaluated at the high-fidelity input $\mathcal{X}_{HF}$ [15]. The estimation is expressed through the hat-accent. To compute the discrepancy $\hat{\delta}$, eq. 1 can be rearranged to

$$\hat{\delta} = \mathcal{Y}_{HF}(\mathcal{X}_{HF}) - \hat{\rho} \cdot \mathcal{M}_{LF}(\mathcal{X}_{HF}). \tag{3}$$

In the next step, a surrogate model $\hat{\mathcal{M}}_{\hat{\delta}}$ is trained on $\hat{\delta}$. The final output prediction $\hat{Y}$ is computed following

$$\hat{Y} = \hat{\mathcal{M}}_{\hat{\delta}}(x) + \hat{\rho}\mathcal{M}_{LF}(x) \tag{4}$$

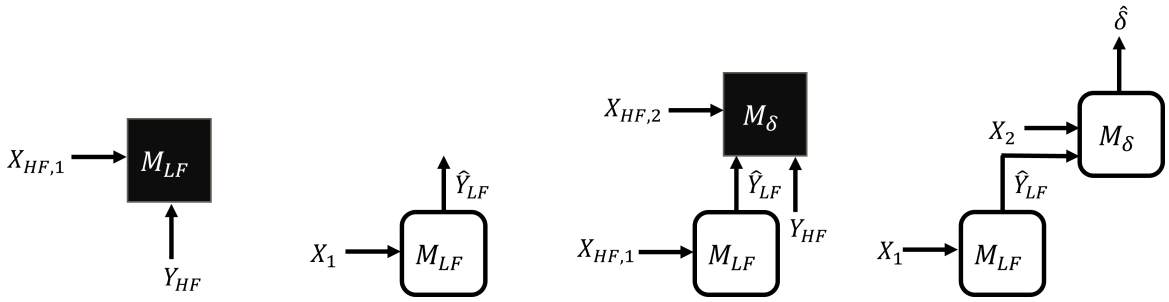
for any input x. Fig. 5 illustrates the described procedure of building a multi-fidelity model.

The black squares describe the model, that is being trained. Arrows pointing from the left into the squares indicate the inputs used for training and evaluation. For example in fig. 5a, a regression model is trained on $X_{HF,1}$ and Y_{HF}. The arrows pointing to the right inside the box indicate input data, the arrows pointing upwards into the model indicate output data relevant for training only. Fig. 5b shows the finished LF-model with X_1 as input and $\hat{Y}_{LF}$ as its output, indicated by the arrows pointing outwards the box.

The operator is free to choose the architecture and information richness for the models. In this case, $X_{HF,1}$ and $X_{HF,2}$ both correspond to the high-fidelity observations Y_{HF} but contain different information. The operator can also decide to use the same input but change the model architecture, depending on their problem. In this paper, the goal is to explore the predictability before welding, so without process information; $X_{HF,1}$ therefore contains only information known before performing any welds, while $X_{HF,2}$ consists of observations unique to each weld. Further, each model is shown as a data-based regression model. Alternatively it is possible to build a surrogate model as the low-fidelity model (fig. 5a) on the inputs and outputs of a corresponding numerical model or use the predictions of the numerical model directly as the LF-model.

1.5 Overview

In this paper, the drift behaviour is studied on an industrial spot welding dataset. A degradation model is developed, which is then included in prediction models regarding the weld's indentation estimation. A single fidelity model is being compared with a nested multi-fidelity model,



(a) Training of $\mathcal{M}_{LF}$ (b) Trained $\mathcal{M}_{LF}$ (c) Training of $\mathcal{M}_{mf}$ (d) Trained $\mathcal{M}_{mf}$

Figure 5: The nested multi-fidelity model contains a low-fidelity model (b), which is trained (a) on the relation between $\mathcal{S}$ and $\mathcal{Q}$. Using its predicted output $\hat{\mathcal{Q}}_{LF}$, a discrepancy model can be trained (c). Evaluating it (d) on $\mathcal{S}$ and $\mathcal{P}$ gives the estimated output discrepancy $\hat{\delta}$ between $\hat{\mathcal{Q}}_{LF}$ and $\mathcal{Q}$.

whose low-fidelity model only takes information known before welding and whose high-fidelity model contains all the features observed during welding itself. Pimenov et al. [17] provide a comprehensive overview of recent advances in AI-based methods to model tool degradation, such as Neural Networks, Fuzzy logic and image recognition. They conclude that decision trees and ensemble methods are particularly effective, offering high predictive accuracy without the need for extensive hyperparameter tuning or large volumes of training data, as typically required by neural networks. In this study, the degradation can be analysed independently and exhibits a relatively simple structure, eliminating the need for complex models. Nevertheless, a bagging ensemble of decision trees [18] was chosen for both the low-fidelity and multi-fidelity models due to its robustness and performance. Both (sub-)models make use of the same architecture. For each model class, three variations are compared to each other: considering no drift at all, considering the electrode health/degradation state, namely the relative lifetime/total number of welds performed and the number of welds performed within the current dress cycle and lastly, considering the degradation model.
In the following section, the dataset used is described, followed by a more in-depth model overview in section 3. In section 4 and 5 the results are presented and discussed. This paper then closes with its conclusion in section 6.

2 DATASET

The dataset to study the degradation and build the models on, was kindly provided by DAF Trucks Vlaanderen from one of their production lines. Included are observations of welds performed at 8 different spots on the same part, two each are welded with the same electrode and welding robot. One observation consists of information from four main sources:

- General welding settings $\mathcal{S}$
  Such as the current profile, welding duration or material type and thickness
- Electrode information $\mathcal{E}$:
  Here: relative lifetime n_{rel}, weld counter in current dress n_{dress}, total welds number n_{welds}
- Process data logged during each process $\mathcal{P}$
  Current and voltage curve, $I(t)$ and $V(t)$ respectively
- Quality assessment $\mathcal{Q}$
  Here: Remaining sheet thickness (see fig. 3, t_r), binary failure evaluation

Table 1: Overview over the dataset used to train the degradation and process model. The limited availability of Q constraints the dataset size.

	Information on welds in dataset			
Information source	No. locations	No. machines	No. samples	No. features
Process settings $\mathcal{S}$	8	/	8	12
Electrode information $\mathcal{E}$				
for degradation model	2	1	934	2
for process model	8	4	140	2
Process data $\mathcal{P}$				
for degradation model	2	1	934	15
for process model	8	4	140	15
Quality assessment Q	8	4	140	1

The general settings $\mathcal{S}$ determine the expected welding result and are consistent per spot. It entails all the information necessary to perform a successful weld. Most relevant are here the current profile, welding duration and thickness. Material variations between welds are considered to be small, and therefore have minor impact on the analysis. The information on the electrode caps $\mathcal{E}$ is highly relevant to determine the effect of electrode degradation and will be used to build a drift model. The process data $\mathcal{P}$ gives live-information about each individual spot and is relevant to predict the individual weld quality. During the process, the current and voltage over the time is logged and relevant features can be computed from them. Relevant features entail the mean value of current and voltage but also the dynamic resistance $R(t) = \frac{V(t)}{I(t)}$ and power $P(t) = V(t) \cdot I(t)$ curve. Especially the dynamic resistance is important to determine the welds quality, as discussed above. Additional to parameters such as mean and maximal value, principal component analysis was performed on the curves of resistance and current to detect changes in its features and gain further information from the curve.

The dataset was created over two weeks. Tab. 1 gives an overview over the utilised data. Welding settings are predefined, for the eight different spots, they contain eight distinct information. Basic logging data is readily available at minimal cost, as they are automatically recorded in any industrial resistance spot welding system due to process steering. In contrast, estimating the quality of each weld is significantly more expensive, since it requires manual, non-destructive testing by an operator. Although such testing is performed regularly to ensure product quality, the number of available measurements remains limited, restricting the available amount of samples. The dataset used in this study includes a total of 140 welds with corresponding ultrasonic measurements. To study the behaviour of degradation, no quality assessment is necessary, thus more samples can be used. During testing, several failure criteria are evaluated, including the remaining sheet thickness and whether it falls within an acceptable threshold. In the case of the dataset, the remaining sheet thickness t_r is used as Q.

3 MODELLING APPROACHES

In this chapter, the various modelling approaches are presented in further detail. This paper follows two goals: studying the effect of a degradation model on predictability, especially before actually welding and studying in what way multi-fidelity modelling can improve the results. The former is relevant for process control, monitoring, knowing when to exchange/dress the electrode. The focus here lies in separating the information known before welding and the

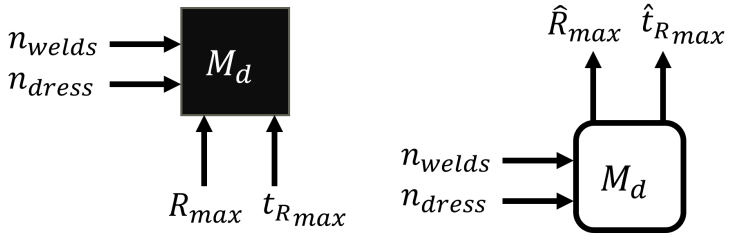
information gained during the welding process to build a reliable and robust model, that can focus on the discrepancy of individual welds from their expected line. For this, a single-fidelity model without information on neither the degradation nor the electrode state is used as the base-model. This model is enhanced once with the degradation model and once with the electrode information. The same is done for the multi-fidelity model. Tab. 2 gives a short overview over the six different models. In this chapter, the different models are explained in more detail, starting with the degradation model.

Table 2: Overview models

Name	Model Type	Information
$\mathcal{M}_{sf}$	Single-fidelity model	no information on electrode state
$\mathcal{M}_{sf,e}$	Single-fidelity model	with information on electrode state
$\mathcal{M}_{sf,d}$	Single-fidelity model	with drift model
$\mathcal{M}_{mf}$	Multi-fidelity model	no information on electrode state
$\mathcal{M}_{mf,e}$	Multi-fidelity model	with information on electrode state
$\mathcal{M}_{mf,d}$	Multi-fidelity model	with drift model

3.1 Degradation model

As discussed earlier, it is known, that the degradation of the electrode cap has an influence on the maximal resistance and the time thereof. The idea is then, build a surrogate model based on the dynamic between information on the electrode, namely the number of welds performed with the electrode and number of welds in current dress, and the dynamic resistance curve data. The model training and evaluation is shown in fig. 6. The black square of the training model (fig. 6a) indicates that the model is a data-based black-box model. The arrows pointing from the left inside the box are the inputs used to train the model and the arrows pointing from the bottom show the outputs observed for training. After training (fig. 6b) only the inputs are necessary to evaluate the model. Its predicted output is symbolised by arrows pointing away from the top of the square.



(a) Training (b) Evaluation

Figure 6: The degradation model is a black-box model trained (a) on the input-output relation of $\mathcal{E}$ (n_{welds} and n_{dress}) and R_{dyn}-key parameters (R_{max} and $t_{R_{max}}$).

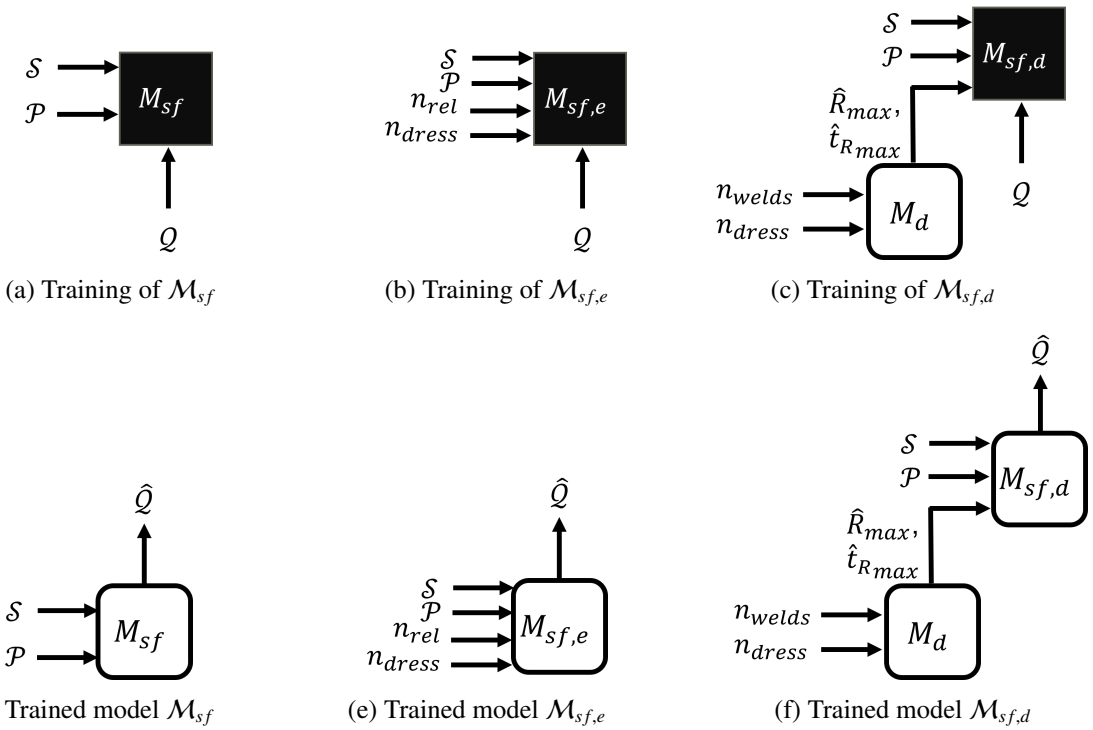
The degradation between electrodes varies, some clearly show the expected behaviour, while it is not as obvious for others. To build a reliable drift model, 934 data points belonging to two spots welded with the same electrode were chosen. The drift behaviour of those two spots is the

same and clearly identifiable Therefore, only two out of the 8 spots are used for that. A gradual mostly monotonic change was observed, so a low order polynomial of degree 5 was chosen for the regression model with dress cycle counter and total weld counter as inputs and R_{max} and $t_{R_{max}}$ as outputs.

3.2 Single-fidelity model

For the single-fidelity (SF) model $\mathcal{M}_{sf}$ a bagging ensemble tree [18] was chosen with 200 learning cycles which selects the best split predictor based on the interaction-curvature-test. Here the split predictor is chosen to minimise the p-value of the chi-square tests of independence between each predictor and response as well as each pair of predictor and their response. [19]

Fig. 7 shows a schematic of the different single-fidelity models used as well their input and outputs. The base-multi-fidelity model $\mathcal{M}_{sf}$ (fig. 7a and 7d), which contains no further information on the electrodes, is trained on a 27-dimensional input consisting of the welding settings $\mathcal{S}$ as well as observations during welding $\mathcal{P}$. The corresponding output for training is the measured remaining sheet thickness Q (fig.7a). $\mathcal{M}_{sf,e}$ additionally contains information on the relative lifetime n_{rel} of the electrode and the weld counter within the current dressing n_{dress} (fig. 7b and 7e). In the degradation-informed model $\mathcal{M}_{sf,d}$, the prediction of the degradation model $\hat{R}_{max}$ and $\hat{t}_{R_{max}}$ is used as additional inputs to the single-fidelity model. An overview of the different methods and their input is provided by tab. 3.



(a) Training of $\mathcal{M}_{sf}$ (b) Training of $\mathcal{M}_{sf,e}$ (c) Training of $\mathcal{M}_{sf,d}$

(d) Trained model $\mathcal{M}_{sf}$ (e) Trained model $\mathcal{M}_{sf,e}$ (f) Trained model $\mathcal{M}_{sf,d}$

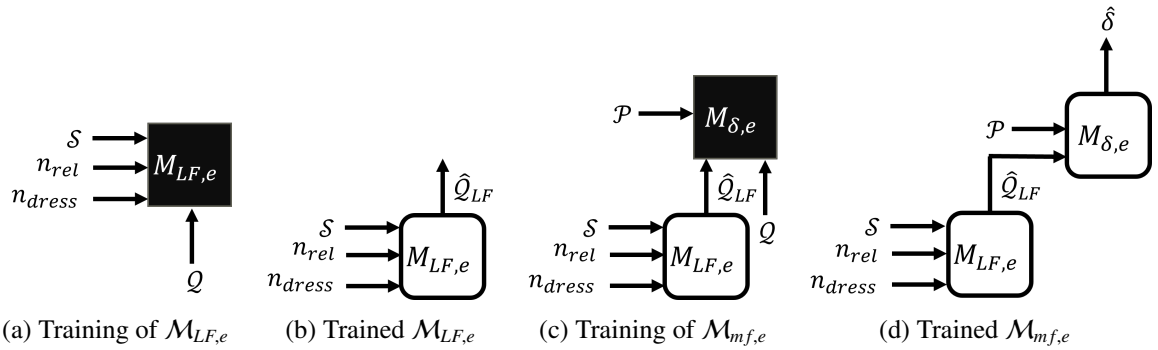
Figure 7: The single-fidelity model has $\mathcal{S}$ and $\mathcal{P}$ as inputs. It is trained on output Q (a), of which it gives a prediction $\hat{Q}$ after training is complete (d). $\mathcal{M}_{sf,e}$ additionally takes electrode information as input (b,e) and $\mathcal{M}_{sf,d}$ the output of the degradation model $\mathcal{M}_d$ (c,f).

3.3 Nested multi-fidelity model

Additionally to the single-fidelity model, a nested multi-fidelity model is used as well. In this case, it is a random forest model, build with the same hyperparameters as the single-fidelity models and the high-fidelity ones. The low-fidelity surrogate model was trained on 140 samples using a 12-dimensional input features space $\mathcal{S}$ (excluding drift information) and a one-dimensional output measure Q. The base model was enhanced in two manners: a) the drift model evaluation of R_{max} and $t_{R_{max}}$ or b) the information on relative lifetime and number of welds in current dressing were added to the input features space. An overview is given in tab. 3.

As the model is trained on the measured remaining sheet thickness, the scaling factor $\hat{\rho}$ can be set to 1. In the next step, the discrepancy $\hat{\delta}$ of each LF-model prediction and actual measured value is computed. This difference becomes the new output features to build the discrepancy model $\mathcal{M}_{\hat{\delta}}$ on. The 15-dimensional input-space for the high-fidelity model contains $\mathcal{P}$, the information gained during the welding process itself. The expected output can then be computed by adding the low-fidelity prediction to the discrepancy model prediction. Fig. 8 and fig. 9 described how the multi-fidelity models with electrode information and degradation model respectively are build. For the base multi-fidelity model, it is referred to fig. 5. It is visible, that the difference in these three models lies not in the high- but only in the low-fidelity model and its predictions. For the schematic of the degradation model $\mathcal{M}_d$, the reader is referred to fig. 6.

In conclusion, the multi-fidelity model consists of a high-fidelity model, that is informed about each individual process and has access to evaluations of the low-fidelity model, which gives only the generally expected behaviour. Each of these models is build from a bagging ensemble tree model.

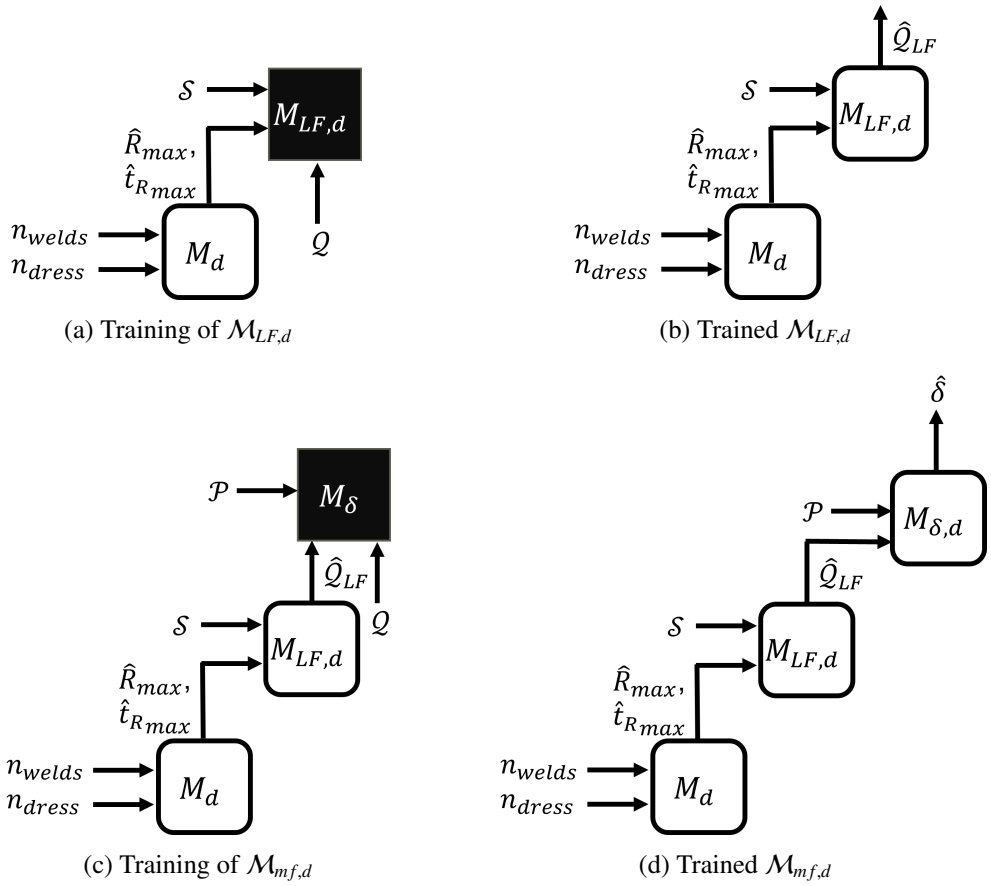


(a) Training of $\mathcal{M}_{LF,e}$ (b) Trained $\mathcal{M}_{LF,e}$ (c) Training of $\mathcal{M}_{mf,e}$ (d) Trained $\mathcal{M}_{mf,e}$

Figure 8: $\mathcal{M}_{mf,e}$ contains a low-fidelity model (b), which is trained (a) on the relation between settings $\mathcal{S}$, electrode information n_{rel}, n_{dress} and observed output Q. Using its predicted output $\hat{Q}_{LF}$, a discrepancy model can be trained (c). Evaluating it (d) on $\mathcal{S}$, $\mathcal{P}$ and $\mathcal{E}$ gives the estimated output discrepancy $\hat{\delta}$.

3.4 Overview used models

In total, six models, three single- and three multi-fidelity models (tab. 2), are compared to each other in this paper. Each of these models (apart from the degradation model) uses a random forest surrogate model with the same training hyperparameters. The main difference is, that the multi-fidelity model first builds an expectation model based on information before the welding and refines it with measurements during the process. The single-fidelity model utilises everything at once. Further information which information on the models can be found in tab. 3.



(a) Training of $\mathcal{M}_{LF,d}$ (b) Trained $\mathcal{M}_{LF,d}$

(c) Training of $\mathcal{M}_{mf,d}$ (d) Trained $\mathcal{M}_{mf,d}$

Figure 9: $\mathcal{M}_{mf,d}$ contains a low-fidelity model (b), which is trained (a) on the relation between $\mathcal{S}$, degradation model $\mathcal{M}_d$ result and $\mathcal{Q}$. Using its prediction $\hat{\mathcal{Q}}_{LF}$, a discrepancy model can be trained (c). Evaluating it gives the estimated output discrepancy $\hat{\delta}$ between $\hat{\mathcal{Q}}_{LF}$ and $\mathcal{Q}$ (d).

4 RESULTS

This section presents the key results of the different models. Table 4 summarises the R-squared values of the different models. The best performing model with this metric is, as expected, $\mathcal{M}_{mf,d}$, the multi-fidelity model, that considers the degradation model in its low-fidelity model with an R-squared value of 0.89. The other multi-fidelity models show a similar performance, all with R-squared values above 0.8. Based on the table alone, the performance of the single fidelity models appears to be comparable to all of the low-fidelity models. However, when inspecting the actual prediction in fig. 10, fig. 11a and fig. 12a, significant differences with the model fit can be observed. These figures compare the actual measured remaining sheet thickness to the predicted values. Without noise in the process and measurements, a perfect prediction would have points aligned with the red dotted line. As they are however noisy, a scatter around the line would be the expected optimal result. $\mathcal{M}_{sf}$ in fig. 10 demonstrates generally good predictive performance, except for three significantly overestimated and six underestimated samples. The discrepancy on those 9 samples appear consistently across models.

While the discrepancy model clearly improves the performance, this is not reflected in the R-squared values of the discrepancy model. The reason lies in the outliers but might also point to a lack of information to further improve the model performance.

The degradation-uninformed low-fidelity model $\mathcal{M}_{mf,LF}$ predicts a limited number of dis-

Table 3: Overview about which information is contained in which model. A filled circle indicates that the information is included directly, an empty circle, that it is included implicitly and a dash, that it is not included.

Name	Sub-model	Has information on				Dimension	
		Settings $\mathcal{S}$	Measurements $\mathcal{P}$	Degradation model $\mathcal{M}_d$	Electrode information $\mathcal{E}$	Input	Output
$\mathcal{M}_{sf}$		●	●	-	-	27	1
$\mathcal{M}_{sf,e}$		●	●	-	●	29	1
$\mathcal{M}_{sf,d}$		●	●	●	-	29	1
$\mathcal{M}_{mf}$	LFM	●	-	-	-	12	1
	MFM	○	●	-	-	15	1
$\mathcal{M}_{mf,e}$	LFM	●	-	-	●	14	1
	MFM	○	●	-	○	15	1
$\mathcal{M}_{mf,d}$	LFM	●	-	●	-	14	1
	MFM	○	●	○	-	15	1

tinctive values, see fig. 11a. This behaviour was expected, as the model exclusively considers information on the weld settings. The process is highly noisy and affected by drift but the model contains no information on neither. Additionally, there are only 8 different settings, so an average output value per setting is computed.
The updated multi-fidelity model $\mathcal{M}_{mf}$ (Fig. 11b) demonstrates a significantly improved fit, with an R-squared value of 0.83, compared to 0.62 for the single-fidelity model. The outliers are still prevalent however, although slightly closer to the measured result.

The drift-informed low-fidelity model $\mathcal{M}_{mf,d,LF}$ (Fig. 12a) suggests that drift plays a crucial role in output predictions. While its visual fit appears worse than that of the single-fidelity model, it achieves similar R-squared values and slightly improves outlier predictions. The corresponding multi-fidelity model performs best out of the six final models with an R-squared value of 0.89 and a better visual fit.

Explicit figures for the other three models are omitted, as their fit closely resembles that of one of the figures shown. The single-fidelity models all perform similarly and the multi-fidelity-model informed about the electrode state, instead of the explicit drift model, shows similar behaviour to the latter. While it is possible, to include the drift model as well as the information on the electrode state, it does not provide additional insights and slightly decreases overall performance.

Table 4: R2-values of different models compared

	Low-fidelity model	Discrepancy model	Final model
$\mathcal{M}_{sf}$	-	-	0.6150
$\mathcal{M}_{sf,e}$	-	-	0.6266
$\mathcal{M}_{sf,d}$	-	-	0.6298
$\mathcal{M}_{mf}$	0.6598	0.0800	0.8296
$\mathcal{M}_{mf,e}$	0.6462	0.1423	0.8862
$\mathcal{M}_{mf,d}$	0.6470	0.1228	0.8923

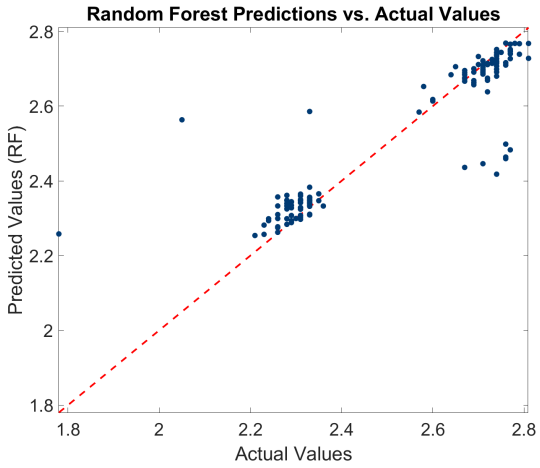
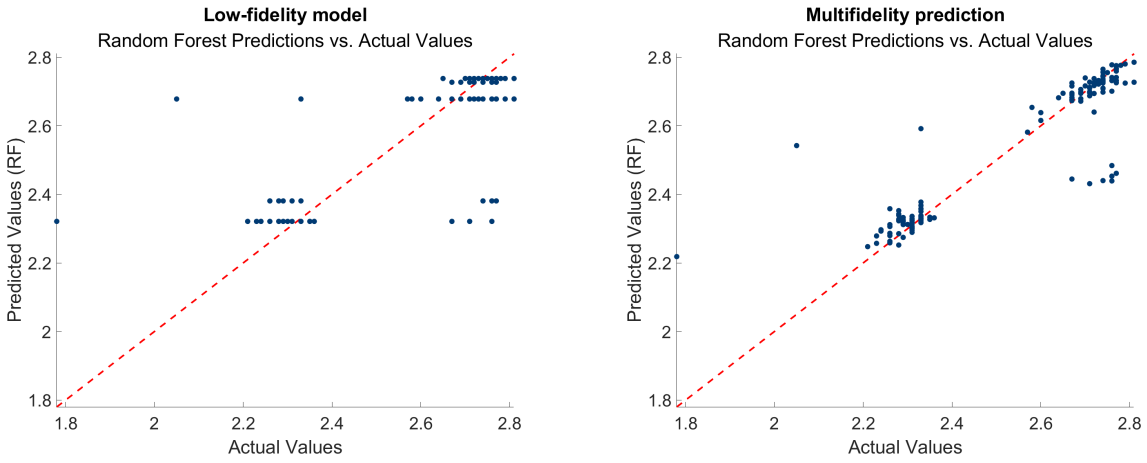


Figure 10: Remaining sheet thickness: measured vs predicted with a single-fidelity model without drift



(a) Low-fidelity model of $\mathcal{M}_{mf}$

(b) Final $\mathcal{M}_{mf}$

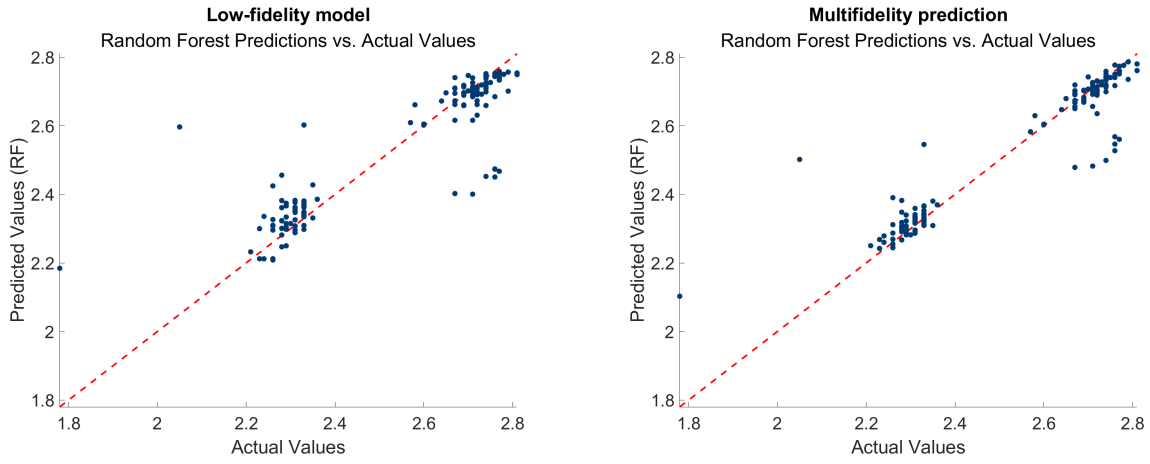
Figure 11: Remaining sheet thickness: measured vs predicted with a low-fidelity model and the final multi-fidelity model without considering the drift model nor electrode information in the low-fidelity model.

5 DISCUSSION

In this section, we are focussing on the meaning of the results with respect to process monitoring and reliable failure detection and prediction.

5.1 Failure detection and outliers

One aspect to evaluate additionally to the general fit of the prediction is, whether failure was correctly identified. There are two failure modes with regards to the remaining sheet thickness. The first one, is a too light indentation, which can point to a stick-weld. This means, that the two sheets are not properly joined but rather stuck together with an insufficient yield strength. To avoid this failure, the minimum indentation requirement was set to 3 % of the total thickness, based on internal guidelines and DVS 2960 [10]. There were seven welds that failed in this manner, all of which, were correctly identified by each method. In the second failure mode the indentation is too heavy, too little material of one sheet is remaining and one can expect a shear failure, also with insufficient yield strength. In the case of the dataset, the ultrasonic device



(a) Low-fidelity model of $\mathcal{M}_{mf,d}$

(b) Final $\mathcal{M}_{mf,d}$

Figure 12: Remaining sheet thickness: measured vs predicted with a low-fidelity model and final multi-fidelity model when considering the drift model in the low-fidelity model.

detected three failed welds. When evaluating the criterion of maximal 40 % indentation of the thinner sheet however, all welds passed. Usually the ultrasonic device uses the same threshold to detect failure. So while on paper all models were able to identify failure correctly, it still became obvious, that, especially heavy indentation, was often times not well estimated.

One possible reason for the overestimation of remaining sheet thickness are outliers, which can affect the modelling process in two distinctive ways. Firstly, outliers in the input features; when looking only at the input features that were measured during the process, 18 observations can be detected, where at least one process measurement is more than 1.5 interquartile ranges above or below the 25-75-quartile. Normally, outliers in the input features are easy to handle, especially, if, as in this case, their corresponding output values are within the expected range. Ensemble methods like random forests are especially robust against them, as they rely on averaging over multiple decision trees. This reduces the impact of a limited number of outliers. However, this filtering can also affect rare event estimation.
Secondly, the model may struggle to accurately predict samples where the output is an outlier but the input is not. This is particularly problematic when such cases are rare (i.e. in this dataset), as the model has insufficient examples to learn the mapping for these extreme cases. Since random forests rely on majority voting or averaging across trees, they inherently bias predictions toward the dominant trend in the training data. As a result, they may fail to capture rare deviations, leading to poor predictions for these output outliers. In the dataset, three such samples exist. One sample is more than 1.2 interquartile ranges below the 25-quartile and two additional samples are 0.2 interquartile ranges below. These three outliers are also identified when looking at the scaled median absolute deviation (MAD) larger than three. Looking instead at three standard deviations from the mean, only one outlier remains. The usage of mean and median to detect outliers needs to be taken with a grain of salt, as we obviously do not have a normal but multimodal distribution here. If it is not possible to augment the dataset to include more rare events, alternative strategies such as weighted training or anomaly detection can be used to address the problem.

5.2 Conclusion for process monitoring

In sec. 4, it became clear, that multi-fidelity modelling improves predictability and model fit. Moreover, including additionally the degradation model further improves the fit and has a positive effect on outliers.

The drift-informed low-fidelity model $\mathcal{M}_{sf,d}$ already results in a strong predictability, pointing out how significant R_{max} and $t_{R_{max}}$ are in determining the output quality. The drift model is therefore able to capture a large portion of the process behaviour and indicate expected quality of future welds already before welding. This holds significant value for process control by enabling inverse quantification to determine optimal input parameters to counter the drift. Further, optimal moments when to dress and exchange the electrode can be defined as well. Improving the model will also allow to detect unexpected behaviour from which further conclusions can be drawn. For actual process monitoring however, it is recommended to still include the live-process parameters, as they are cheap to include and hold additional valuable information.

While these results are already very promising and interesting, it needs to be pointed out that the degradation model was trained not including every spot available and that the degradation behaviour can vary significantly between different electrodes and robots. The observed drift behaviour largely depends on the electrode, the welding history and appropriate welding settings. So there is no one-model-fits-all and one should build their own drift model and check, where it is applicable.

6 CONCLUSION

To enable reliable process control and good quality estimation in industrial manufacturing processes, a model is necessary, that can handle limited quality assessment availability, noisy as well as degradation within the production line. This paper investigates how modelling electrode degradation affects the quality prediction of resistance spot welds. To that end, various black-box and multi-fidelity models, with and without an explicit drift model were compared with each other. The results can be summarised in three key findings.

- The multi-fidelity model outperforms the single-fidelity model.
- Models including a separate degradation model show better fit than not considering process drift or only considering the electrode information directly.
- The fit of the low-fidelity model including the degradation model $\mathcal{M}_{mf,d,LF}$ comparable to single-fidelity models and multi-fidelity models.

The degradation-informed multi-fidelity model $\mathcal{M}_{mf,d}$ and its corresponding low-fidelity model $\mathcal{M}_{mf,d,LF}$ both serve two important but different tasks. The multi-fidelity model is relevant for process monitoring, it can be used to detect abnormal behaviour and to predict each weld result more accurately. The low-fidelity model on the other hand can be used for general process steering, decisions about the electrode dressing and exchange, as it provides already many insights before the process. As such, it will need special consideration with respect to the available measurements. In this case, it can prove beneficial, to include a numerical modelling scheme or perform separate experiments as well.

In the future, inverse quantification schemes or model updating can be applied to determine not only optimal moments to dress and exchange the electrode but also optimal control parameters to counter the effect of the drift. The latter is a nested optimisation scheme, as changing the welding parameters obviously also affects not only the maximal resistance and time thereof, but also the speed of degradation.

ACKNOWLEDGEMENTS

The authors would like to thank DAF Trucks Vlaanderen for the opportunity to work with their data and give special thanks to Gerry Volders for his support in working with and understanding the dataset as well as his expert knowledge on not only the welding process in general but also about the specifics at DAF.

REFERENCES

[1] K. Zhou and P. Yao, "Overview of recent advances of process analysis and quality control in resistance spot welding," *Mechanical Systems and Signal Processing*, vol. 124, pp. 170–198, Jun. 2019.

[2] C. Mathiszik, D. Köberlin, S. Heilmann, J. Zschetzsche, and U. Füssel, "General Approach for Inline Electrode Wear Monitoring at Resistance Spot Welding", *Processes*, vol. 9, Apr. 2021, doi: 10.3390/pr9040685.

[3] A. Olleak and Z. Xi, "Calibration and Validation Framework for Selective Laser Melting Process Based on Multi-Fidelity Models and Limited Experiment Data", *Journal of Mechanical Design*, vol. 142, no. 081701, Feb. 2020, doi: 10.1115/1.4045744.

[4] P. Nath, M. Sato, P. Karve, and S. Mahadevan, "Multi-fidelity Modeling for Uncertainty Quantification in Laser Powder Bed Fusion Additive Manufacturing", *Integr Mater Manuf Innov*, vol. 11, no. 2, pp. 256–275, Jun. 2022, doi: 10.1007/s40192-022-00260-9.

[5] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital Twin in manufacturing: A categorical literature review and classification", *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1016–1022, 2018, doi: 10.1016/j.ifacol.2018.08.474.

[6] M. G. Fernández-Godino, "Review of multi-fidelity models", *ACSE*, vol. 1, no. 4, pp. 351–400, 2023, doi: 10.3934/acse.2023015.

[7] A. O'Brien and American Welding Society, Eds., *Welding handbook". Vol. 3: Welding processes, part 2* / Annette O'Brien, ed, 9. ed., vol. 3. Miami, Fla: *American Welding Soc*, 2007.

[8] L. Bogaerts, A. Dejans, M. G. R. Faes, and D. Moens, "A machine learning approach for efficient and robust resistance spot welding monitoring", *Weld World*, vol. 67, no. 8, pp. 1923–1935, Aug. 2023, doi: 10.1007/s40194-023-01519-1.

[9] Q. Fan, G. Xu, and X. Gu, "Expulsion characterization of stainless steel resistance spot welding based on dynamic resistance signal", *Journal of Materials Processing Technology*, vol. 236, pp. 235–240, Oct. 2016, doi: 10.1016/j.jmatprotec.2016.05.026.

[10] ”Unregelmäßigkeiten bei Widerstandspunktschweißverbindungen an Stahlwerkstoffen”, *DVS 2960*, Aug 2015

[11] ”Welding and allied processes — Classification of geometric imperfections in metallic materials”, EN ISO 6520-2:2013, 2013.

[12] D.J. Pitchforth, T.J. Rogers, U.T. Tygesen, and E.J. Cross. ”Grey-box models for wave loading prediction”. *Mechanical Systems and Signal Processing*, 159, October 2021.

[13] H. J. A. F. Tulleken, ”Grey-box modelling and identification using physical knowledge and bayesian techniques”, *Automatica*, vol. 29, no. 2, pp. 285–308, Mar. 1993, doi: 10.1016/0005-1098(93)90124-C.

[14] B. Peherstorfer, K. Willcox, and M. Gunzburger, “Survey of multi-fidelity Methods in Uncertainty Propagation, Inference, and Optimization,” *SIAM Review*, vol. 60, no. 3, pp. 550–591, Jan. 2018.

[15] K. Giannoukou, S. Marelli, and B. Sudret, ‘Uncertainty-aware multi-fidelity surrogate modeling with noisy data’, Oct. 29, 2024, doi: 10.48550/arXiv.2401.06447.

[16] M.B. Dodt, L. Bogaerts, A. Persoons, M.G.R. Faes, D. Moens, D. ”A multifidelity framework for grey-box modelling of automated welding processes.” *Proceedings of ISMA2024 International Conference on Noise and Vibration Engineering USD2024 International Conference on Uncertainty in Structural Dynamics*, 4142-4150. Leuven, Belgium, Sep 9-11 2024. ISBN: 978-90-82893-17-5

[17] D. Y. Pimenov, A. Bustillo, S. Wojciechowski, V. S. Sharma, M. K. Gupta, and M. Kuntoğlu, ”Artificial intelligence systems for tool condition monitoring in machining: analysis and critical review”, *J Intell Manuf*, vol. 34, no. 5, pp. 2079–2121, Jun. 2023, doi: 10.1007/s10845-022-01923-2.

[18] L. Breiman, ”Bagging predictors”. *Machine Learning*. 24 (2): 123–140. 1996. doi:10.1007/BF00058655.

[19] W.Y. Loh, “Regression Trees with Unbiased Variable Selection and Interaction Detection.” *Statistica Sinica*, Vol. 12, pp. 361–386, 2002.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

PHYSICALLY-INFORMED CHANGE-POINT KERNELS FOR INTERMITTENT INCLUSION OF PHYSICAL KNOWLEDGE

D.J. Pitchforth[1], S.J. Gibson[2], and E.J. Cross[2]

[1]The University of Sheffield
School of Mechanical, Aerospace and Civil Engineering
Western Bank, Sheffield S10 2TN
e-mail: {d.j.pitchforth,s.j.gibson,e.j.cross}@sheffield.ac.uk

Abstract. *The encoding of physical knowledge within the design of covariance functions (kernels) can provide an avenue to enforce desirable constraints and behaviours within the predictions of Gaussian process regression models. When used in combination with flexible, general purpose kernels, models may be constructed that incorporate a user's understanding of a system, whilst maintaining a degree of flexibility to capture parts of a process that may be less well understood. A key challenge of constructing such models is achieving the optimum balance of explanatory power between the kernel components. Typically, the higher the extent to which the behaviour of a system is understood, the more one would wish to rely upon a physically-derived kernel component. This paper develops physically-informed change-point kernels that dynamically adjust the relative reliance upon each kernel component to reflect the user's physical understanding of a process at any given point in time. This is presented in the context of aircraft wing strain prediction, where the oscillatory behaviour of a cantilever beam, representing modes of the wing, is phased into the kernel during aerial manoeuvres.*

Keywords: Physics-informed Kernel Design, Change-point Kernel, Gaussian Processes, Aircraft Manoeuvres.

1 INTRODUCTION

Engineering knowledge and our understanding of how physical systems behave is a valuable resource. Physics-informed Machine Learning aims to utilise this resource to enhance the capabilities of data driven models, by improving performance, and providing insight into their results. Although highly flexible, and capable of capturing complex relationships, purely data-based learners can only expect to predict reliably within the realms of previously observed conditions. For many engineering structures, particularly those in harsh environments, the possible set of observed conditions is too large to feasibly measure completely. Through leveraging physical knowledge, PIML offers a potential solution to reduce the demand for data collection to a more attainable level.

A key decision when constructing any PIML model, and the one discussed in this paper, is the balance of explanatory power between physics and data within a model. Typically, our understanding of a system is used to create some mathematical representation, whilst potential gaps in knowledge (variations due to environment, manufacturing tolerances etc.) are captured through data [1–3]. Earlier work of the authors [4] discusses the relationship between how the structure of Gaussian process models affects this balance and how it may change across modelling tasks.

This work looks toward how one might be able to tune the relative reliance on physics and data within physics-informed Gaussian Process (GP) models. A GP is a flexible, non-parametric, Bayesian technique, adept within a variety of engineering regression problems, including crack growth [5, 6], tool wear [7, 8] and modelling of wind turbine power curves [9, 10]. For conciseness, an introduction to GP regression theory is not presented here, with interested readers encouraged to consult [11]. The covariance function (kernel) of a Gaussian process is responsible for the family of functions from which predictions may be drawn, with commonly used kernels enforcing properties such as smoothly varying functions, periodicity and localised behaviours [11, 12]. Through careful design of the kernel, it is possible to mimic physically desirable behaviours within drawn functions; such examples include the representation of a physical process [13, 14] or the enforcement of axial and rotational symmetries [15–18]. Here, a combination of physics-informed kernels and flexible, more general purpose, kernels are used in combination. The constructed physics-informed change-point kernels aim to dynamically switch the relative reliance between kernel components, allowing for variation of the physics-data balance within the model.

Intermittent physical phenomena occur in many engineering structures, and depending on their rarity, a scarcity of data may make them challenging to model using purely data-based techniques. This paper highlights how one might phase in physical knowledge to aid model predictions when such phenomena occur. Here, a familiar example of engineering knowledge, the oscillation of a cantilever beam, is embedded within a Gaussian process kernel to aid predictions of aircraft wing strain. This is phased in and out of the model using a change-point structure, utilising the inputs of a pilot to change the relative explanatory power of kernel components.

2 Physics-informed change-point kernels

First presented in [19], and built upon the work of [20, 21], physics-informed change-point kernels provide an effective means to vary the reliance upon physically-derived kernels within a GP. A pair of sigmoid kernels $K_\sigma(Z, Z')$ and $K_\sigma(-Z, -Z')$ are used to control the explanatory power of a physics-informed kernel and a more flexible 'data-based' kernel within the model.

This leads to a kernel structure of:

$$K(Z, Z', X, X') = \underbrace{K_\sigma(Z, Z')K_{Phy}(X, X')}_{\text{Physics-informed kernel}} + \underbrace{K_\sigma(-Z, -Z')K_{Data}(X, X')}_{\text{Flexible kernel}} \tag{1}$$

where the input to the sigmoid kernel Z is used to control the switching between kernels $K_{Phy}(X, X')$ and $K_{Data}(X, X')$. An effective choice of Z relates closely to how a physical relationship might change with a variable. For example, temperature, humidity, excitation level, or measures of turbulence can all effect the extent to which one might want to rely on a given piece of physical knowledge.

3 Case study: Changing physical behaviour of an aircraft wing.

Here, a case study focuses on the prediction of aircraft wing strain during an turn and how a changing physical relationship is induced due to the inputs of the pilot. Reed [22] previously achieved success modelling wing strain with MLP networks, resulting in an ability to estimate fatigue damage for the wing within $\pm 3\%$. Later work of Fuentes [23] and Gibson [24] on the dataset used GP regression to enable access to a full predictive distribution over the strains. Gibson explored how draws from the predictive distribution may be propagated to achieve distributions over the fatigue damage.

It is worth emphasising that, where previous work focussed on the accuracy of fatigue damage estimation [22, 24], the main aim here is the identification and modelling of changing physical phenomena. For this reason, the work predicts wing strain from a subsection of the dataset, where the pilot used an input to the rudder to cause the aircraft to turn. A time series of the wing strain and rudder angle is shown in Figure 1. At high rudder angles, an aircraft will turn about its Z axis, and the introduction of some additional dynamic behaviour can be seen.

3.1 Kernel design

The wing strain is considered to consist of two main parts: a quasi-static component, capturing the more slowly varying relationship between flight parameters and strain; and a dynamic component, induced by oscillations occurring during manoeuvres. In this case, the manoeuvre of interest is a turn due to a high rudder angle. To represent this, the problem is structured as:

$$y(X) = \underbrace{f_1(X)}_{\text{Quasi-static component}} + \underbrace{f_2(X)}_{\text{Dynamic, manouver-induced behaviour}} + \underbrace{\epsilon}_{\text{Noise}} \tag{2}$$

To capture the relationship between the measured flight variables and the wing strain, inline with previous kernel selections within the literature [23, 24], we employ the Squared Exponential (SE) kernel:

$$K_{SE}(X_{Fl}, X'_{Fl}) = \sigma_f^2 \text{exp}\left(-\frac{1}{2}(X_{Fl})^T \Lambda^{-1}(X'_{Fl})\right) + \sigma_n^2 \delta_{ij} \tag{3}$$

where σ_f^2 is the signal variance, σ_n^2 is the noise variance, Λ is the matrix of length scales such that $diag(\Lambda) = [l_1^2, l_2^2, ..., l_D^2]$ for a D dimensional input and X_{Fl} is a matrix of flight parameters.

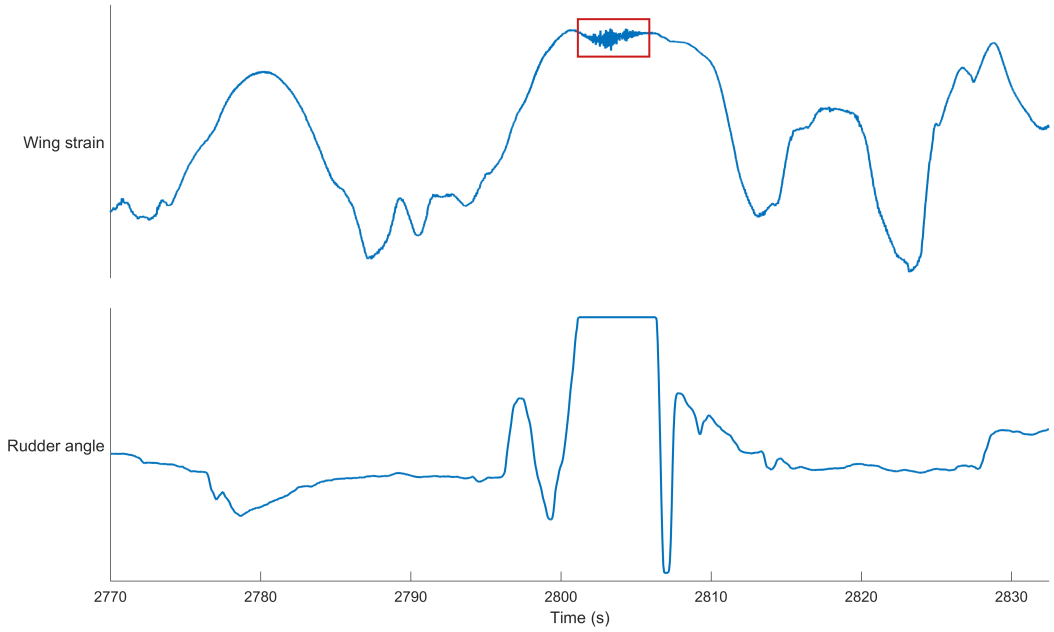


Figure 1: Time series plots of wing strain and rudder angle. The red box highlights an instance of oscillatory behaviour, believed to be caused by a turning of the aircraft, occurring at a high rudder angle.

To capture the covariance of the oscillatory behaviour, highlighted in Figure 1, I employ the derived SDOF kernel [13]:

$$K_{SDOF}(\tau) = \frac{\sigma^2}{4m^2\zeta\omega_n^3} e^{-\zeta\omega_n|\tau|}\left(cos(\omega_d\tau) + \frac{\zeta\omega_n}{\omega_d} sin(\omega_d|\tau|)\right) \tag{4}$$

where the hyperparameters of the kernel now relate to physical properties of a SDOF oscillator: m is the mass, $\zeta = c/2\sqrt{km}$ is the damping ratio, $\omega_n = \sqrt{k/m}$ is the natural frequency and $\omega_d = \omega_n\sqrt{1-\zeta^2}$ is the damped natural frequency. Draws from this kernel are constrained to obey the behaviour of an SDOF oscillator, a useful property to encode. Here the SDOF kernel aims to capture the oscillatory behaviour of the wing, similarly to how previous implementations of the kernel have been used to represent modes of an oscillating cantilever beam [25, 26].

A feature of the dataset, and one of the primary reasons it has been selected for this case study, is that the dynamic behaviour is intermittent. To allow the SDOF kernel to be phased in and out appropriately, it is integrated in a change-point structure of the form:

$$\begin{aligned} K(X, X') = & \underbrace{K_{SE}(X_{Fl}, X'_{Fl})}_{\text{Capture the static signal component with a SE kernel...}} \\ & + \underbrace{K_\sigma(\text{RUDD}, \text{RUDD}')K_{SDOF}(t, t')}_{\text{... whilst introducing the SDOF kernel for rudder-based manoeuvres.}} \end{aligned} \tag{5}$$

where X_{Fl} is a matrix of measured flight parameters, RUDD is the rudder angle and t is the time vector.

3.2 Results

For the test cases shown here, models were shown every 8th point of a 4000 data point subset during training. The subset was centred around a rudder-based manoeuvre. For the strain data, measured at 128Hz, this meant the models were allowed access to an effective sample rate of 16Hz and were being asked to upsample.

To highlight the challenge of the task, the problem was first modelled using what might be considered a typical black-box approach, $\mathcal{GP}(0, K_{SE}(X_{Fl}, X'_{Fl}))$. A zero mean GP, with a squared exponential kernel acting on the same matrix of flight parameters, X_{Fl}. The results for this model are shown in Figure 2.

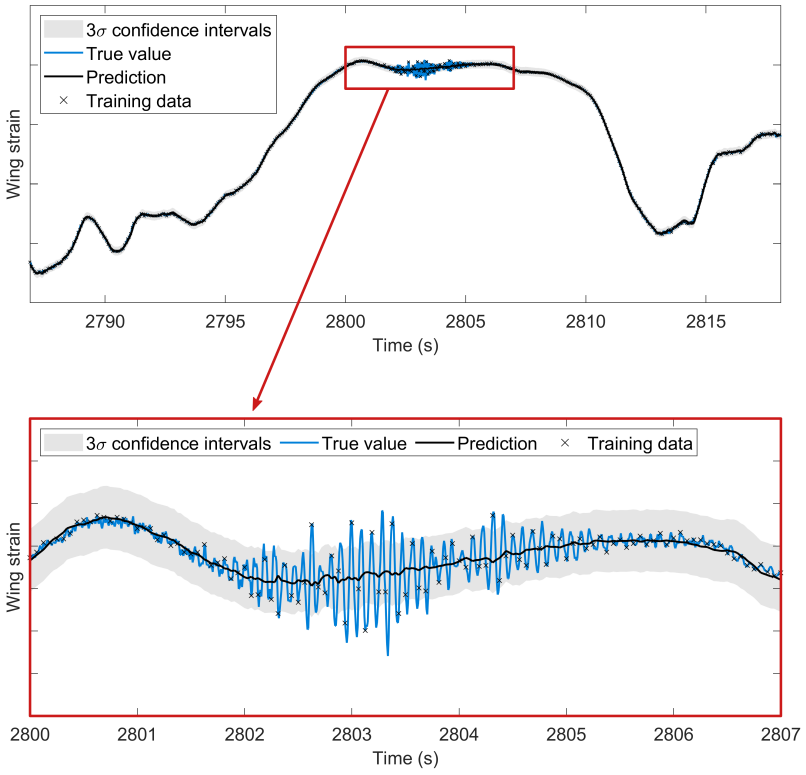


Figure 2: SE kernel prediction of inner port side wing strain. The model has observed every 8th data point of a 4000 point subset, centred around an in-flight manoeuvre. The red box and lower subplot highlights performance in a region of induced oscillatory behaviour within the wing. The model struggles to characterise the higher frequency dynamic behaviour effectively.

For the majority of the time series, the prediction of the SE kernel achieved a good fit to the data, with an NMSE of 0.0389% obtained across the full 4000 point test set. As a task of interpolation, one would expect a black-box model to perform well here and a good performance is to be expected. Where the model performs less well, and the primary region of focus, is where the dynamic behaviour is induced due to the turning of the aircraft. This is highlighted within the red box and lower subplot of Figure 2. In this region, the NMSE of the model worsens to 48.2424% and the model does a poor job of characterising the oscillatory behaviour. The 16Hz observed sample rate was not high enough for the SE kernel to reconstruct the higher frequency

behaviour present within the 128Hz strain measurements. Instead, the model smoothed through the dynamic content, missing out on much of the structure present. As a side effect of this, a relatively high estimate was placed on the noise present within the signal, inflating the confidence bounds higher than necessary away from the dynamic content.

The prediction of the physics-informed change-point kernel, defined in Equation 5 is shown in Figure 3. The performance across the full test set improved to an NMSE of 0.0124%, less than one third of the NMSE achieved by the black-box model. A large contributor to this was a greatly improved ability to capture the dynamic content through the phasing in of a SDOF kernel during turning manoeuvres. In the region of the dataset highlighted by the red box, centred around the dynamic content, the NMSE of the model dropped from 48.24% to 6.93% and the model was much better able to capture the structure present within the signal. Instead of smoothing through the oscillations, the SDOF kernel was able to effectively capture the higher frequency content and provide an appropriate quantification of uncertainty. The inclusion of the SDOF kernel also allowed the SE kernel to better capture the static component of the signal. Since the SE kernel was not attempting to capture two distinct behaviours, a longer lengthscale and smaller noise variance were able to be used away from the dynamic behaviour, thereby improving overall performance.

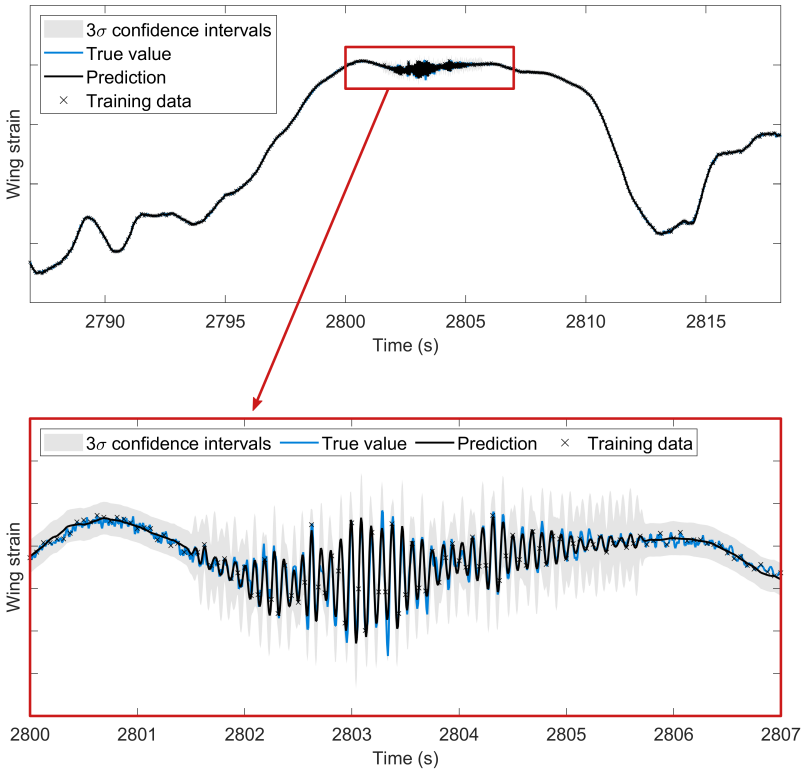


Figure 3: The prediction of inner port side wing strain using the physics-informed change-point kernel, defined in Equation 5. The model has observed every 8th data point of a 4000 point subset, centred around an in-flight manoeuvre. The red box and lower subplot highlights performance in a region of induced oscillatory behaviour within the wing. The model is able to better characterise the higher frequency dynamic behaviour than a purely data-based approach.

3.3 Posterior contribution breakdown

One of the advantages of kernel design, particularly when combining multiple kernels, is that an additional level of insight can be gained by looking at the posterior contributions of individual kernel components. For a sum of kernels, this required some decomposition. Following [27] it is possible to derive the conditional predictive distribution for the contribution of a kernel K_i within a combined additive kernel of the form $K = \sum_{i=1}^{i=n} K_i$.

$$\begin{aligned} &p(\boldsymbol{f_i^*}|\boldsymbol{f^*}, X^*, \boldsymbol{f}, X, \boldsymbol{\theta}) \sim \\ &\mathcal{N}(K_i^{*T}(\sum_{i=1}^{i=n} K_i)\boldsymbol{y},\ K_i^{**} - K_i^{*T}(\sum_{i=1}^{i=n} K_i)K_i^*) \end{aligned} \tag{6}$$

where $\boldsymbol{f_i^*}$ is the prediction contribution of kernel K_i within the combined prediction $\boldsymbol{f^*} = \sum_{i=1}^{i=n} \boldsymbol{f_i^*}$, given test observations X^*, training observations X and hyperparameters θ.

For the change-point kernel in Equation 5, denoting the complete kernel as K, the contributions of kernel components to the posterior may be retrieved. Firstly, the Squared Exponential kernel K_{SE}, capturing the quasi-static relationship between the flight parameters and wing strain:

$$\begin{aligned} &p(\boldsymbol{f^*_{\text{Quasi-static}}}|\boldsymbol{f^*}, X^*, \boldsymbol{f}, X, \boldsymbol{\theta}) \sim \\ &\mathcal{N}(K_{SE}^{*T}(K)\boldsymbol{y},\ K_{SE}^{**} - K_{SE}^{*T}(K)K_{SE}^*) \end{aligned} \tag{7}$$

The contribution of the SDOF kernel $K_\sigma K_{SDOF}$ aims to capture the dynamic component of the signal, induced due to the manoeuvring of the aircraft, and is expressed:

$$\begin{aligned} &p(\boldsymbol{f^*_{\text{Dynamic}}}|\boldsymbol{f^*}, X^*, \boldsymbol{f}, X, \boldsymbol{\theta}) \sim \\ &\mathcal{N}((K_\sigma K_{SDOF})^{*T}(K)\boldsymbol{y},\ (K_\sigma K_{SDOF})^{**} - (K_\sigma K_{SDOF})^{*T}(K)(K_\sigma K_{SDOF})^*) \end{aligned} \tag{8}$$

Finally, the contribution of noise to the complete posterior is obtained as:

$$\begin{aligned} &p(\boldsymbol{f^*_{\text{Noise}}}|\boldsymbol{f^*}, X^*, \boldsymbol{f}, X, \boldsymbol{\theta}) \sim \\ &\mathcal{N}((\sigma_n^2\delta_{ij})^{*T}(K)\boldsymbol{y},\ (\sigma_n^2\delta_{ij})^{**} - (\sigma_n^2\delta_{ij})^{*T}(K)(\sigma_n^2\delta_{ij})^*) \end{aligned} \tag{9}$$

Used together, the above expressions may provide a breakdown of how the contributions of kernel components change through time. For the case study here, this is of particular interest during the extent of an in-flight manoeuvre. For a zoomed section of time series, centred around an in-flight turn, subplots of kernel posterior contributions, along with how this combine to make the full posterior is shown in Figure 4.

The posterior contributions from the change-point kernel provide an intuitive way to visualise the model's operation, acting as both a sense check, to ensure the model is working as expected, and to provide additional insight alongside results. Firstly, when constructing the model, the intended operation was for the more slowly varying, quasi-static, relationship between the flight parameters and strain to be captured by the SE kernel, with the dynamic, manoeuvre-induced oscillatory behaviour captured by the SDOF kernel. The results of the posterior breakdown verify this, with the majority of the strain magnitude represented through the long lengthscale K_{SE} contribution of the top subplot in Figure 4. The phasing in and out of the $K_\sigma K_{SDOF}$ posterior contribution aligns well with the introduction of dynamic signal content during manoeuvring, with the complete model capturing the oscillatory behaviour well.

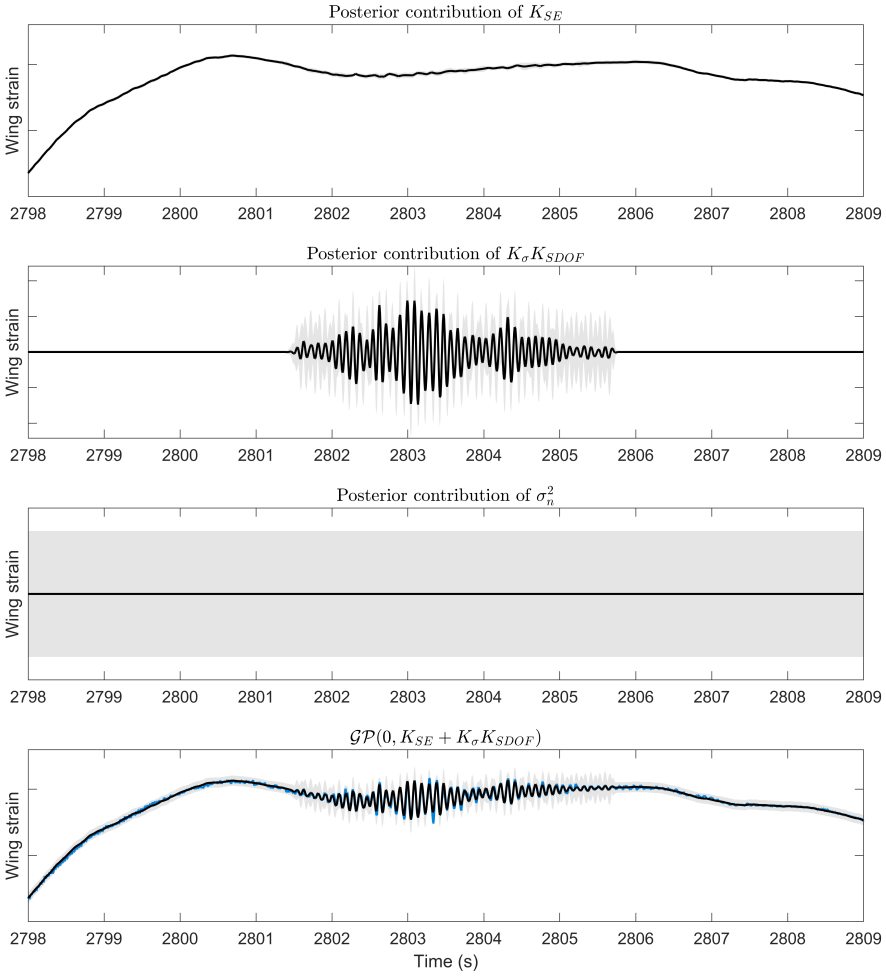


Figure 4: A breakdown of posterior kernel contributions within the change-point kernel defined in Equation 5. The contribution of K_{SE} (Equation 7), the contribution of $K_\sigma K_{SDOF}$ (Equation 8), contribution of the noise (Equation 9) and full posterior are shown.

4 Summary and conclusions

The relative explanatory power of physics and data within a PIML model was explored through an implementation of the physics-informed change-point kernel [19]. This kernel provides a means to incorporate physical knowledge within the kernel of a Gaussian process whilst being able to vary the extent to which the model relies upon this knowledge. This provides a promising tool to cope with both regime-switching and varied presence of physical behaviours.

Data from an aircraft was used to investigate oscillatory behaviour within the wings, induced during in-flight manoeuvring. The behaviour was found to occur during turns of the aircraft (high rudder angles), and was represented using a SDOF kernel within the change point structure. The model was able to effectively upsample from 16Hz to 128Hz and provide a breakdown of static and dynamic signal components. Compared with a purely data-based approach, model performance during the manoeuvre was improved from 48.24% to 6.93%, with performance across the complete time series improving by more than a factor of three.

Acknowledgements

The authors gratefully acknowledge the support of The University of Sheffield and Innovate UK through the OLLGA project grant 10040817.

REFERENCES

[1] E. J. Cross, S. J. Gibson, M. R. Jones, D. J. Pitchforth, S. Zhang, and T. J. Rogers, "Physics-informed machine learning for structural health monitoring," in *Structural Health Monitoring Based on Data Science Techniques*, pp. 347–367, Springer, 2022.

[2] G. Karniadakis, Y. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, "Physics-informed machine learning," *Nature Reviews Physics*, pp. 1–19, 2021.

[3] L. von Rueden, S. Mayer, K. Beckh, B. Georgiev, S. Giesselbach, R. Heese, B. Kirsch, M. Walczak, J. Pfrommer, A. Pick, R. Ramamurthy, J. Garcke, C. Bauckhage, and J. Schuecker, "Informed machine learning - a taxonomy and survey of integrating prior knowledge into learning systems," *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[4] E. J. Cross, T. J. Rogers, D. J. Pitchforth, S. J. Gibson, S. Zhang, and M. R. Jones, "A spectrum of physics-informed Gaussian processes for regression in engineering," *Data-Centric Engineering*, vol. 5, p. e8, 2024.

[5] S. Pfingstl and M. Zimmermann, "On integrating prior knowledge into Gaussian processes for prognostic health monitoring," *Mechanical Systems and Signal Processing*, vol. 171, p. 108917, 2022.

[6] D. An, N. H. Kim, and J.-H. Choi, "Practical options for selecting data-driven or physics-based prognostics algorithms with reviews," *Reliability Engineering & System Safety*, vol. 133, pp. 223–236, 2015.

[7] M. Tóth, A. Brown, E. Cross, T. Rogers, and N. D. Sims, "Resource-efficient machining through physics-informed machine learning," *Procedia CIRP*, vol. 117, pp. 347–352, 2023. 19th CIRP Conference on Modeling of Machining Operations.

[8] D. Kong, Y. Chen, and N. Li, "Gaussian process regression for tool wear prediction," *Mechanical Systems and Signal Processing*, vol. 104, pp. 556–574, 2018.

[9] T. Rogers, P. Gardner, N. Dervilis, K. Worden, A. Maguire, E. Papatheou, and E. Cross, "Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian process regression," *Renewable Energy*, vol. 148, pp. 1124–1136, 2020.

[10] L. Bull, P. Gardner, T. Rogers, N. Dervilis, E. Cross, E. Papatheou, A. Maguire, C. Campos, and K. Worden, "Bayesian modelling of multivalued power curves from an operational wind farm," *Mechanical Systems and Signal Processing*, vol. 169, p. 108530, 2022.

[11] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for Machine Learning*. The MIT Press, 2005.

[12] E. Schulz, M. Speekenbrink, and A. Krause, "A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions," *Journal of Mathematical Psychology*, vol. 85, pp. 1–16, 2018.

[13] E. J. Cross and T. J. Rogers, "Physics-derived covariance functions for machine learning in structural dynamics," *IFAC-PapersOnLine*, vol. 54, pp. 168–173, 2021. 19th IFAC Symposium on System Identification SYSID 2021.

[14] M. Kok and A. Solin, "Scalable magnetic field slam in 3d using Gaussian process maps," in *2018 21st International Conference on Information Fusion (FUSION)*, pp. 1353–1360, 2018.

[15] M. M. Noack and J. A. Sethian, "Advanced stationary and nonstationary kernel designs for domain-aware Gaussian processes," *Communications in Applied Mathematics and Computational Science*, vol. 17, 2022.

[16] M. Haywood-Alexander, N. Dervilis, K. Worden, E. J. Cross, R. S. Mills, and T. J. Rogers, "Structured machine learning tools for modelling characteristics of guided waves," *Mechanical Systems and Signal Processing*, vol. 156, p. 107628, 2021.

[17] S. Klus, P. Gelß, F. Nüske, and F. Noé, "Symmetric and antisymmetric kernels for machine learning problems in quantum physics and chemistry," *Machine Learning: Science and Technology*, vol. 2, 2021.

[18] A. Grisafi, D. Wilkins, G. Csányi, and M. Ceriotti, "Symmetry-adapted machine learning for tensorial properties of atomistic systems," *Physical Review Letters*, vol. 120, 2018.

[19] D. J. Pitchforth, M. R. Jones, and E. J. Cross, "Physically-informed change-point kernels for variable levels of physical knowledge inclusion in gaussian processes," in *Proceedings of the 11th European Workshop on Structural Health Monitoring (EWSHM 2024)*, 2024.

[20] A. G. Wilson, "The change point kernel," tech. rep., 2013.

[21] A. G. Wilson, *Covariance kernels for fast automatic pattern discovery and extrapolation with Gaussian processes*. PhD thesis, University of Cambridge, 2014.

[22] S. C. Reed, "Development of a parametric-based indirect aircraft structural usage monitoring system using artificial neural networks," *The Aeronautical Journal*, vol. 111, p. 209–230, 2007.

[23] R. Fuentes, E. Cross, A. Halfpenny, K. Worden, and R. Barthorpe, "Aircraft parametric structural load monitoring using gaussian process regression," in *Proceedings of the European Workshop on Structural Health Monitoring*, 2014.

[24] S. J. Gibson, T. J. Rogers, and E. J. Cross, "Distributions of fatigue damage from data-driven strain prediction using gaussian process regression," *Structural Health Monitoring*, vol. 22, pp. 3065–3076, 2023.

[25] D. J. Pitchforth, T. J. Rogers, U. T. Tygesen, and E. J. Cross, "Incorporation of partial physical knowledge within gaussian processes," in *Proceedings of the 30th International Conference on Noise and Vibration Engineering (ISMA 2022)*, 2022.

[26] M. R. Jones, D. J. Pitchforth, and E. J. Cross, "Partially structured gaussian processes for grey-box learning in shm," in *Proceedings of the 14th International Workshop on Structural Health Monitoring (IWSHM 2023)*, 2023.

[27] D. Duvenaud, *Automatic model construction with Gaussian processes*. PhD thesis, University of Cambridge, 2014.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

PREDICTING BUILDING OPERATIONAL ENERGY UNDER MATERIAL DEGRADATION AND CLIMATE UNCERTAINTY: A SENSITIVITY ANALYSIS

Anna Maria Koniari[1], Charalampos P. Andriotis[1], Simona Bianchi[1], Pablo G. Morato[1], Seyran Khademi[1], and Mauro Overend[1]

[1] Faculty of Architecture and the Built Environment, Delft University of Technology

Julianalaan 134, 2628BL, Delft
e-mail: {a.m.koniari, c.andriotis, s.bianchi, p.g.moratodominguez, s.khademi, m.overend}@tudelft.nl

Abstract

Building energy prediction models expedite performance assessment and assist in decision-making, from early-stage design to retrofit planning at single- or multi-building scales. However, the number of parameters involved in the energy performance evaluation often impede the prediction process requiring the assimilation of high-dimensional, uncertain input. This is compounded further at multi-building scale e.g. urban energy modelling, due to the increased complexity of evaluating diverse building geometries. While single-building sensitivity and uncertainty analysis is well-established for identifying the most influential input parameters and evaluate the uncertainty effects on energy demand, these are hard to generalize at multi-building scale which remains relatively unexplored. The present study advances existing research by applying a variance-based sensitivity analysis to assess the impact of varying (i) building façade layout, (ii) envelope thermal properties, (iii) envelope air tightness and (iv) building occupancy. The analysis is conducted for multiple buildings under two future climate variations, while also considering the degradation of material thermal properties. The latter is derived from known deterioration models for single-building uncertainty propagation, relying on experimental and simulated data. The approach is applied to a temperate oceanic climate with particular focus on the Dutch building stock, including a sample of buildings with diverse geometric characteristics in Rotterdam. First-order Sobol indices are computed to evaluate the impact with respect to the heating, cooling and total energy demand. Our findings indicate that infiltration is the most influential factor for heating energy demand, whereas cooling is mostly affected by the envelope thermal properties and, particularly, window solar heat gain coefficient. Common patterns regarding the impact of insulation across different envelope components can be identified among buildings with similar orientation and compactness ratio indicating the importance of considering these geometric properties in retrofit decision-making workflows.

Keywords: sensitivity analysis, uncertainty propagation, building energy performance, climate change, material degradation.

1 INTRODUCTION

One third of the global energy consumption currently stems from the operational energy needs of buildings [1], making the building industry a key sector in the efforts to reduce carbon emissions. Energy upgrading of the building stock is essential since, in the EU, 80% of the current buildings is expected to be still in place by 2050 [2], but most of them are currently energy inefficient [3]. Moreover, ageing degradation of insulation is expected to cause long-term decay in thermal performance of the building envelope [4, 5], thus influencing the building energy performance [6]. In this direction, retrofit decision-making frameworks are needed to achieve maximum energy efficiency considering multi-faceted criteria [7]. These often rely on high-fidelity simulators and surrogate models for energy performance prediction [8], requiring a multitude of input parameters and associated uncertainties that can be hard to obtain [9].

Therefore, sensitivity analysis (SA) studies have been developed to identify the most impactful parameters for energy demand [10].

So far, the research has revolved around building sub-systems such as the building envelope or single-buildings [9, 11], while few SA studies examine the impact on diverse geometries [12] and urban-scale applications [13, 14]. Besides building-specific parameters [9], the effect of weather has also been evaluated, examining different locations [9, 12] or future climate scenarios [15, 16, 17]. Particularly, in cold climates, conventional overinsulating solutions, such as improving wall insulation or installing triple-glazed windows, can prove to be unsuitable in the future, since they can increase overheating risk during summer [18, 19] because of the expected increase in outside temperature due to climate change [20].

From a methodological perspective, SA studies for building energy performance analysis (BEPA) employ both local and global SA techniques, with global techniques being mostly preferred since they allow to explore the entire input space [21]. The most common GSA methods in BEPA are: (i) Morris method [22], which provides a qualitative ranking [13]; (ii) Sobol indices [23], which quantify the relative impact of the input parameters on the output [13]; (iii) hybrid approaches, using Morris method to screen out non-important parameters followed by the Sobol indices computation [9]. Sampling is usually conducted using Monte-Carlo sampling; Latin Hypercube Sampling (LHS) [9]; or Sobol sequence via the Saltelli scheme [13, 24].

In the Netherlands, given that heating outweighs cooling demand because of the cold climate, SA studies have primarily focused on evaluating the most influential parameters for heating energy demand using the standardized rank regression coefficient method [25, 26]. Moreover, evaluation of the impact of the total heating and cooling demand, as calculated according to the Dutch standards, has been conducted using Morris and Sobol methods [27]. The effect of insulation degradation on building energy performance has, however, not yet been examined in the Dutch context.

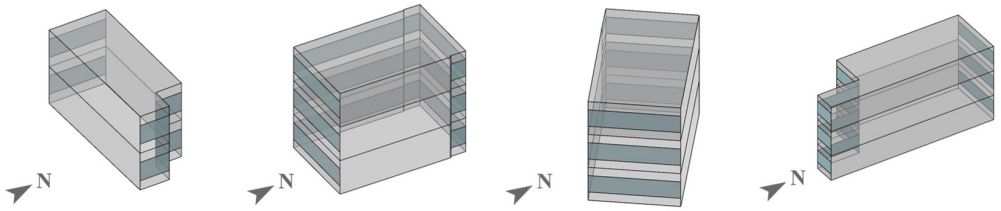
Focusing on the Dutch building stock, this study performs SA to examine the relative importance of building envelope and occupant-related parameters on both heating and cooling annual energy needs individually as well as in the total annual energy outcome. The study expands existing knowledge by identifying common patterns among different outputs and evaluates how these change in future weather conditions. Global sensitivity analysis (GSA) is employed to effectively explore the entire input space and, particularly, first-order Sobol indices are computed to allow for a quantitative comparison among input parameters. Diverse building geometries in Rotterdam are used to further examine how geometric characteristics affect both the Sobol indices and the distribution of the energy results. Moreover, material degradation uncertainties are considered during sampling. The energy results from all model evaluations are compared with the respective results at a pristine state to evaluate the effect of ageing in the energy demand. Overall, key insights are drawn regarding the impact of relevant parameters

and parameter groups on the energy outcome given current and future weather conditions; the effect of geometric characteristics on the parameter ranking; as well as the influence of material degradation in the energy demand in the Dutch context.

2 METHODOLOGY

The sensitivity analysis study is conducted on four existing single-family buildings of Rotterdam, capturing different adjacency types, size, layout and orientation (Table 1). The research methodology is structured into three main parts: (i) selection of example buildings; (ii) identification of input parameters; and (iii) sensitivity analysis (Fig.1).

Table 1: Geometric characteristics of example buildings. (AT: adjacency type (-) ; NF: number of floors (-); A_BF: total building floor area (m^2); AZ_EW: average azimuth of exposed walls (º); CR: compactness ratio (-); FH: floor height (m); A_EW: area of exterior walls (m^2); A_R: area of roof (m^2))



	Building 0 **(B0)**	Building 1 **(B1)**	Building 2 **(B2)**	Building 3 **(B3)**
AT	Terraced	Corner	Terraced	Terraced
NF	2	3	3	3
A_BF	93.90 m^2	190.62 m^2	138.58 m^2	179.70 m^2
AZ_EW	163.59º	179.65º	232.08º	151.64º
CR	1.67	1.73	1.24	1.17
FH	2.98 m	2.81 m	2.22 m	2.24 m
A_EW	62.48 m^2	203.37 m^2	80.06 m^2	90.81 m^2
A_R	46.95 m^2	63.54 m^2	46.19 m^2	59.90 m^2

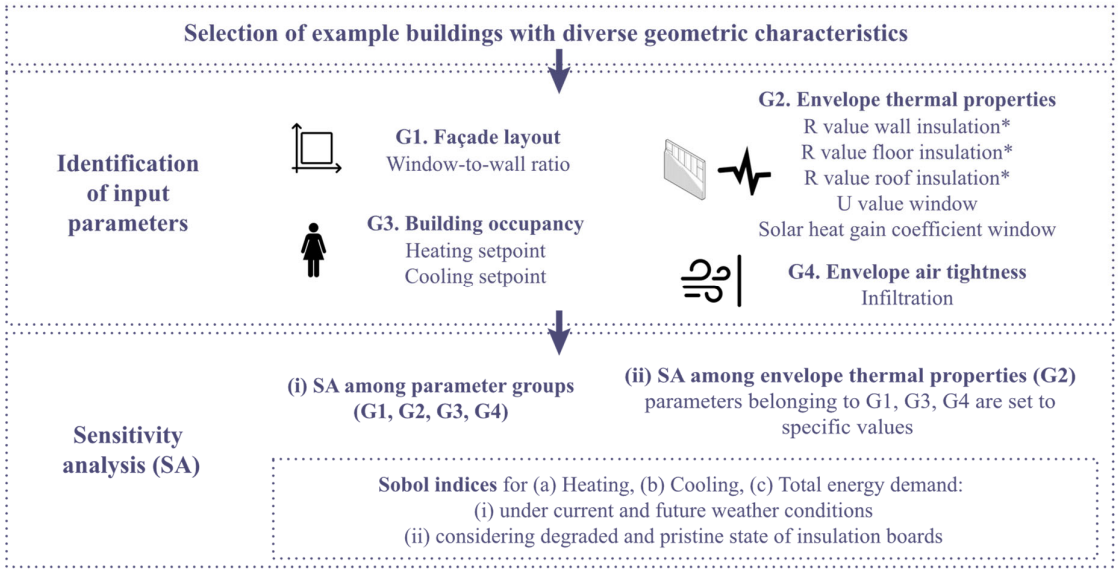


Figure 1:Schematic sensitivity analysis workflow. Input variables marked with an asterisk are calculated considering the degradation effect on insulation performance.

2.1 Selection of example buildings

Specifically, the example buildings consist of terraced houses (B0, B2, B3), which are situated in a row between two neighboring geometries, and a corner house (B1), which is placed at the end of the row and has only one adjacent building. The exposed walls and openings of B0, B2, B3 are in different orientations, with B0 being aligned in the east-west axis; B3 in the north-south; and B2 having an intermediate orientation, with exposed walls placed in south-east and north-west directions. Moreover, although B3 and B2 have the largest and second-largest total building floor area respectively, they are also the most compact when compared to B0, given their relatively smaller floor height. Lastly, unlike the other 3 buildings, B1 has only one neighboring building on the east side and all other wall surfaces are exposed to exterior weather conditions. Given the large area of exposed envelope surfaces, B1 is the least compact building, while, additionally, it has the largest total floor area of all example buildings.

2.2 Simulation setup

The building energy performance simulations are conducted using EnergyPlus [28] and the respective input data files (.idf) are generated using Eppy [29] and GeomEppy [30] Python libraries. All buildings correspond to single-family dwellings without individual apartments on each level. Therefore, although different thermal zones per floor are considered in the simulation, the final energy outcome is computed from the cumulative sum of the energy across all floors normalized by the total building floor area.

The building geometries are reconstructed in LoD (Level of Detail) 1.2, i.e. assuming a uniform volume extrusion with planar roofs and no height variations, using open-source data from the 3DBAG dataset [31]. Due to unavailability of data related to façade layout, all openings are generated in a uniform way using the presumed window-to-wall ratio. Specifically, they are created as horizontal openings in all walls that are exposed to exterior weather conditions.

Adiabatic conditions are assumed for interior floors, ceilings and wall surfaces that are adjacent to neighboring buildings, while all other surfaces are modelled as exposed to exterior conditions. Under the ground floor slab, the soil temperature is set to 18ºC throughout the year. This reflects the special case of soil under a conditioned slab, avoiding any extreme ground temperatures that can result in misleading building losses [32]. The effect of shading from neighboring buildings is not taken into account in the simulations. Lastly, mechanical ventilation is not considered and an ideal system with infinite capacity is assumed instead.

The computations follow the Dutch standards [33], which define that heating demand is evaluated only between October-March, while cooling is assessed between April-September. The annual total energy demand is the sum total of annual heating and cooling energy needs. Two different weather conditions are evaluated in the sensitivity analysis study; current and future weather variations (Fig.2). Specifically, the Typical Meteorological Year (TMY) weather file for Rotterdam-The Hague [34] is selected to represent current weather conditions, capturing interannual variability in the climate input. Future weather data are generated by applying the morphing technique through the future weather generator [35]. In this case, the TMY data are used as base to perform the morphing according to SSP 5-8.5 as defined from EC-Earth-3 climate model [36] for year 2050, indicating a worst-case scenario in which the rate of carbon emissions is not reduced in the long-term. Due to the nature of the morphing technique, which generates future data through adding or multiplying on top of historical weather values, trends in weather patterns are assumed to be maintained in future periods (Fig.2).

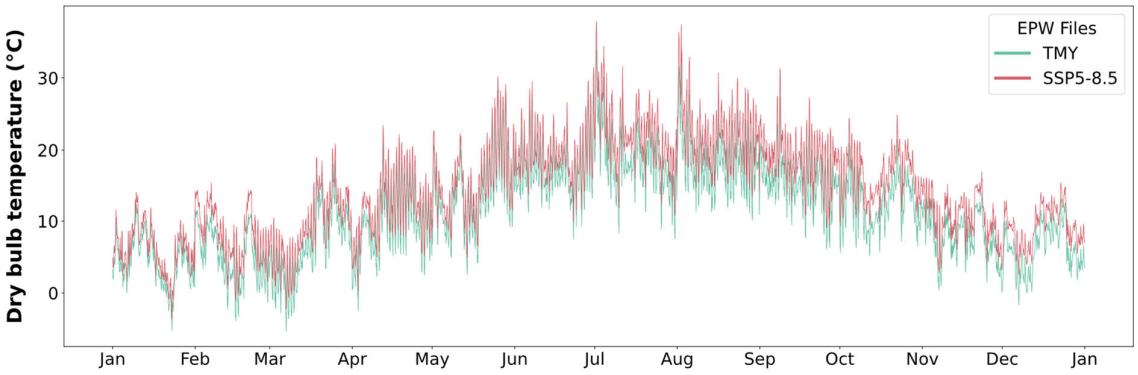


Figure 2: Hourly dry bulb temperature patterns for Rotterdam-The Hague during the course of a year according to TMY and SSP 5-8.5 data.

2.3 Sensitivity analysis setup

The input parameters X_i that are considered during sampling as well as their probabilistic distributions are defined based on literature and belong to the following groups (Table 2):

G_1. Façade layout. The variations in this group are expressed through window-to-wall (WWR) ratio (X_1) uncertainties. Given the use of existing buildings in this study, all other geometric characteristics, such as building layout, orientation and size, are extracted from open-source GIS data [31]. However, the distribution of openings in the façade is not directly available and, thus, assumptions are made regarding the WWR. In this study, the ranges used for the probabilistic distributions vary for terraced and corner houses and are based on information extracted from the Energy module of Woon2018 survey [37] and presented in [38].

G_2. Envelope thermal properties, including the thermal resistance (R value) of the insulation in opaque envelope components, i.e. wall (X_{2a}), roof (X_{3a}) and ground floor (X_{4a}); the effect of ageing in the thermal conductivity of insulation boards (X_{2b}, X_{3b}, X_{4b}); the thermal conductivity (U value) and solar heat gain coefficient (SHGC) of window glazing (X_5, X_6).

Apart from the ageing factor, the probabilistic distributions for all envelope thermal properties are uniform with minimum and maximum values defined based on typical archetype values [39] and potential retrofits designated particularly for the Dutch housing stock [40]. Specifically, in the case of insulation for opaque envelope components, the retrofits represent polyisocyanurate (PIR) insulation panels with varying material thicknesses. The minimum threshold of 2.7 m^2K/W (corresponding to the thermal resistance of an insulation board with thickness 80 mm) is further adjusted to 0.1 in (X_{2a}, X_{3a}, X_{4a}), capturing the possibility of the envelope components being uninsulated. Lastly, window thermal conductivity (U) values span a range from vacuum glazing to triple, double, and single glazing, while solar heat gain coefficient values do not consider the effect of shading and cover a range from clear single glazing to triple glazing with low-emissive coating.

The ageing factor is calculated accounting for an operation period of 20 years. By accounting for different weather files in the evaluation, the study captures both the option of buildings that were retrofitted in the past and have aged and the option of buildings being retrofitted today and aging over the next 20 years. In both cases, the ageing factor is defined following the approach described in [5].

$$d_{ins,20} \sim \mathcal{N}\left(\hat{d}_{ins,20}, \frac{\hat{d}_{ins,20} - 1}{3}\right) \quad (1)$$

where $d_{ins,20}$ is the sampled ageing factor; and $\hat{d}_{ins,20}$ is the mean ageing factor at year 20, defined using the Agesim software [41]. In the software calculations, loss of thermal performance in PIR insulation boards results only from alterations in gas phase conduction due to progressive gas diffusion. The standard deviation of the distribution is defined as a function of the mean ageing factor, thus progressively increasing over time. The thermal resistance of the insulation $r_{i,20}$ at each envelope component i (i.e. floor, wall, roof) is computed as:

$$r_{i,20} = r_{init,i} \cdot \frac{1}{d_{ins,20}} \quad (2)$$

where $r_{init,i}$ is the thermal resistance of the insulation in the i component at a pristine state. Since the ageing factor is defined for thermal conductivity, which is inversely proportional to thermal resistance, the inverse fraction of the ageing factor is considered in the calculation.

G_3. Envelope air tightness, including only infiltration (X_7) without accounting for uncertainties in natural ventilation schedules. Similarly to the thermal parameters, the interval of the uniform distribution is determined based on typical archetype values and potential retrofit interventions.

G_4. Building occupancy. The parameters here express uncertainties around the fixed values of heating and cooling thermostat setpoints (X_8, X_9), which can occur as a result of the building operation schedule.

Table 2: Random variables and probabilistic distributions employed in sensitivity analysis study.

Group	Variable	Distribution	Unit	Reference
G_1. Façade layout	X_1.Window-to-wall ratio	Uniform (Range: 0.43-0.66 for terraced houses, 0.20-0.30 for detached & corner houses	-	[38]
G_2. Envelope thermal properties	X_{2a}. R value of wall insulation at pristine state	Uniform (Range: 0.1-8.3)	m^2K/W	[40]
	X_{2b}. Ageing factor of wall insulation after 20 years	Normal (μ: 1.243, σ: 0.081)	-	[5]
	X_{3a}. R value of roof insulation at pristine state	Uniform (Range: 0.1-8.3)	m^2K/W	[40]
	X_{3b}. Ageing factor of roof insulation after 20 years	Normal (μ: 1.243, σ: 0.081)	-	[5]
	X_{4a}. R value of ground floor insulation at pristine state	Uniform (Range: 0.1-5.7)	m^2K/W	[40]
	X_{4b}. Ageing factor of ground floor insulation after 20 years	Normal (μ: 1.243, σ: 0.081)	-	[5]
	X_5. U value of window glazing	Uniform (Range: 0.6-2.9)	W/m^2K	[40, 42]
	X_6. Solar heat gain coefficient of window glazing	Uniform (Range: 0.5-0.9)	-	[13, 15, 27]
G_3. Envelope air tightness	X_7. Infiltration	Uniform (Range: 0.0004-0.003)	$\frac{m^3}{s \cdot m^2}$	[38, 39]
G_4. Building occupancy	X_8. Heating setpoint	Normal (μ: 20, σ: 0.7)	$°C$	[33, 43]
	X_9. Cooling setpoint	Normal (μ: 24, σ: 0.7)	$°C$	[33, 43]

2.4 First-order Sobol indices calculation

The sensitivity study is conducted into two phases: (i) assessing the impact of each parameter group $G_1 - G_4$ through grouped sampling and (ii) evaluating the relative impact of the individual envelope thermal parameters $X_{2a} - X_6$ when the input variables belonging to all other groups (G_1, G_3, G_4) are fixed. Specifically, in the second case, WWR and thermostat setpoints are fixed to the mean values of their probabilistic distributions, whereas infiltration is set to a slightly improved value, corresponding to a retrofit of crack sealing ($0.0007\ m^3/(s \cdot m^2)$) [38]. In this regard, the importance of interventions in the different envelope components is isolated and insights can be drawn on which interventions are important for retrofit decision-making workflows in order to mitigate heating, cooling and total energy needs. In each opaque envelope component, i.e. wall, roof and ground floor, values for insulation thermal resistance and ageing factor are grouped together during sampling and, thus, first-order Sobol indices are assigned per component (X_2, X_3, X_4).

In this study, sampling and first-order Sobol indices computation are conducted using the SALib Python library [44]. Specifically, sampling is performed using Sobol sequence [23] via the Saltelli scheme [24] to improve input space coverage. The total number of model evaluations for each building is defined as $N(2k + 2)$, where N corresponds to the number of samples and k corresponds to the number of sampling variables and/or groups, following the approach described in [45]. Due to computational time restrictions, 1,024 samples are drawn for each building in each of the two phases. Subsequently, 10,240 model evaluations are conducted per building in the case of grouped sampling ($k = 4$), and 12,288 model evaluations in the second phase, when only envelope thermal properties are assessed ($k = 5$). Confidence intervals are calculated using bootstrapping in order to assess the precision of the estimated values.

The relative impact of each input variable is evaluated using first-order Sobol indices which are computed following the approach described in [24]. Total-order Sobol indices are not presented in this study, since the higher-order interactions among variables proved to be negligible. The first-order Sobol index S_i for each variable X_i is defined as:

$$S_i = \frac{V_{X_i}(E_{\boldsymbol{X}_{\sim i}}(g(\boldsymbol{X}) \mid X_i))}{V\,[g(\boldsymbol{X})]} \tag{3}$$

where $g(\boldsymbol{X})$ is the output function; $E_{\boldsymbol{X}_{\sim i}}(g(\boldsymbol{X}) \mid X_i)$ is the expected value of the output function with respect to all input variables except X_i when X_i is fixed; $V_{X_i}(E_{\boldsymbol{X}_{\sim i}}(g(\boldsymbol{X}) \mid X_i))$ is the variance of the expected value when different values of X_i are assessed; and $V\,[g(\boldsymbol{X})]$ is the total variance of the output function. In this study, three main output functions are considered $g_h(\boldsymbol{X})$, $g_c(\boldsymbol{X})$ and $g_t(\boldsymbol{X})$, which represent the annual heating, cooling and total building energy demand respectively. All values are normalized with respect to the total building floor area.

3 RESULTS

The first-order Sobol indices (SIs) showing the impact of the different parameters and parameter groups along with the 95% confidence intervals are illustrated in Fig.3. Moreover, Fig.4 demonstrates the first-order Sobol indices computed for the variables belonging to G_2 without considering material degradation. Lastly, Fig. 5 shows a summary of the distribution of energy results among all model evaluations per building, while Fig. 6 provides a comparative analysis between the energy results corresponding to degraded and pristine material state.

When examining the influence of different parameter groups, it is observed that envelope air tightness (G_3) has the largest impact on both heating and total energy demand. In contrast, cooling is mostly affected by envelope thermal properties (G_2), followed by façade layout (G_1)

and air tightness (G_3) which are ranked equally. Building occupancy (G_4) has minor impact (SI < 0.15) in all buildings. Interestingly, the median and variance of cooling demand are significantly smaller than the heating and total energy demand distributions (Fig. 5).

In future weather conditions, the influence of all parameter groups ($G_1 - G_4$) on heating demand remains largely unchanged. However, the impact of envelope thermal properties (G_2) and façade layout (G_1) on cooling demand increases over time, resulting in their ranking as the highest and second-highest parameters, followed by building occupancy (G_4). The impact of envelope air tightness (G_3) on total energy demand also decreases in future weather conditions. In particular, the respective SIs show a greater reduction in B0 and B2, while in B1 it is maintained above 0.9. However, G_3 remains the most important parameter group in all buildings. This can be explained by the fact that heating remains overall larger than cooling demand regardless of the expected temperature increase due to climate change (Fig.4). Therefore, the parameters that influence heating remain the most influential for total energy demand as well. However, it can overall be observed that both median and variance of heating demand in future weather decrease when compared to current weather conditions, while the respective properties of cooling demand distributions increase in all buildings.

Although all buildings show uniform behavior in the grouped analysis, different patterns among example buildings can be observed in the second part of the study. Specifically, in current weather conditions, the heating demand of terraced houses (B0, B2, B3) is mostly influenced by the U value of window glazing (X_5), followed by the thermal resistance of roof (X_3) and wall (X_2) insulation respectively. However, X_2 becomes the most important feature in the case of B1, followed by the X_5 and X_3. In contrast, the cooling demand in all buildings is driven primarily by the solar heat gain coefficient of window glazing (X_6). The second most important parameter for cooling is X_5 for terraced houses and X_2 in the case of B1. Unlike heating which is not affected at all by the thermal resistance of the ground floor insulation (X_4), X_4 has a slight impact in cooling demand. However, X_3 does not affect at all the cooling energy outcome.

The observed trends vary in total energy demand. In buildings B0 and B2, X_6 is the most influential parameter, while X_3 and X_5 are the next most impactful variables. On the other hand, in B3, X_3 and X_5 are the most significant factors, with the third being X_6. This is likely due to the building orientation that leaves openings in the north and south sides, thus resulting in smaller impact of X_6 due to alterations in the solar heat gain that is received from the windows. In B1, X_2 has the largest SI, which can be explained by the larger area of exposed walls in this building (Table 1). X_4 has the lowest SI for total energy demand in all buildings.

Similar trends are repeated for heating and cooling in future weather conditions, with X_6 having an even larger impact on cooling compared to current weather conditions. Additionally, X_6 also becomes the determining factor for total energy demand in all terraced buildings, which can be explained by the fact that cooling almost outweighs heating in future conditions. Notably, B0 and B2, which have openings in east and west orientation, have the highest SI values for X_6. However, the SI for X_6 is smaller in the case B1 and almost matches the value of X_2, since the exposed wall area in this case is larger compared to the terraced houses. In fact, X_2 and X_4 are the least important parameter for total energy in B0, B2 and B3.

Comparing with the Sobol indices at the pristine state (Fig. 4), it can be observed that X_2 and X_3 have a larger impact on heating and total energy demand when degradation is considered. In contrast, X_5 and X_6 have overall larger impact in heating and total energy demand when all components are at their pristine state. All parameters have similar SIs in the case of cooling.

Notably, the variance of both heating and total energy demand are considerably smaller when infiltration is not varying, reflecting the large impact of this parameter in the outcome (Fig. 5). Moreover, the median of heating and total energy demand is decreased in all buildings

when compared to the grouped analysis, but cooling distributions remain almost unchanged. This reflects the small impact of infiltration in cooling demand, as demonstrated through the grouped analysis. In the analysis focusing on envelope thermal properties, total energy demand distributions remain roughly unchanged between current and future weather conditions, since the increase in cooling demand negates the reduction in heating due to climate change.

Lastly, in order to evaluate the effect of that insulation performance degradation on the energy outcome, the results of simulations with the same input values both with and without the degradation effect are compared. Afterwards, the ratio of the results at degraded state to the ones at pristine state is computed for each of the model evaluations in all buildings (Fig.6). Overall, total energy demand remains approximately unchanged in all cases. However, as a general pattern, heating energy is increased at the degraded state up to 8%, whereas cooling is decreased by up to 9%. The effect of degradation is in general small (< 10%), which is directly related to the fact that, in this study, only ageing due to gas diffusion in insulation of opaque envelope components is considered.

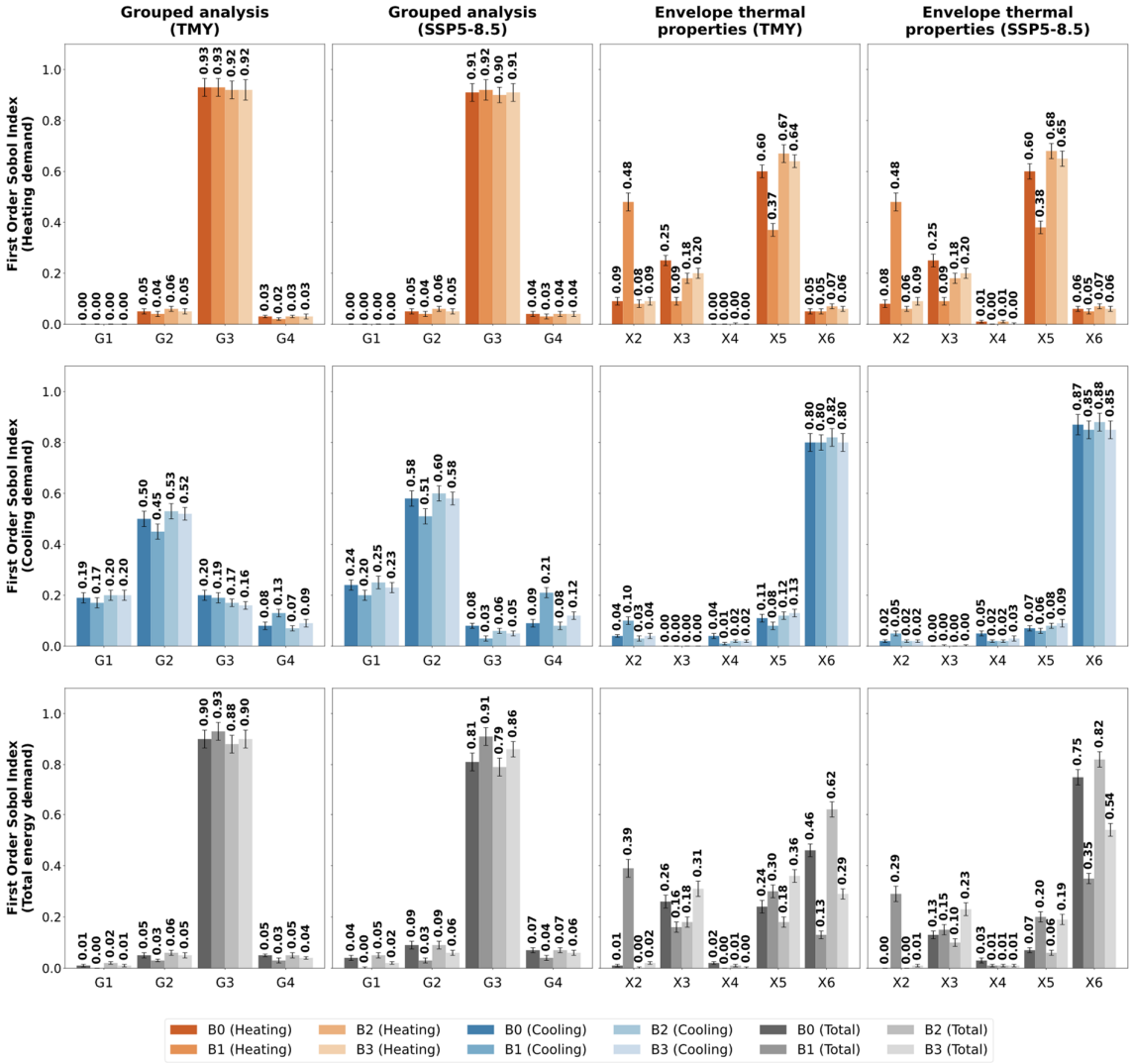


Figure 3: First-order Sobol indices showing the impact of distinct parameters and parameter groups in building heating, cooling and total energy demand under current (TMY) and future (SSP 5-8.5) weather conditions. The error bars represent the 95% confidence intervals computed via bootstrapping.

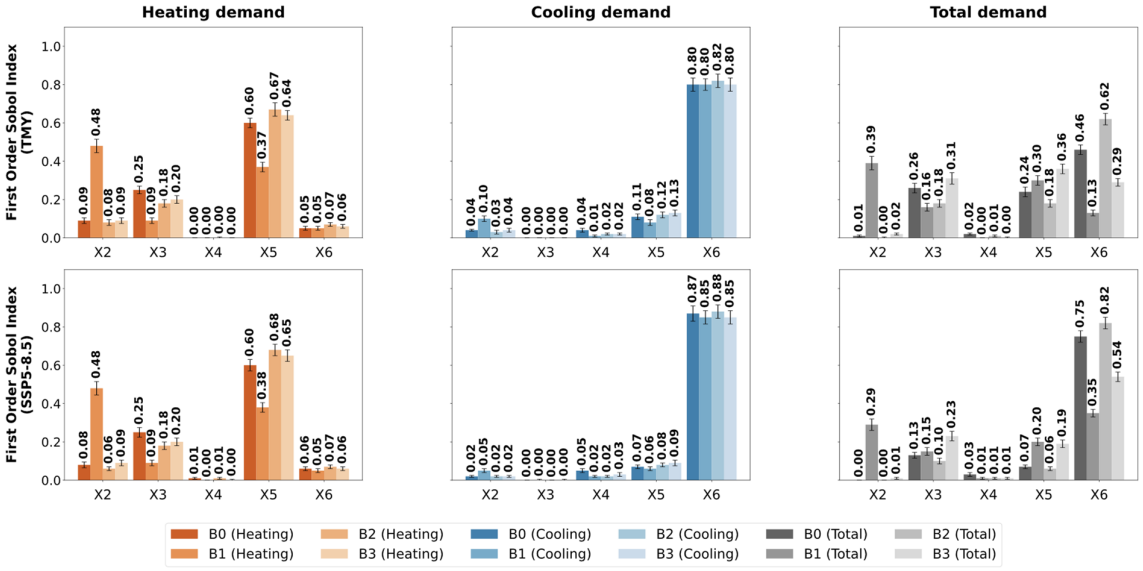


Figure 4: First-order Sobol indices showing the impact of the envelope thermal properties under current (TMY) and future (SSP 5-8.5) weather when all components are at pristine state. The error bars represent the 95% confidence intervals computed via bootstrapping.

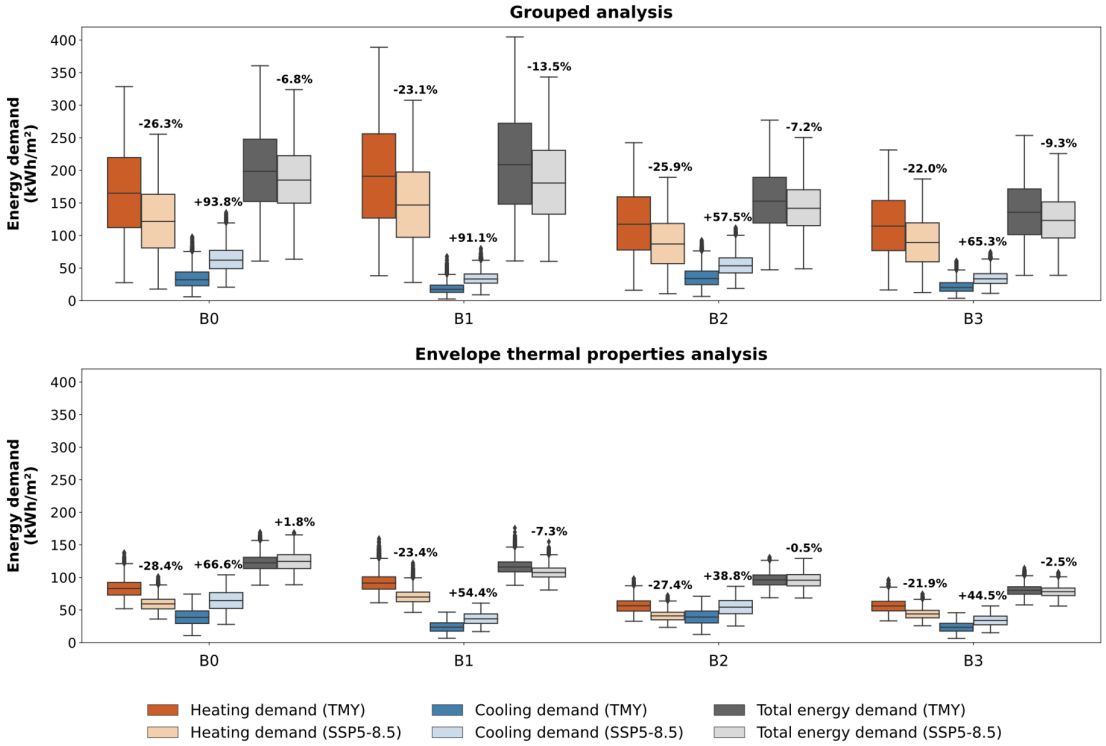


Figure 5: Distribution of heating, cooling and total energy demand results among all model evaluations per building. The boxes, whiskers and lines depict the minimum, first quartile, median, third quartile and maximum of the energy results distribution, showing the variance of the energy results at each SA run. The annotated percentages show the difference in the medians of each energy type between current and future weather conditions.

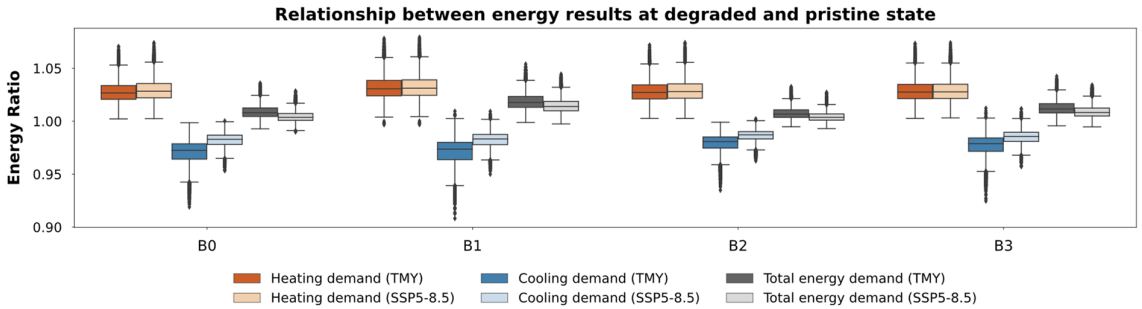


Figure 6: Ratio of heating, cooling and total energy demand at the degraded state with respect to the pristine state.

3.1 Discussion

The first part of the sensitivity analysis focuses on identifying which groups of input variables have the largest impact on heating, cooling and total energy demand in current and future weather conditions. Similarly to other results from literature [26], heating demand is primarily influenced by air infiltration. This result is due to increased air exchange which results in larger heat losses and, subsequently, increases the building heating demand. However, unlike other studies which show that temperature setpoints have a large influence on energy demand [12, 26], this study finds their effect to be marginal in all cases. This can be attributed to the probabilistic distributions used for sampling. In this study, normal distributions are used to assess the effect of uncertainty around the mean setpoint values, which are defined according to the Dutch standards [33]. However, in the other studies [12, 26], uniform distributions are used, evaluating how different setpoint values in an extended interval can affect the energy performance.

Moreover, it is worth emphasizing the change in Sobol indices between heating and cooling demand, given that the importance of input parameters for cooling demand in oceanic climates (mild winters and cool summers) is currently understudied. However, this will gain significant importance over time, given that the share of cooling to total energy demand increases in all cases. Particularly, the most important parameters for cooling are shown to be envelope thermal properties followed by window-to-wall ratio. The effect of infiltration is considerably lower in this case and decreases to an almost insignificant level for cooling demand in future weather. This is also seen from the fact that the median and variance of cooling demand are nearly identical in both grouped and envelope thermal properties analysis, when infiltration is kept constant. This result is directly linked to the way that infiltration is calculated in EnergyPlus [46] accounting for: (a) the stack effect, which occurs because of differences between interior and exterior air density due to temperature and moisture variations; and (b) the effect of wind speed. Both effects are expected to be lower during the summer period. This result is expected to change in the case that natural ventilation effect is taken into account in the simulations.

Although the trends change significantly for cooling, the Sobol indices for total energy demand follow similar patterns to the heating, because all example cases are heating dominated. However, the higher cooling demand in future weather conditions affects slightly the outcome and thus, infiltration is shown to have slightly lower Sobol indices in this case. The effect is particularly evident in B0 and B2 which are aligned to east/west and south-east/north-west directions respectively.

Besides the various differences in geometric properties of all 4 example cases, the ranges of total energy demand between buildings B0-B1 and B2-B3 are similar in both current and future weather scenarios. This is attributed to the similar values of compactness ratio in these pairs of buildings. This agrees with other studies where compactness ratio is shown to affect largely

heating demand in the Dutch context [26, 47]. Nonetheless, the share of cooling to total energy demand is larger in B0 and B2, which is presumably due to their orientation which is mostly aligned to the east/west axis. This is also reflected in these buildings having larger Sobol indices for solar heat gain coefficient and almost dominating cooling over heating energy demand in the detailed analysis in future climate conditions. As a general trend, the energy values for heating are reduced in all buildings in future conditions, whereas cooling demand increases by 38.8%-93.8%, when comparing the median values of the cooling energy distributions.

Besides compactness ratio, the total area of exposed walls also affects the outcome. In this regard, the wall R value is mostly important for B1, whereas in the case of terraced houses, U value of windows and R value of roof are overall more important. However, in this setup, the least influential parameter is the ground floor insulation. This can be linked to the way of simulation modelling, accounting for a constant soil temperature that reflects soil which is situated below a conditioned slab. Although window-to-wall ratio influences only cooling demand, it is considered as important given the increased value of cooling needs in future conditions. In heating demand, the SIs of WWR are surprising, given that the window U value is shown to influence heating in all cases in the second part of the analysis. However, the influence of infiltration is dominating in this case and, therefore, the relative impact of the other parameter groups may be affected accordingly.

Additionally, it is noteworthy that when infiltration, thermostat setpoints and window-to-wall ratio values are constant, the median and variance of energy demand in all buildings is considerably lower. Cooling energy in this case accounts for a larger proportion of the total energy demand and even outweighs heating in B0 and B2 in future weather conditions. Degradation of insulation performance has only small effects on the energy performance when compared to results with materials at pristine state, resulting in increased heating and decreased cooling energy demand because of lower thermal resistance of the building envelope. However, the effect on energy demand is only marginal, which is linked to the fact that, in this study, only ageing due to gas diffusion in insulation boards is considered. Other sources from uncertainty have also been mentioned in the literature, such as loss in effective area of insulation [5]; degradation of insulation performance due to moisture [48] and temperature differences [49]; and weakening of thermal conductivity due to loss of argon in windows [50]. Integrating the various sources of uncertainty in the energy-retrofit decision-making workflow is considered important to allow for a realistic evaluation of energy performance [7].

4 CONCLUSIONS

This study performs sensitivity analysis to identify the most important parameters among geometric, material and occupant-related criteria to include in energy assessment workflows. The study is performed in a temperate oceanic climate under current and future weather conditions to capture the expected shift in heating and cooling needs due to global warming. Among all example buildings, important features to integrate in energy prediction models are: (i) compactness ratio, which influences the total energy needs; (ii) window orientation and window-to-wall ratio, influencing primarily cooling demand. Uncertainties in temperature setpoints only have a marginal effect, unless sampled from uniform distributions that cover diverse human preferences. Overall, improving infiltration is found to be the most important retrofit in the given setup, because it impacts both heating and total energy demand, but its effect on cooling is small, especially in future climate conditions. In contrast, envelope thermal parameters are the most impactful factor for cooling energy demand. Building upon this, when the infiltration is set to a slightly improved value, both the median and variance of the heating and total energy distribution are significantly smaller, whereas cooling is increased, indicating a potentially higher risk of overheating during summer. In this case, heating is mostly affected by the window

U value for terraced and the wall insulation for the corner house, while window solar heat gain coefficient has the largest impact on cooling energy. Moreover, although roof insulation affects heating and total energy demand, it has minimal effect on cooling. Other patterns vary per building, indicating that it is important to consider geometric differences in urban energy retrofit decisions, unlike typical archetype classifications that are usually only based on building adjacency type and construction year. Notably, given fixed infiltration rate, window solar heat gain coefficient has larger impact in the total energy demand for east-west orientation, while wall insulation affects more the total energy in corner houses, given the larger wall area. Ground floor insulation has minimal effect in all cases. Lastly, long-term degradation of insulation performance due to gas diffusion modifies the building energy profile, but other sources of uncertainty should also be included to allow for more realistic energy outcome. Future research can conduct time-dependent sensitivity analysis to identify the importance of different degradation sources along the building lifecycle, incorporate natural ventilation effects and/or evaluate the results for different climate models and weather scenarios.

5 ACKNOWLEDGEMENTS

This material is based upon work supported by the DE-CIST project under the ICLEI Action Fund, the MultiCare EU project (GA no. 101123467), and the TU Delft AI Labs program.

6 REFERENCES

[1] IEA, "Tracking Clean Energy Progress 2023," IEA, Paris, 2023.

[2] World Economic Forum, "Transforming Energy Demand," 2024.

[3] European Union, "Directive (EU) 2024/1275 of the European Parliament and of the Council of 24 April 2024 on the energy performance of buildings (recast)," Official Journal of the European Union, 2024.

[4] D. A. Waddicor, E. Fuentes, L. Sisó, J. Salom, B. Favre, C. Jiménez and M. Azar, "Climate change and building ageing impact on building energy performance and mitigation measures application: A case study in Turin, northern Italy.," *Building and Environment,* vol. 102, 2016.

[5] Z. Liu, X. Zhou, W. Tian, X. Liu and D. Yan, "Impacts of uncertainty in building envelope thermal transmittance on heating/cooling demand in the urban context," *Energy and Buildings,* vol. 273, p. 112363, 2022.

[6] A. Taki and A. Zakharanka, "The Effect of Degradation on Cold Climate Building Energy Performance: A Comparison with Hot Climate Buildings," *Sustainability,* vol. 15, no. 8, p. 6372, 2023.

[7] W.-S. Yun, W. Ryu, D. Lee and H. Seo, "Energy-saving potential estimation of retrofitting aged buildings considering external wall insulation degradation," *Journal of Building Engineering,* vol. 94, p. 110022, 2024.

[8] C.-H. Park and C. S. Park, "Limitations and issues of conventional artificial neural network-based surrogate models for building energy retrofit," *Journal of Building Performance Simulation,* vol. 17, no. 3, pp. 361-370, 2024.

[9] Z. Pang, Z. O'Neill, Y. Li and F. Niu, "The role of sensitivity analysis in the building performance analysis: A critical review," *Energy and Buildings,* vol. 209, p. 109659, 2020.

[10] C. Carpino, R. Bruno, V. Carpino and N. Arcuri, "Improve decision-making process and reduce risks in the energy retrofit of existing buildings through uncertainty and sensitivity analysis," *Energy for Sustainable Development,* vol. 68, pp. 289-307, 2022.

[11] Q. Jin and M. Overend, "Sensitivity of façade performance on early-stage design variables," *Energy and Buildings,* vol. 77, pp. 457-466, 2014.

[12] A. Demir Dilsiz, K. Ng, J. Kämpf and Z. Nagy, "Ranking parameters in urban energy models for various building forms and climates using sensitivity analysis," *Building Simulation,* vol. 16, no. 9, pp. 1587-1600, 2023.

[13] M. Mosteiro-Romero, J. A. Fonseca and A. Schlueter, "Seasonal effects of input parameters in urban-scale building energy simulation," *Energy Procedia,* vol. 122, pp. 433-438, 2017.

[14] M. Y. C. Van Hove, M. Delghust and J. Laverge, "Uncertainty and sensitivity analysis of building-stock energy models: sampling procedure, stock size and Sobol' convergence," *Journal of Building Performance Simulation,* vol. 16, no. 6, pp. 749-771, 2023.

[15] A. J. Ibrahim, D. D. Zangana, S. Liu, H. Samuelson and L. Yang, "Impacts of climate change on energy-saving sensitivity of residential building envelope design parameters in three hot-dry cities," *Journal of Building Engineering,* vol. 99, p. 111481, 2025.

[16] R. Chen, H. Samuelson, Y. Zou, X. Zheng and Y. Cao, "Improving building resilience in the face of future climate uncertainty: A comprehensive framework for enhancing building life cycle performance," *Energy and Buildings,* vol. 302, p. 113761, 2024.

[17] S. Liu, Y. Wang, X. Liu, L. Yang, Y. Zhang and J. He, "How does future climatic uncertainty affect multi-objective building energy retrofit decisions? Evidence from residential buildings in subtropical Hong Kong," *Sustainable Cities and Society,* vol. 92, p. 104482, 2023.

[18] D. R. Liyanage, K. Hewage, S. A. Hussain, F. Razi and R. Sadiq, "Climate adaptation of existing buildings: A critical review on planning energy retrofit strategies for future climate," *Renewable and Sustainable Energy Reviews,* vol. 199, p. 114476, 2024.

[19] U. Y. Ayikoe Tettey and L. Gustavsson, "Energy savings and overheating risk of deep energy renovation of a multi-storey residential building in a cold climate under climate change," *Energy,* vol. 202, 2020.

[20] U. Berardi and P. Jafarpur, "Assessing the impact of climate change on building heating and cooling energy demand in Canada," *Renewable and Sustainable Energy Reviews,* vol. 121, p. 109681, 2020.

[21] W. Tian, "A review of sensitivity analysis methods in building energy analysis," *Renewable and Sustainable Energy Reviews,* vol. 20, pp. 411-419, 2013.

[22] M. Morris, "Factorial Sampling Plans for Preliminary Computational Experiments," *Technometrics,* vol. 33, no. 2, pp. 161-174, 1991.

[23] I. M. Sobol, "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimate," *Mathematics and Computers in Simulation,* vol. 55, no. 1-3, pp. 271-280, 2001.

[24] A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto and S. Tarantola, "Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index," *Computer Physics Communications,* vol. 181, pp. 259-270, 2010.

[25] A. Ioannou and L. Itard, "Energy performance and comfort in residential buildings: Sensitivity for building parameters and occupancy," *Energy and Buildings,* vol. 92, pp. 216-233, 2015.

[26] P. Wahi, T. Konstantinou, H. Visscher and M. J. Tenpierik, "Evaluating building-level parameters for lower-temperature heating readiness: A sampling-based approach to addressing the heterogeneity of Dutch housing stock," *Energy and Buildings,* vol. 322, p. 114703, 2024.

[27] M. Kafaei, "Sensitivity Analysis of NTA8800 for a Dutch Building Renovation Tendering System," 2021.

[28] National Renewable Energy Laboratory (NREL), *EnergyPlus™,* 2017.

[29] "Eppy," 2013. [Online]. Available: https://github.com/santoshphilip/eppy. [Accessed 12 03 2025].

[30] "GeomEppy," [Online]. Available: https://github.com/jamiebull1/geomeppy. [Accessed 12 03 2025].

[31] R. Peters, B. Dukai, S. Vitalis, J. Van Liempt and J. Stoter, "Automated 3D Reconstruction of LoD2 and LoD1 Models for All 10 Million Buildings of the Netherlands," *Photogrammetric Engineering & Remote Sensing,* vol. 88, no. 3, pp. 165-170, 2022.

[32] U.S. Department of Energy, "EnergyPlus version 24.2.0 Documentation Input Output Reference," 2024.

[33] S. K. N. N. Instituut, "Nederlandse technische afspraak NTA 8800:2024 Energieprestatie van gebouwen - Bepalingsmethode," 2024.

[34] "Climate-OneBuilding.Org," [Online]. Available: https://climate.onebuilding.org/. [Accessed 13 03 2025].

[35] E. Rodrigues, M. S. Fernandes and D. Carvalho, "Future weather generator for building performance research: An open-source morphing tool and an application," *Building and Environment,* vol. 233, p. 110104, 2023.

[36] R. Döscher, M. Acosta, A. Alessandri, P. Anthoni, T. Arsouze, T. Bergman, R. Bernardello, S. Boussetta, L.-P. Caron, G. Carver, M. Castrillo, F. Catalano, I. Cvijanovic, P. Davini, E. Dekker, F. J. Doblas-Reyes, D. Docquier, P. Echevarria, U. Fladrich, R. Fuentes-Franco, M. Gröger, J. v. Hardenberg, J. Hieronymus, M. Karami, J.-P. Keskinen, T. Koenigk, R. Makkonen, F. Massonnet, M. Ménégoz, P. A. Miller, E. Moreno-Chamarro, L. Nieradzik, T. van Noije, P. Nolan, D. O'Donnell, P. Ollinaho, G. van den Oord, P. Ortega, O. Prims, A. Ramos, T. Reerink, C. Rousset, Y. Ruprich-Robert, P. Le Sager, T. Schmith, R. Schrödner, F. Serva, V. Sicardi, B. Smith, Tian, T. Tian, E. Tourigny, P. Uotila, M. Vancoppenolle, S. Wang, D. Wårlind, U. Willén, K. Wyser, S. Yang, X. Yepes-Arbós and Q. Zhang, "The EC-Earth3 Earth system model for the Coupled Model Intercomparison Project 6," *Geoscientific Model Development,* vol. 15, no. 7, pp. 2973-3020, 2022.

[37] Woononderzoek Nederland, "WoON 2018".

[38] Rijksdienst voor Ondernemend Nederland, "Voorbeeldwoningen," 2022.

[39] Nieman De Raadgevende Ingenieurs, "Verduurzaming Rotterdamse woningvoorraad," 2023.

[40] [Online]. Available: https://regelhulpenvoorbedrijven.nl/kostenkentallen/. [Accessed 09 February 2025].

[41] S. N. Singh and P. D. Coleman, "Accelerated Aging Test Methods for Predicting the Long Term Thermal Resistance of Closed-Cell Foam Insulation," in *Center for Polyurethanes Industries Conference*, 2007.

[42] "TABULA WebTool," [Online]. Available: https://webtool.building-typology.eu/#bm. [Accessed 26 September 2024].

[43] G. Calleja Rodríguez, A. Carrillo Andrés, F. Domínguez Muñoz, J. M. Cejudo López and Y. Zhang, "Uncertainties and sensitivity analysis in building energy simulation using macroparameters," *Energy and Buildings,* pp. 79-87, 2013.

[44] J. Herman and W. Usher, "SALib: An open-source Python library for Sensitivity Analysis," *The Journal of Open Source Software,* vol. 2, no. 9, p. 97, 2017.

[45] A. Saltelli, "Making best use of model evaluations to compute sensitivity indices," *Computer Physics Communications,* vol. 145, no. 2, pp. 280-297, 2002.

[46] U.S. Department of Energy, "Engineering Reference (EnergyPlus Version 24.2.0 Documentation)," 2024.

[47] Nieman De Raadgevende Ingenieurs, "Rapport standaard en streefwaardes bestaande woningbouw," 2021.

[48] F. Tariku, Y. Shang and S. Moletti, "Thermal performance of flat roof insulation materials: A review of temperature, moisture and aging effects," *Journal of Building Engineering,* vol. 76, no. 2023, p. 107142.

[49] S. Moletti and D. Van Reenen, "Effect of Temperature on Long-Term Thermal Conductivity of Closed-Cell Insulation Materials," *Buildings,* vol. 12, no. 4, p. 425, 2022.

[50] S. K. Asphaug, B. P. Jelle, L. Gullbrekken and S. Uvsløkk, "Accelerated ageing and durability of double-glazed sealed insulating window panes and impact on heating demand in buildings," *Energy and Buildings,* vol. 116, pp. 395-402, 2016.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

PROPAGATING ML MODEL UNCERTAINTIES IN OFFSHORE WIND TURBINE FATIGUE DAMAGE ACCUMULATION

Francisco de Nolasco Santos[1], Nandar Hlaing[1], Wout Weijtjens[1] & Christof Devriendt[1]

[1]OWI-Lab, Vrije Universiteit Brussel
Pleinlaan 2, 1050 Elsene, Belgium
e-mail: {francisco.de.nolasco.santos,nandar.hlaing}@vub.be

Abstract. *Modern offshore wind farm operators are increasingly looking at keeping tabs of the structural lifetime consumption of their assets. As lifetime estimation is closely linked with complex decision-making processes like lifetime extension or certification, there has been a move from a deterministic framework to a modeling which quantifies uncertainties. This is especially relevant during unseen load- or new environmental conditions, where uncertainty quantification can bring light into understanding the severity of these phenomena and how to appropriately deal under uncertain conditions (e.g. discard prediction, retrain model). This has led to data-driven approaches with probabilistic reasoning for fatigue load estimation.*
In this contribution, Bayesian neural networks (BNN) and conformal predictors are compared for fatigue load prediction and uncertainty quantification. These outputs have a ten-minute frequency (industry standard). However, fatigue accumulates non-linearly over time. Thus, ten-minute estimations need to be rescaled and aggregated, along with their corresponding uncertainties to reflect the entire interval under study and, eventually, the lifetime. As such, in this contribution, we attempt to extend this approach by devising a methodology to aggregate and propagate the model uncertainty related to each ten-minute prediction over a one-year period of real-world data from an offshore wind turbine through Monte Carlo Simulations and discuss its impact for lifetime.

Keywords: Offshroe wind farm, Fatigue Loads, Uncertainty Propagation, Bayesian Neural Networks, Conformal predictors

1 Introduction

Offshore wind farms are an integral part of current-day electrical grids. Their participation in the energy market often implies aggressive operation through set-point variation in order to match demand. The way offshore wind turbines are operated has a direct impact on their structural lifetime. For newer farms where design codes have been updated and structural margins decreased, aggressive operation of turbines can jeopardize reaching the intended lifetime. For farms where structural reserve exists – as are many older farms – one can envision lifetime extensions. Therefore, structural fatigue monitoring is crucial for modern wind farm operators to keep tabs on their assets' lifetime. It can also provide further insights for design, as fatigue is a major design driver for structural components of wind turbines.

Structural health monitoring of offshore structures hinges on the use of sparse strain gauge instrumentation and availability of farm-wide data, such as supervisory control and data acquisition (SCADA) or acceleration data. Thus, data-driven models can be trained based on both SCADA and acceleration inputs for a small number of strain-gauged installed turbines, before being employed farm-wide to estimate a metric of fatigue damage, *e.g.* Damage Equivalent Loads (DEL) [1]. These models can provide valuable insights into fatigue accretion in offshore structures, connect it with environmental and operational conditions (EOCs) and allow operators to take informed decisions concerning structural management. By accurately quantifying fatigue damage, operational strategies that attempt to optimally plan wind farm damage over their lifetime can be drawn, minimizing costs [2]. However, a fundamental element is still needed – quantification of the uncertainties related to each prediction. For decisions like lifetime extension, quantifying accurately uncertainty is key and studies must evolve to probabilistic frameworks [3]. As fatigue life reassessment of wind turbine structural components sees actual conditions deviating from the assumptions of the original design documentation, there is much gain in replacing deterministic methods with probabilistic ones. [4]. This has to be extended to fatigue load estimation models themselves, with probabilistic data-driven models like Bayesian Neural Networks (BNN) or frequentist approaches like conformal predictors, as will be introduced in this work. We select to have both approaches in this work as they have different advantages and challenges between them. Importantly, we want to contrast a model which inherently quantifies uncertainty (BNN) with a model-agnostic method (which can be updated as data is aggregated without having to retrain the model) and how this affects time-wise uncertainty propagation. Naturally, model uncertainty is only a small component of the full uncertainty that affects fatigue life consumption (other sources are, *e.g.* uncertainty from measurement systems), but it is still necessary to aggregate this uncertainty over time.

Therefore, in presence of a harsh marine environment and subject to deterioration mechanisms throughout their operational lifetime, monitoring systems allow to reduce uncertainties and grant greater confidence at decisions taken regarding the structural stability of offshore wind turbines. Strain gauges are, however, sparsely instrumented, which makes uncertainty quantification for non-strain gauge instrumented turbines essential to ensure that models trained on a small number of turbines with strain gauges (so-called fleet-leaders) are also applicable farm-wide. However, even for fleet-leaders, uncertainty quantification and uncertainty propagation over time is crucial, as it allows to discern how changes in the environment and control might affect model confidence and accuracy. If fatigue rates are to be effectively translated to lifetime consumption, then propagating uncertainty over time is, again, essential. Finally, these metrics need to be correctly interpreted and communicated.

2 Wind farm probabilistic data-driven load estimation

As mentioned in the introduction, probabilistic approaches are able to increase the reliability of data-driven models, particularly in relation to lifetime. *E.g.*, if environmental or structural conditions change over time, as is the case, respectively, of when wind distributions are altered due to macroclimate changes or for scour [5], then the underlying assumptions of design – and, implicitly, during model training – no longer hold. This change needs to also be reflected in the confidence we deposit in fatigue load estimation models. If structural dynamics fundamentally change, one cannot blindly trust predictions and increased uncertainty effectively correlated with a change on EOCs can help trigger decisions like model retraining or prediction discarding. In our work, we will present two models for wind farm probabilistic fatigue load based on previous works: Bayesian neural networks (BNN) [6] and conformal predictors [7]. Greater detail on the implementation, inputs and architecture of these models can be found in their respective references. We will begin in this section by introducing theoretically both models and showing some examples of probabilistic outputs before focusing on propagating uncertainties over time and how these are affected by EOC changes.

2.1 Bayesian neural networks

A Bayesian neural network (BNN) is a stochastic artificial neural network trained via Bayesian inference, featuring the combined strength of deep learning and Bayesian theory in order to provide a rich probabilistic interpretation of the generated predictions. The key defining characteristic of BNNs is their stochastic model weights representing model's epistemic uncertainty. In many applications, the output prediction of a BNN is also formulated as a probability distribution in order to allow intrinsic modeling of aleatory uncertainty. Whereas aleatory uncertainty is irreducible, model uncertainty can be reduced by adding more training data.
BNNs estimate model parameters using Bayesian inference, which requires computing the posterior distribution $P(\boldsymbol{\theta} \mid D)$ when observing data D. This posterior distribution is obtained via Bayes' theorem:

$$P(\boldsymbol{\theta} \mid D) = \frac{P(D \mid \boldsymbol{\theta})P(\boldsymbol{\theta})}{\int P(D \mid \boldsymbol{\theta})P(\boldsymbol{\theta})d\boldsymbol{\theta}}. \tag{1}$$

However, in continuous probabilistic settings, this computation is intractable due to the complexity of the high-dimensional integral in the denominator. To address this challenge, various approximation techniques have been proposed, such as the Laplace approximation [8], variational inference [9], and Markov Chain Monte Carlo (MCMC) sampling [10].
In this work, we rely on variational inference to train the Bayesian neural networks, in which the unknown posterior $P(\boldsymbol{\theta} \mid D)$ is approximated by a variational distribution $q_\lambda(\boldsymbol{\theta})$. The variational parameters can then be obtained by optimizing the evidence lower bound objective (ELBO) function:

$$\lambda^* = \text{argmin}\{\mathbb{KL}\left(q_\lambda(\boldsymbol{\theta}) \mid\mid P(\boldsymbol{\theta})\right) - \mathbb{E}_{\boldsymbol{\theta}\sim q_\lambda}\left[\log\left(P(D \mid \boldsymbol{\theta})\right)\right]\}. \tag{2}$$

In particular, the objective function consists of two competing factors: regularizing the model weights to remain close to the prior assumption (often zero-centered), and maximizing the probability of the observed training labels to encourage accurate predictions.
An example of the model's predictions can be seen in Figure 1 where one week worth of data is presented (January 2024) for a real operating offshore wind turbine. The expected fore-aft DEL (normal bending moment, DEL_{Mtn}) is plotted, $\mathbb{E}[DEL] = \mu_{DEL}$, and the predictive

uncertainty is indicated as $\pm 2\sigma_{DEL}$ in the figure. In this work, we modeled the BNN assuming the aleatory uncertainty is zero, and therefore, the predictive uncertainty $\pm 2\sigma_{DEL}$ is purely epistemic. For a systematic uncertainty quantification and decomposition for Bayesian neural networks, the interested reader is referred to [6].

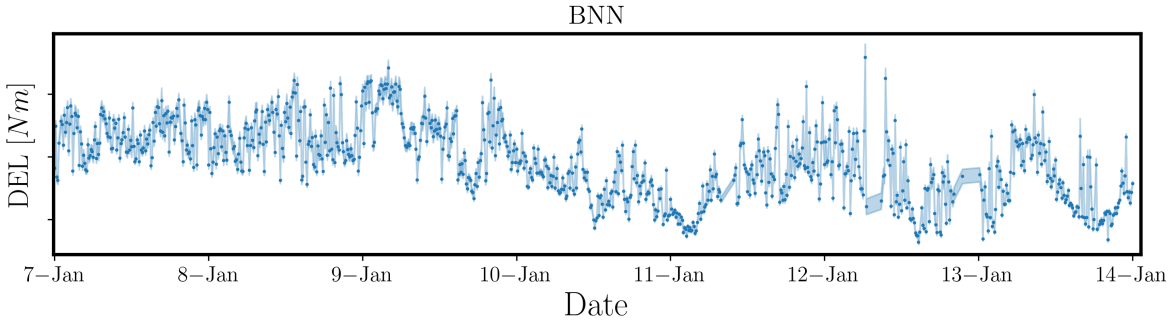


Figure 1: Probabilistic predictions of BNN, $\mathbb{E}[DEL] \pm 2\sigma_{DEL}$.

2.2 Conformal prediction

We saw above how we can explicitly model epistemic uncertainty through the use of Bayesian neural networks, where the learning of these quantities is embedded in the model training. However, what happens if we already have a model with deterministic outputs, *e.g.* the model from [1], but we'd still like to to have an after-prediction uncertainty quantification? Also, given the constraints of BNNs (model needs to be retrained to learn from incoming data), can we discern a methodology which might answer these concerns? Several techniques allow us to achieve this, like bootstrapping [11] or the use of ensembles [12]. However if we want a model-agnostic, *distribution-free* (no prior assumption is made on the data's distribution) uncertainty quantification method, one of the most widely-employed and mathematically sound algorithms is conformal prediction [13] (we'll restrict ourselves to discussing the computational efficient inductive conformal prediction [14]).

2.3 Theoretical introduction

Given an underlying point-wise model (neural network, decision tree, boosting, etc.), conformal prediction *'envelops'* predictions in a prediction region or set $1-\alpha$, where $\alpha \in [0,1]$ stands for the significance level. Its sole assumption is exchangeability — a weaker form of i.d.d. — that is to say, the joint probability distribution of the random variables $x_1, x_2, \ldots, x_n$ is the same as for the permuted indices $P(x_1, x_2, \ldots, x_n) = P(x_{\sigma(1)}, x_{\sigma(2)}, \ldots, x_{\sigma(n)})$, with σ, a finite permutation (*i.e.*, the ordering of variables isn't relevant) [15, 16]. Now take our neural network model f which, given an input $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \in \mathcal{X}$, makes a point prediction $\hat{\mathbf{y}} = \{\hat{y}_1, \ldots, \hat{y}_n\} \in \mathcal{Y}$, with $(\mathcal{X}, \mathcal{Y}) \in \mathbb{R}$. Let us then consider that $(\mathbf{X}, \hat{\mathbf{y}})$ belong to a so-called *'calibration set' — i.e.*, a set unseen during f's training, used for calibrating the conformal prediction model. We can thus, for $(\mathbf{x}_{n+1}, \hat{y}_{n+1})$, construct a prediction set $\mathcal{T}(\mathbf{x}_{n+1})$, valid for:

$$[1-\alpha \leq \mathbb{P}[\hat{y}_{n+1} \in \mathcal{T}(\mathbf{x}_{n+1})] \leq 1-\alpha+\frac{1}{n+1}] \tag{3}$$

In order to construct $\mathcal{T}$, we need to set a conformal score (a nonconformity measure that measures how different a new sample is from old examples). In this case, the nonconformity measure is the absolute error $|y - \hat{y}|$. Thereafter, we compute the quantile $\hat{q} = [(n+1)(1-\alpha)]/n$ of the nonconformity scores, essentially the $1-\alpha$ quantile, where n stands for the number of samples. We can then create the prediction set $\mathcal{T}(\mathbf{x_{n+1}}) = \{y : f(\mathbf{x_{n+1}})_y \geq 1 - \hat{q}\}$, with our prediction being actualized to $\hat{Y}_{n+1} = \hat{y}_{n+1} \pm \mathcal{T}(\mathbf{x}_{n+1})$ [17]. We can observe this procedure in Algorithm 1.

Algorithm 1 Conformalize prediction

for $\mathbf{X}, \mathbf{y}, \hat{\mathbf{y}} \in (\mathcal{X}, \mathcal{Y}, \hat{\mathcal{Y}}) \in \mathbb{R}$ **do** ▷ For each tuple of inputs, ground truth and model outputs
  $\mathbf{s}(\mathbf{X}, \mathbf{y}, \hat{\mathbf{y}}) \in \mathbb{R} = |\mathbf{y} - \hat{\mathbf{y}}|$ ▷ Nonconformity score is the absolute residuals
end for
for $\mathbf{s} = \{s(\mathbf{X_1}, y_1, \hat{y}_1), \ldots, s(\mathbf{X_n}, y_n, \hat{y}_n)\} \in \mathbb{R}$ **do** ▷ For each nonconformity score
  $\hat{q} : [(n+1)(1-\alpha)/n]$ quantile of $\mathbf{s}$ ▷ Compute the quantile
end for
$\mathcal{T}(\mathbf{X_{n+1}}) = \{\hat{y}_{n+1} : \mathbf{s}(x, y) \leq \hat{q}\}$ ▷ Use quantile to form prediction set of new example

However, $\mathcal{T}(\mathbf{x}_{n+1})$ always produces intervals of the same size [18]. In order to normalize the nonconformity measure to the expected accuracy of the underlying model (smaller regions for easier predictions, larger for more difficult)[14], one is required to employ a so-called *'difficulty estimator'* (μ_{n+1} [18]). There are several possible ways to estimate difficulty, but in this work we follow [18] by using the absolute errors of the k nearest neighbors, employing the crepes module [19]. Thus, given a significance level α, our normalized predictions shall be updated to:

$$\hat{Y}^{\alpha}_{n+1} = \hat{y}_{n+1} \pm \mathcal{T}(\mathbf{x}_{n+1})(\mu_{n+1} + \beta) \quad , \tag{4}$$

where β is a nonconformity sensitivity control parameter (0.01).
In this contribution, we begin with the deterministic predictions from the model of [1]. We shall call this model DNN (Deep Neural Network). Then, we implement the above mentioned algorithm. We used one year (October 2022 – September 2023) of data from 5 real-world offshore wind turbines with strain gauges, known as fleet-leaders. This year of data for training does not coincide with the testing data we'll present in the following section. Only 75% of the data is used to train the conformal prediction algorithm and a k-nearest neighbours difficulty estimator with the absolute residuals as the reference objects. Furthermore, we perform a 50% train-test split for the test and calibration datasets. The original DEL predictions are then conformalized for a 95% confidence interval with a minimal value for the predictions of 0 (negative fatigue loads are nonsensical). These results can be observed in Figure 2. When compared with Figure 1, we can see how the confidence interval is wider. This might be because the BNN is modelling epistemic uncertainty (asymptotically, which might produce overly narrow confidence intervals [20]), whereas conformal predictors provide finite-sample coverage guarantees.

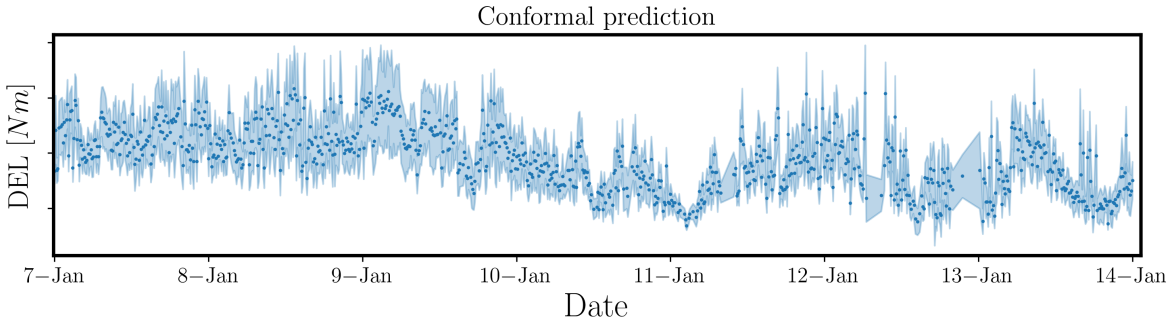


Figure 2: Predicted DEL with 95% conformalized confidence interval.

3 Long-term fatigue damage accumulation and uncertainty quantification

Having described both BNNs and conformalized predictions in section 2, in this section we'll address the long term accumulation of DELs and how uncertainty can similarly propagated over time. As mentioned above and seen in the previous figures, both approaches (BNN and DNN with conformalized predicitons) allow us to not only have estimated values at each ten-minutes, but also a confidence interval related to each prediction. Naturally, fatigue does not restrict itself to ten-minutes, but it is rather a slowly evolving phenomenon over the lifetime. Therefore, ten-minute DEL predictions need to be accumulated long-term. Given that DELs are based on Palmgren-Miner's rule, individual ten-minute DELs cannot simply be added, but, if derived for the same reference cycle number, N_{eq} and Wöhler exponent, m, DELs can be aggregated into a *'representative'* DEL for the period in question through the m-root of the weighted summation of the m-power DEL instance, as seen in Equation 5 [21]:

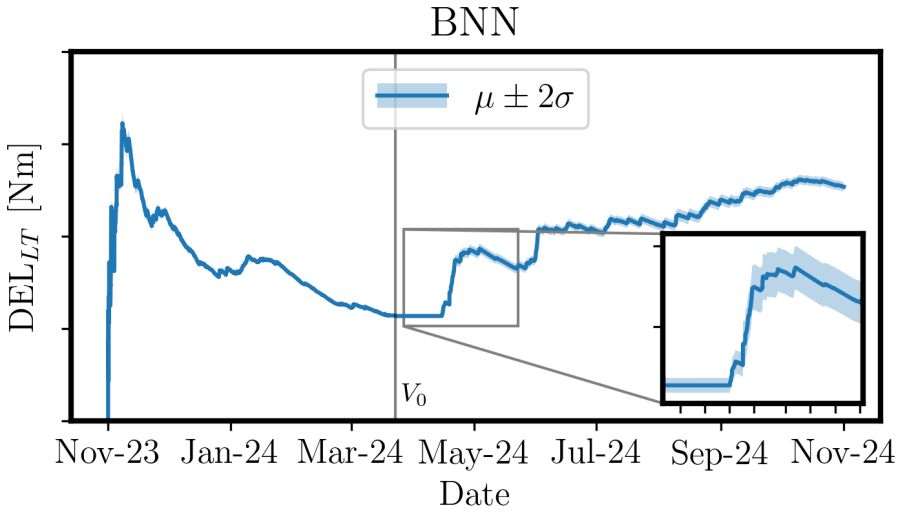
$$DEL_{LT} = \left(\frac{1}{n} \sum_{i=1}^{n} DEL_i^m \right) \quad . \tag{5}$$

Here DEL_{LT} represents the long-term DEL rescaling of n ten-minute DEL instances into a ten-minute DEL representative of the long-term period in question.

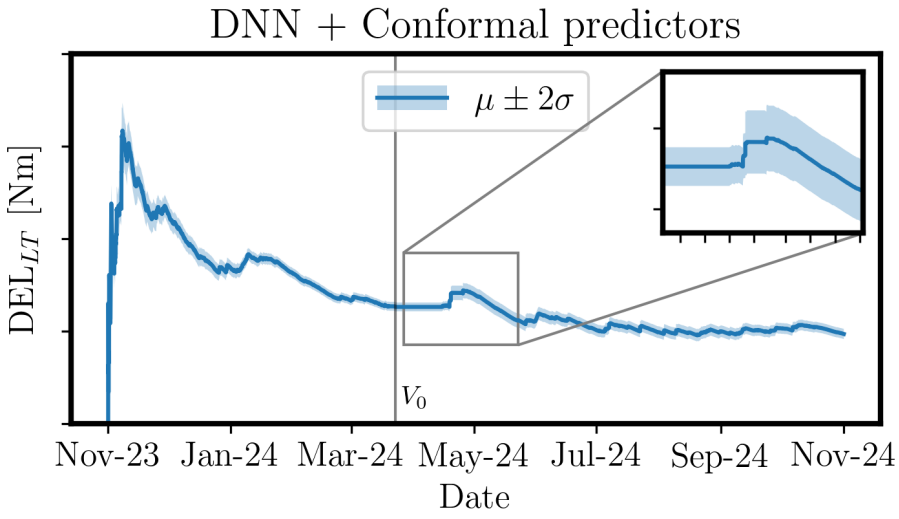
In the probabilistic framework, we employed a Monte Carlo simulation approach (MCS [22]) to accumulate short-term fatigue damages and uncertainties of Bayesian neural networks (BNNs). Within each simulation, short-term random damage values were obtained through forward passes of the BNN and subsequently aggregated using Equation 5. By repeating this process over multiple simulations, we derived the expected long-term DEL along with its uncertainty. However, extending this approach to conformal predictions presents challenges, primarily due to the unknown probability distribution within the estimated damage interval. As a preliminary solution, we addressed this issue by assuming that each ten-minute damage follows a truncated normal distribution within the given interval, enabling the accumulation of conformal predicted damages through MCS. Furthermore, given the flexibility of conformal prediction, which doesn't require model retraining to update its confidence intervals, we re-estimate the nonconformity measure and difficulty estimator at each month.

In this contribution, we employed MCS to a period of one year of data from a strain gauge-installed real-world wind turbine in the Belgian offshore. Both models were trained with non-overlapping data, with relation to the test data set. They use data from a SCADA system which records turbine's environmental and operational conditions and nacelle-installed accelerome-

ters as inputs and bending moments (we focus on the normal component in this contribution, *i.e.* fore-aft) from tri-axial strain gauges on the transition piece for the ground truth. We can see the results for both BNN and DNN with conformalized predictions in Figure 3a and Figure 3b where one year of representatives DELs (DEL_{LT}) are progressively aggregated over time, along with a 95% confidence interval ($\pm 2\sigma$).



(a)



(b)

Figure 3: Comparison of long-term representative DEL (DEL_{LT}) and corresponding sampled 95% confidence interval based on BNN (a) and DNN conformalized predictions (b). V_0, turbine control changes moment.

In Figure 3 we observe how the progressive aggregation of ten-minute DELs into a long-term DEL initially shoots up, before slowly starting to converge until V_0. The same mechanism oc-

curs for the aggregated uncertainty, with it progressively lowering (more visible for Figure 3b). Nevertheless, both models (BNN and DNN) present very similar fatigue load predictions and the trend is the same. However, after V_0, both models start to diverge. The DNN with conformalized predictions continues to converge into a stable value, not dissimilar to the value before V_0 and its uncertainty isn't altered. Conversely, after V_0, the BNN has a great increase in its predicted representative DEL, with this trend progressing over the rest of the interval. Importantly, the model uncertainty also increases for this period (see the zoomed region in Figure 3a).
This means that there has been a change in the inputs which the BNN i. considers it induces higher loads and ii. make the model less certain of its predictions. Both of these phenomena do not, however, occur for the DNN model with conformalized predictions. Indeed, if we compare the mean average estimated DEL for both models and the measured DEL (along with a P95-P5 envelope) we can see how, beginning in April, BNN predictions start to diverge from both measured DELs and DNN predictions (see Figure 4).

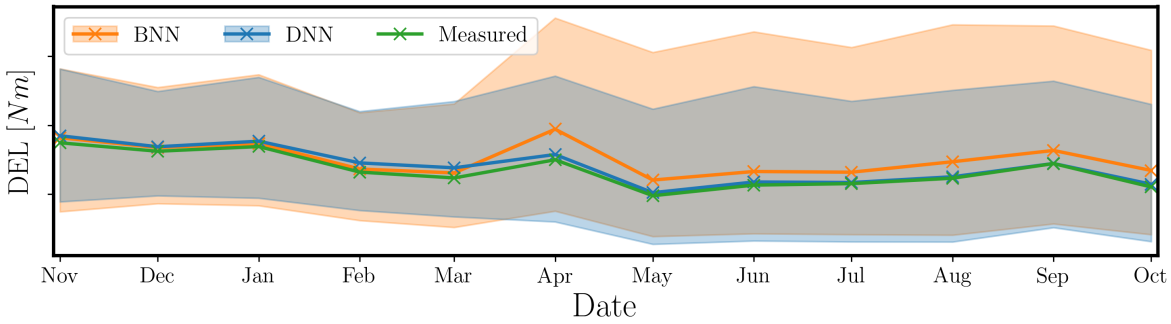


Figure 4: Monthly mean DEL as measured (green) and predicted by BNN (orange) and DNN (blue). 95% prediction spread in shaded colors.

Based on this figure, we can better understand Figure 3. Specifically, that the DNN's predictions are correct with relation to the as-measured response and, therefore, the stability of its uncertainty (of the conformalized predictors confidence interval) seems to reflect that, for this model, confidence in the predictions is correctly maintained. As for the BNN, the predictions have indeed deviated from measurements and over predicted fatigue loads. Upon further investigation, it has been ascertained that, after V_0, the turbine in question has encountered a new control strategy, which has altered the power curve. The DNN and BNN have not only different inputs (although overwhelmingly overlapping), but also have slightly different training periods. This has meant that the DNN was able to tackle appropriately these changes, but not the BNN. The problem surrounding the control-driven changes to the power curve at V_0 can be seen as paradigmatic for uncertainty quantification and propagation in offshore wind: we may be able to estimate uncertainty and even translate it into aggregates over a long time, but the offshore realm is a dynamic one, where changes to operation and environmental variables can be expected. It is not uncommon that, during the lifetime of a farm, different control strategies will be enacted, wind conditions will change (if not climate change, then if newer farms are constructed in the vicinity) or sea-state and -bed conditions will vary (*e.g.* scour).

4 Discussion

In this contribution we presented two methods (Bayesian neural networks, BNN, and conformal predictors) that estimate model uncertainty while predicting fatigue loads on offshore wind turbines' substructures. We saw how, even at a ten-minute level, coverage was different, which is dependent on the way both methods estimate uncertainty. Furthermore, beginning with ten-minute estimations of fatigue loads for one year of data from a real-world offshore wind turbine, we aggregated these into Damage Equivalent Loads (DEL), representative for the entire period. Similarly, to obtain a corresponding uncertainty, we employed Monte Carlo simulations. For conformal predictors, we assumed that each ten-minute damage follows a truncated normal distribution within the given interval.

In the results section, we showed how long-term DELs converge and uncertainty decreases with increasing data, so long as the models' underlying assumptions (*i.e.*, that the training data space will be sufficiently representative of future conditions) are held. If this fundamental assumption is broken, then i. DEL convergence cannot be guaranteed, and ii. uncertainty will increase. A key question here is the relationship between epistemic uncertainty, as related to the model, which can increase and have future predictions outside previous confidence bounds, and the uncertainty related to the fatigue lifetime. Naturally, operators are more concerned with having uncertainty bounds in their assets' lifetime, and research (and how it is communicated) should also reflect this. This contribution, as previously mentioned, is limited in its scope, only investigating model uncertainty. Therefore, the final link with lifetime uncertainty is still missing.

Another crucial point to raise relates with conformal prediction. As could be observed, this algorithm might produce wider uncertainty bounds than BNNs, but its flexible methodology can be highly advantageous. By being model-agnostic, it offers an enviable flexibility and can be adaptable to any form of preexisting model. This agnostic property has a further advantage when compared with BNNs or other models which inherently model uncertainty, has confidence bounds can be updated with incoming data without requiring model retraining. In this contribution, albeit implemented, its ramifications weren't fully explored, and future research should take this into account. Furthermore, in this preliminary study, it was assumed that each conformalized interval followed a truncated normal distribution. However, another flexibility of conformal predictors is that they require no underlying distribution assumption. Future approaches could also investigate distribution-free methods for aggregating uncertainties over time.

Data availability

Due to its proprietary nature and confidentiality concerns, supporting data of this research cannot be made publicly available.

Acknowledgements

The present research work is part of the WILLOW project, Funded by the European Union with GA No. 1011122184. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

REFERENCES

[1] Francisco de N Santos, Nymfa Noppe, Wout Weijtjens, and Christof Devriendt. Farm-wide interface fatigue loads estimation: A data-driven approach based on accelerometers. *Wind Energy*, 27(4):321–340, 2024.

[2] N. Requate, T. Meyer, and R. Hofmann. From wind conditions to operational strategy: optimal planning of wind turbine damage progression over its lifetime. *Wind Energy Science*, 8(11):1727–1753, 2023.

[3] Niklas Requate and Tobias Meyer. Optimal operational planning of wind turbine fatigue progression under stochastic wind uncertainty. In *European Safety and Reliability Conference 2023*, 2023.

[4] Jannie Sønderkær Nielsen, Lindsay Miller-Branovacki, and Rupp Carriveau. Probabilistic and risk-informed life extension assessment of wind turbine structural components. *Energies*, 14(4):821, 2021.

[5] Kristof Winkler, Maximillian Weil, C Sastre Jurado, Bruno Stuyts, Wout Weijtjens, and Christof Devriendt. Quantifying the effect of rock armour scour protection on eigenfrequencies of a monopile supported ohvs. In *Journal of Physics: Conference Series*, volume 2626, page 012039. IOP Publishing, 2023.

[6] Nandar Hlaing, Pablo G Morato, Francisco de Nolasco Santos, Wout Weijtjens, Christof Devriendt, and Philippe Rigo. Farm-wide virtual load monitoring for offshore wind structures via bayesian neural networks. *Structural Health Monitoring*, 23(3):1641–1663, 2024.

[7] Gregory Duthé, Francisco de Nolasco Santos, and Eleni Chatzi. Stochastic estimation of wake-induced loads in wind farms using conformalized graph neural networks. In *Proceedings of the 11th European Workshop on Structural Health Monitoring (EWSHM 2024)*, volume 29, page 621. NDT. net, 2024.

[8] Hippolyt Ritter, Aleksandar Botev, and David Barber. A scalable laplace approximation for neural networks. In *6th international conference on learning representations, ICLR 2018-conference track proceedings*, volume 6. International Conference on Representation Learning, 2018.

[9] Kazuki Osawa, Siddharth Swaroop, Mohammad Emtiyaz E Khan, Anirudh Jain, Runa Eschenhagen, Richard E Turner, and Rio Yokota. Practical deep learning with bayesian principles. *Advances in neural information processing systems*, 32, 2019.

[10] Florian Wenzel, Kevin Roth, Bastiaan S Veeling, Jakub Świątkowski, Linh Tran, Stephan Mandt, Jasper Snoek, Tim Salimans, Rodolphe Jenatton, and Sebastian Nowozin. How good is the bayes posterior in deep neural networks really? *arXiv preprint arXiv:2002.02405*, 2020.

[11] Thomas J DiCiccio and Bradley Efron. Bootstrap confidence intervals. *Statistical science*, 11(3):189–228, 1996.

[12] Andrey Malinin and Mark Gales. Uncertainty estimation in autoregressive structured prediction. *arXiv preprint arXiv:2002.07650*, 2020.

[13] Volodya Vovk, Alexander Gammerman, and Craig Saunders. Machine-learning applications of algorithmic randomness. 1999.

[14] Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. In *Machine learning: ECML 2002: 13th European conference on machine learning Helsinki, Finland, August 19–23, 2002 proceedings 13*, pages 345–356. Springer, 2002.

[15] Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*, volume 29. Springer, 2005.

[16] Yuan Shih Chow and Henry Teicher. *Probability theory: independence, interchangeability, martingales*. Springer Science & Business Media, 2003.

[17] Anastasios N Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *arXiv preprint arXiv:2107.07511*, 2021.

[18] Henrik Boström, Henrik Linusson, Tuve Löfström, and Ulf Johansson. Accelerating difficulty estimation for conformal regression forests. *Annals of Mathematics and Artificial Intelligence*, 81:125–144, 2017.

[19] Henrik Boström. crepes: a python package for generating conformal regressors and predictive systems. In Ulf Johansson, Henrik Boström, Khuong An Nguyen, Zhiyuan Luo, and Lars Carlsson, editors, *Proceedings of the Eleventh Symposium on Conformal and Probabilistic Prediction and Applications*, volume 179 of *Proceedings of Machine Learning Research*. PMLR, 2022.

[20] Frederik Hoppe, Claudio Mayrink Verdun, Hannah Laus, Felix Krahmer, and Holger Rauhut. Non-asymptotic uncertainty quantification in high-dimensional learning. *Advances in Neural Information Processing Systems*, 37:122524–122555, 2025.

[21] Francisco de N Santos, Pietro D'Antuono, Koen Robbelein, Nymfa Noppe, Wout Weijtjens, and Christof Devriendt. Long-term fatigue estimation on offshore wind turbines interface loads through loss function physics-guided learning of neural networks. *Renewable Energy*, 205:461–474, 2023.

[22] Robert L Harrison. Introduction to monte carlo simulation. In *AIP conference proceedings*, volume 1204, page 17, 2010.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

DATA DRIVEN METHODS FOR UNCERTAINTY PROPAGATION IN PREDICTING BRITTLE RESPONSES OF REINFORCED CONCRETE STRUCTURES

Afaq Ahmad[1] **and Vagelis Plevris**[2]

[1] Department of Built Environment, Oslo Metropolitan University
Pilestredet 35, Oslo 0166, Norway
e-mail: afaq.ahmad@oslomet.no

[2] Department of Civil and Environmental Engineering, College of Engineering, Qatar University
P.O. Box: 2713, Doha, Qatar
e-mail: vplevris@qu.edu.qa

Abstract

Predicting the brittle response of reinforced concrete (RC) members remains a significant challenge, especially for continuous or indeterminate beams where shear-critical behavior introduces considerable uncertainty. While professional tools like SAP2000 and research platforms such as ABAQUS offer trade-offs between usability and precision, they face notable limitations—ranging from underestimating failure modes to requiring substantial computational resources. This study introduces a hybrid Artificial Neural Network–Finite Element Analysis (ANN–FEA) methodology as a computationally efficient, data-driven alternative for the analysis and design of shear-critical RC structures. The approach integrates ANN-based failure prediction models, trained on experimental data, with segment-wise nonlinear pushover analysis. A curated dataset of 315 RC beams was used to develop and calibrate eight ANN configurations, among which the BWS-7 model achieved the highest predictive accuracy. The ANN–FEA method was validated against experimental results and benchmarked through a case study involving indeterminate RC beams subjected to axial and transverse loading. The results demonstrate that the ANN–FEA model closely matches experimental outcomes, achieves accuracy comparable to ABAQUS, and outperforms SAP2000 in detecting brittle failure. This work advances uncertainty quantification in RC analysis and offers a practical tool for structural assessment under complex loading conditions.

Keywords: Reinforced Concrete Beams, Multi-span, Design Tools, Artificial Neural Network.

1 INTRODUCTION

Since the mid-20th century, the rapid expansion of concrete construction has significantly transformed the role and technical responsibilities of structural engineers [1-5]. These professionals are tasked with ensuring the structural safety, serviceability, and cost-effectiveness of reinforced concrete (RC) buildings. In recent years, research has increasingly focused on developing predictive models that capture the brittle behavior of RC frame elements. Such analytical tools have become essential for a variety of applications, including: (i) designing new RC structures in compliance with modern codes; (ii) reassessing existing structures based on updated regulations; (iii) evaluating structural performance; (iv) identifying design inconsistencies between the planning and construction phases; and (v) supporting forensic investigations of structural failures [1-5].

In professional engineering practice, various national and international design codes are commonly used to assess the structural performance of complex RC members, particularly columns subjected to axial loads at the ultimate limit state (ULS). Prominent among these are the American ACI 318 [6], the European Eurocode 2 (EC2) [7], the Japanese JSCE [8], New Zealand's NZS 3101.1:2006 [9], the Korean KBSC [10], and the Canadian CSA A23.3 [11]. These standards typically rely on analytical models grounded in truss-based representations and strut-and-tie mechanisms, often supplemented by cross-sectional analysis methods to characterize structural behavior at ULS. An alternative to these conventional approaches is the Compressive Force Path (CFP) method, which provides a distinct framework for evaluating RC member behavior [12].

To meet these evolving challenges, the field has gradually integrated a variety of advanced analytical methods over the past decades [13-15]. Today, structural engineers depend on sophisticated computational tools both to design new RC structures in compliance with modern regulatory frameworks and to reassess existing structures according to current standards [6-11]. This transition has marked the emergence of a new era in high-fidelity modeling and simulation—enhancing the ability to predict brittle failure mechanisms but also requiring substantial computational resources and specialized expertise [16-20].

However, nonlinear finite element analysis (NLFEA) tools such as ABAQUS [13] and ADINA [15] are primarily used in research settings due to their complexity and computational demands. In contrast, software commonly used in the design industry—such as SAP2000 and ETABS [14]—is typically employed for practical structural analysis and design tasks. While these tools are generally reliable for predicting flexural (ductile) failure modes, they often fall short in accurately capturing shear (brittle) failure mechanisms [1-5]. Given that shear failures are usually sudden and catastrophic, occurring with little to no prior indication, it is essential for structural analysis software to provide accurate predictions of such failure modes. Reliable forecasting is critical to developing safe and cost-effective design strategies that preserve the structural integrity and resilience of RC frame systems. Conversely, inaccurate predictions can result in unsafe designs by overestimating the load-bearing capacity and ductility of structural elements.

Recent advancements in artificial intelligence have significantly influenced civil engineering practice and education, highlighting the transformative potential of machine learning and large language models in structural analysis and design workflows [21-24]. In recent years, soft computing (SC) techniques—particularly Artificial Neural Networks (ANNs)—have emerged as a promising alternative to traditional finite element (FE) methods for structural analysis [25-30]. This advancement has led to the development of hybrid ANN–FEA models, which integrate neural networks with finite element analysis to enhance prediction accuracy

and computational efficiency [28, 31-33]. The proposed ANN–FEA approach consists of three main steps, as illustrated in Figure 1:

- **Step 1**: ANN models are trained using experimental datasets. These models function as failure criteria specifically designed to predict shear failure in individual RC members, such as beams and columns, as established in the authors' prior studies [27, 32, 34-39].
- **Step 2**: The structural system is systematically divided into discrete segments based on the locations of peak bending moments and points of contraflexure, or between successive contraflexure points. This segmentation emulates the behavior of supported beam elements and enables localized structural analysis [31, 32].
- **Step 3**: During nonlinear pushover analysis, the trained ANN models are employed as failure indicators. The analysis terminates once the applied load exceeds the predicted shear capacity of any segment, ensuring accurate detection of potential failure.

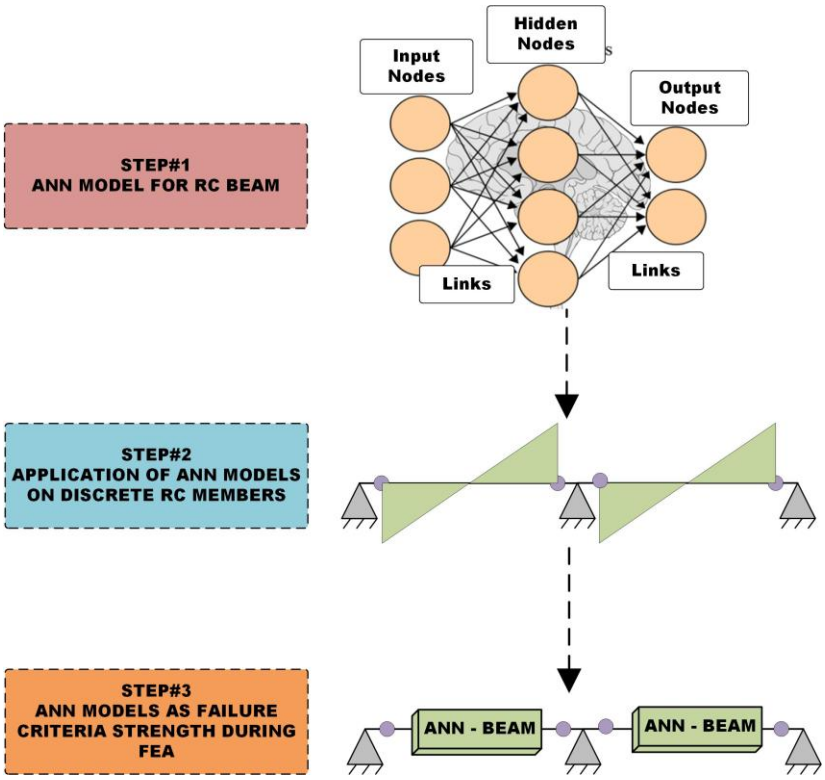


Figure 1: Overview of the proposed ANN–FEA hybrid methodology for reinforced concrete (RC) continuous beams [31, 32].

The proposed ANN–FEA structural analysis methodology demonstrates strong effectiveness in both the design of new RC structures and the assessment of existing ones. Moreover, it enhances the predictive capabilities of current structural analysis software used in professional practice [31, 32]. This study emphasizes the key strengths of the ANN–FEA approach, particularly its ability to simulate the behavior of indeterminate RC structural systems, including beams that fail in flexure or shear. The predictions generated by the ANN–FEA method are compared with those from ABAQUS [13], a high-fidelity finite element analysis tool widely used in research, and SAP2000 [14], a popular commercial software used in engineering practice. These predictions are further validated against experimental data. Following this validation, a parametric study is carried out to evaluate the method's performance. The results

indicate that the ANN–FEA approach yields brittle failure predictions that closely align with experimental and ABAQUS outcomes. In contrast, SAP2000 fails to capture brittle shear failure accurately, as it does not terminate the analysis when such failure occurs due to inherent software limitations.

2 EXPERIMENTAL DATABASE OF RC BEAMS WITH STIRRUPS (BWS)

A comprehensive database of RC beam specimens with stirrups (BWS) was compiled to support the current study. The dataset consists of 315 beams tested under either three-point or four-point loading configurations, as illustrated in Figure 2. This curated database systematically captures key design parameters that influence the load-carrying capacity and failure modes of RC beams at the ULS. It includes detailed information on material properties, geometric dimensions, reinforcement details, and loading conditions—enabling in-depth analysis of structural behavior under extreme loading scenarios.

The database serves as a critical foundation for predictive modeling and performance evaluation, facilitating the development of safer and more efficient structural designs. Key features of the dataset include: Material properties; Geometric dimensions; Reinforcement configurations; Loading conditions; Observed failure modes; Measured load-carrying capacity. Table 1 presents statistical summaries for these parameters. Among the 315 specimens, 204 (65%) failed in shear, while the remaining 111 (35%) failed in flexure.

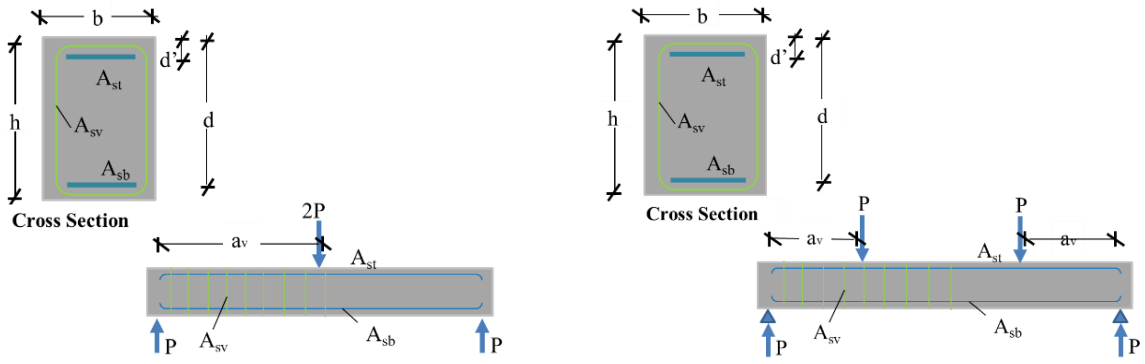


Figure 2: Experimental setup for RC Beams with Stirrups BWS: (a) three-point loading configuration, (b) four-point loading configuration.

ID	Quantity	Unit	Range (Min–Max)	Avg.	COV	St. Dev
1	b	mm	100 – 510	207.27	0.33	67.64
2	d	mm	113 – 975	342.87	0.46	157.03
3	a_v/d		1 – 7	3.43	0.39	1.35
4	ρ_l	(%)	0.18 – 5.60	2.35	0.45	1.05
5	f_{yl}	MPa	250 – 880	411.33	0.19	77.88
6	f_c	MPa	12 – 125	44.51	0.5	22.41
7	ρ_w	(%)	0.08 – 2.25	0.56	0.86	0.48
8	f_{yw}	MPa	225 – 875	399.4	0.28	112.62
9	M_f	kN·mm	2648 – 1738423	283485	1.34	378729
10	V_u	kN	4 – 760	185.54	0.72	134.25

Table 1: Statistical information for BWS.

2.1 Correlation Coefficient of the Parameters

Pearson's correlation coefficient (R), computed using Eq. (1) [40], is used to quantify the strength and direction of the linear relationship between variables in the dataset. Figure 3 illustrates the correlation values between the shear capacity (V_u) and each input parameter included in the BWS database.

$$r = \frac{n\sum xy - \left(\sum x\right)\left(\sum y\right)}{\sqrt{n\sum x^2 - \left(\sum x\right)^2}\sqrt{n\sum y^2 - \left(\sum y\right)^2}} \tag{1}$$

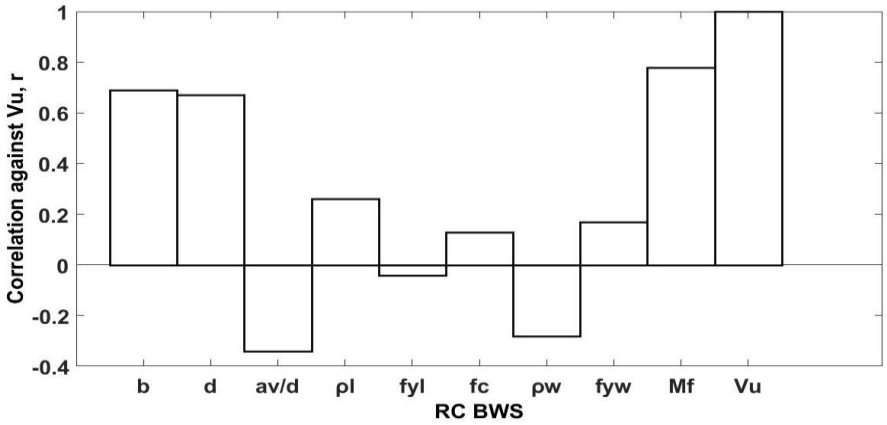


Figure 3: Pearson correlation coefficient r between the shear capacity V_U and each of the parameters including itself, for the RC beams with stirrups (BWS).

3 ARTIFICIAL NEURAL NETWORK MODELING FOR SHEAR CAPACITY PREDICTION

Artificial Neural Networks are widely used in machine learning and data mining for their ability to learn patterns and predict outputs from input data. An ANN consists of a system of interconnected nodes, or neurons, as shown in Figure 4, and is inspired by the structure and functioning of biological neural networks in the human brain [27, 32, 34, 35]. Unlike traditional computational models, ANNs do not require predefined assumptions about the underlying relationships between variables. This flexibility allows them to effectively capture and model complex, nonlinear interactions within real-world data. During training, ANNs can also uncover hidden patterns and trends, making them especially useful in large datasets due to their parallel processing capabilities.

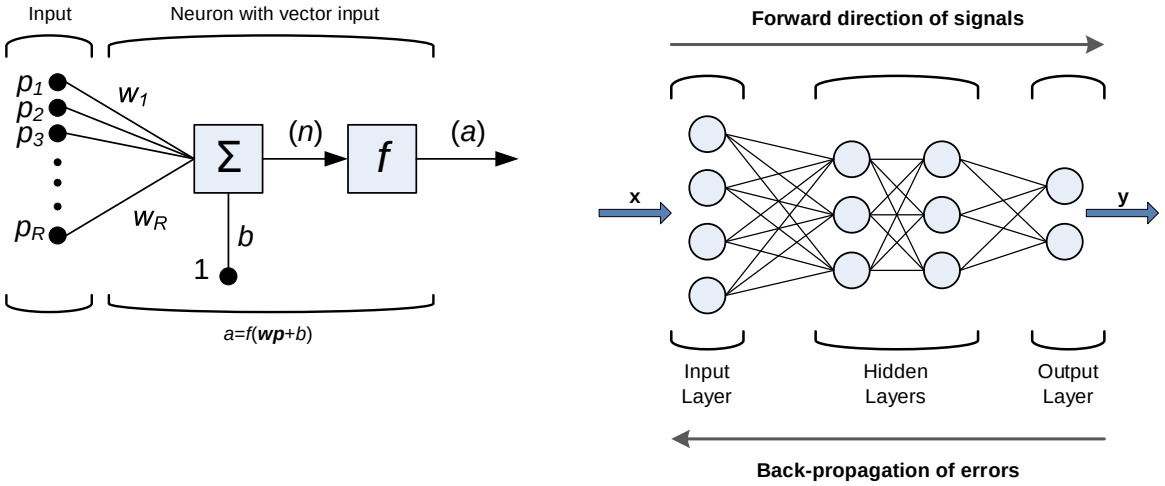
An ANN typically consists of three main components: an input layer, one or more hidden layers, and an output layer. The core mathematical operation of a single neuron within the network is described in Eq. (2), where the output is determined by the weighted sum of inputs and a bias term, passed through an activation function. Eq. (2) describes the output of a single artificial neuron in the network, not the response of the entire ANN model. The predictive capability of the network depends on three key parameters: the input values, the connection weights, and the bias term.

$$O = f\left(\sum x_i w_i + b\right) \tag{2}$$

During training, the network uses the backpropagation algorithm in conjunction with the delta rule to iteratively adjust the weights, minimizing the difference between the predicted outputs and the actual target values from the training data. As illustrated in Figure 4, this ad-

justment process flows backward through the network—from the output layer to the input layer. The optimization procedure, governed by the minimization of the mean squared error (*MSE*) as expressed in Eq. (3), continues until the error converges below a specified tolerance threshold, thereby enhancing the model's predictive accuracy [27, 32, 34, 35].

$$E(w) = \frac{1}{2}\sum_{i}(T-O)^2 \tag{3}$$



(a) (b)

Figure 4. (a) Structure of a single artificial neuron with multiple weighted inputs, bias, and activation function. (b) Feedforward ANN architecture showing input, hidden, and output layers with backpropagation of errors.

In this study, input variables were normalized using Eq. (4), which scales values to a range of 0.2 to 0.8, deviating from the conventional 0 to 1 range commonly found in the literature [27, 32, 34, 35]. This modified normalization approach was adopted to prevent saturation in activation functions—particularly in sigmoid or tanh-based networks—by avoiding input values near the function's boundaries. Maintaining input values within this narrower range helps preserve gradient activity during training, thereby improving numerical stability and accelerating convergence.

$$X = \left(\frac{0.6}{\Delta}\right)x + \left(0.8 - \left(\frac{0.6}{\Delta}\right)x_{\max}\right) \tag{4}$$

3.1 Calibration of Proposed ANN Model

In this study, an ANN was employed to estimate the strength characteristics of reinforced concrete. The modeling process began with the division of the dataset: 60% of the data was used for training, while the remaining 40% was equally split into validation and testing subsets (20% each).

The ANN was constructed using the training data, and its performance was evaluated by comparing predicted results with experimental values. This evaluation used statistical metrics such as mean absolute error (*MAE*), root mean square error (*RMSE*), and the Pearson's correlation coefficient (*R*) to assess accuracy and reliability [40]. The modeling workflow is illustrated in Figure 4 (a) and (b).

A multi-layer feedforward backpropagation ANN architecture was adopted for this study. The network was trained using the Levenberg–Marquardt optimization algorithm and em-

ployed a logistic sigmoid activation function in the hidden layers. Once the input and output variables were transformed into numerical arrays (datasets), the supervised learning process was initiated.

To test the model's generalization ability, new material combinations and proportions were introduced during the testing phase. The model's ability to accurately predict outcomes for previously unseen data highlights its effectiveness in capturing the complex physical behavior of RC elements. This approach ensures that both input and output responses are appropriately learned and validated within the training domain.

3.2 ANN model development

To evaluate the predictive performance of different input configurations and network structures, eight distinct ANN models were developed and tested for estimating the load-carrying capacity of RC beams with stirrups (BWS). Each model, labeled BWS-1 through BWS-8, was designed using varying combinations of input parameters and network architectures, as detailed in Table 2. The goal of this approach was to identify the most effective model configuration in terms of accuracy, generalization, and alignment with physical behavior. These models incorporated a range of geometric, material, and derived parameters, while the architectures varied in terms of the number of input neurons, hidden layers, and neurons per layer. By systematically comparing the results of these eight configurations, the study aims to determine the optimal ANN structure for reliable prediction of shear strength in RC beam elements.

ANN label	Parameters considered	No. of parameters	Network architecture
BWS-1	b, d, a_v/d, ρ_l, f_{yl}, f_c, ρ_w, f_{yw}	8	8-16-16-1
BWS-2	b, d, a_v/d, M_f, f_c, ρ_w, f_{yw}	7	7-14-14-1
BWS-3	b/d, a_v/d, M_f/f_cbd^2, ρ_w, f_{yw}	5	5-10-10-1
BWS-4	b/d, a_v/d, ρ_w/ρ_l, f_c/f_{yw}	4	4-8-8-1
BWS-5	d, a_v/d, M_f/bd^2, f_c, $\rho_w{\cdot}f_{yw}$	5	5-10-10-1
BWS-6	d, b/d, a_v/d, M_f/f_cbd^2, ρ_w, f_{yw}	6	6-12-12-1
BWS-7	d, b/d, a_v/d, M_f/f_cbd^2, f_c, ρ_w, f_{yw}	7	7-14-14-1
BWS-8	d, b/d, a_v/d, M_f/f_cbd^2, f_c, $\rho_w{\cdot}f_{yw}$	6	6-12-12-1

Table 2: Summary of the eight developed ANN models (BWS-1 to BWS-8), showing the input parameters considered, the number of parameters used, and the corresponding network architecture for each model.

The performance results of the eight ANN models (BWS-1 to BWS-8) are illustrated in Figure 5 and Figure 6. Figure 5 presents scatter plots showing the correlation between predicted and experimental values of V_u for each model, providing a visual assessment of prediction accuracy. Figure 6 quantitatively evaluates each model using three key statistical metrics: mean squared error (*MSE*), mean absolute error (*MAE*), and Pearson's correlation coefficient (R).

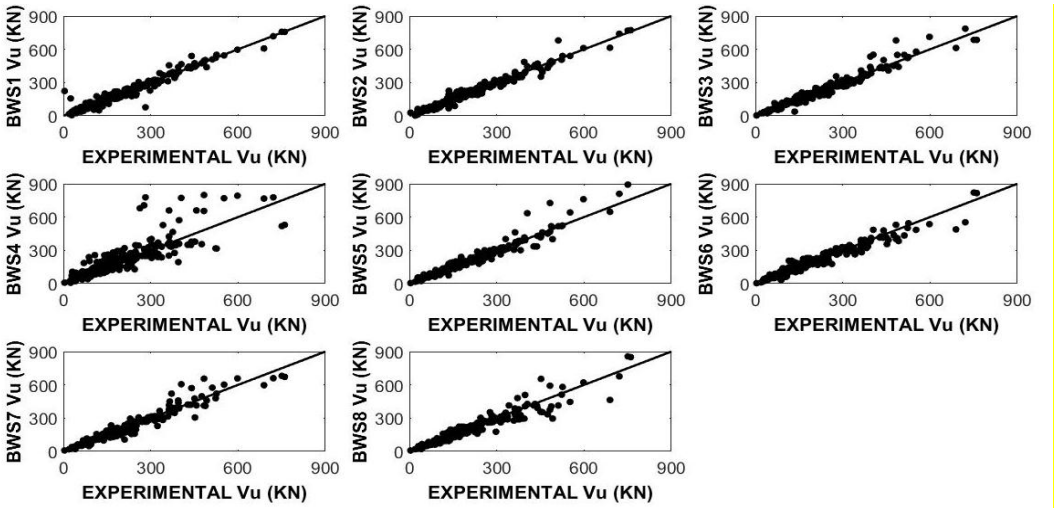


Figure 5: Comparison of predicted shear capacity (V_u) vs. experimental values for the eight ANN models (BWS-1 to BWS-8).

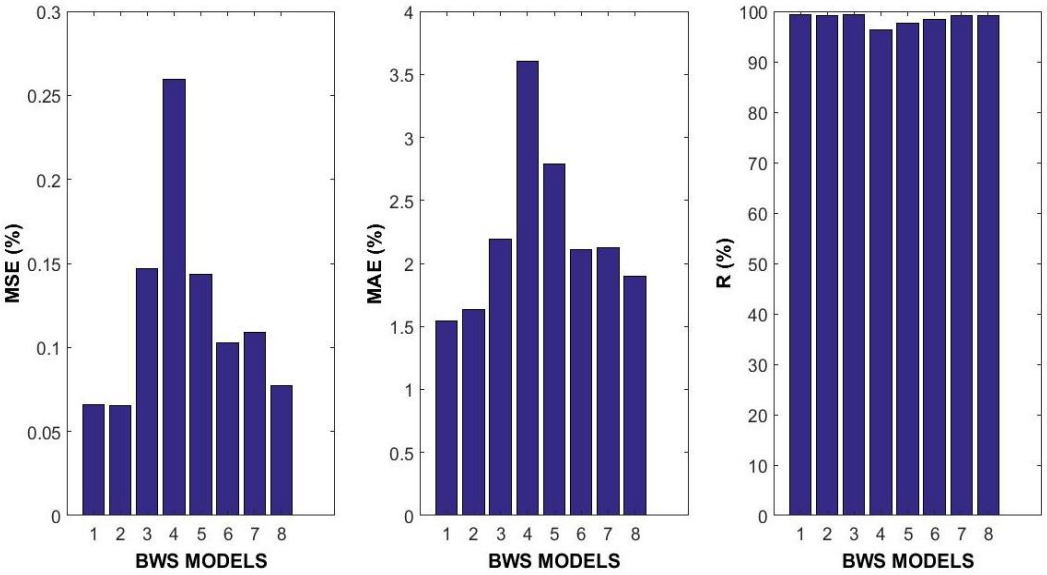


Figure 6: Statistical performance metrics for the eight ANN models (BWS-1 to BWS-8).

These metrics offer insight into each model's accuracy and generalization capability, following the evaluation framework outlined in previous studies [27, 32, 34, 35, 38, 39]. Among the tested models, BWS-7 achieves the best overall performance, exhibiting the lowest error values and the highest correlation coefficient. This model utilizes a well-balanced set of seven input parameters: beam depth (d), width-to-depth ratio (b/d), shear span-to-depth ratio (a_v/d), normalized moment capacity parameter ($M_f/f_c bd^2$), concrete compressive strength (f_c), transverse reinforcement ratio (ρ_w), and yield strength of web reinforcement (f_{yw}). The BWS-7 configuration effectively captures the structural behavior and delivers the most reliable predictions among the models tested.

4 COMPARATIVE STUDY OF THE ANN MODEL WITH DESIGN CODES

This section presents a comparative evaluation of shear capacity predictions for reinforced concrete beams with stirrups (RC BWS) using three distinct approaches: conventional design codes, the CFP method, and the data-driven ANN model—specifically, BWS-7. The predicted shear capacities from each method are benchmarked against experimental results to assess their accuracy and reliability.

As illustrated in Figure 7, the ANN model (BWS-7) exhibits the strongest correlation with experimental data, outperforming the traditional design codes and other alternative methods. This highlights the potential of ANN-based approaches, which rely on heuristic, data-driven learning, to capture complex structural behavior more effectively than conventional, mechanics-based formulations.

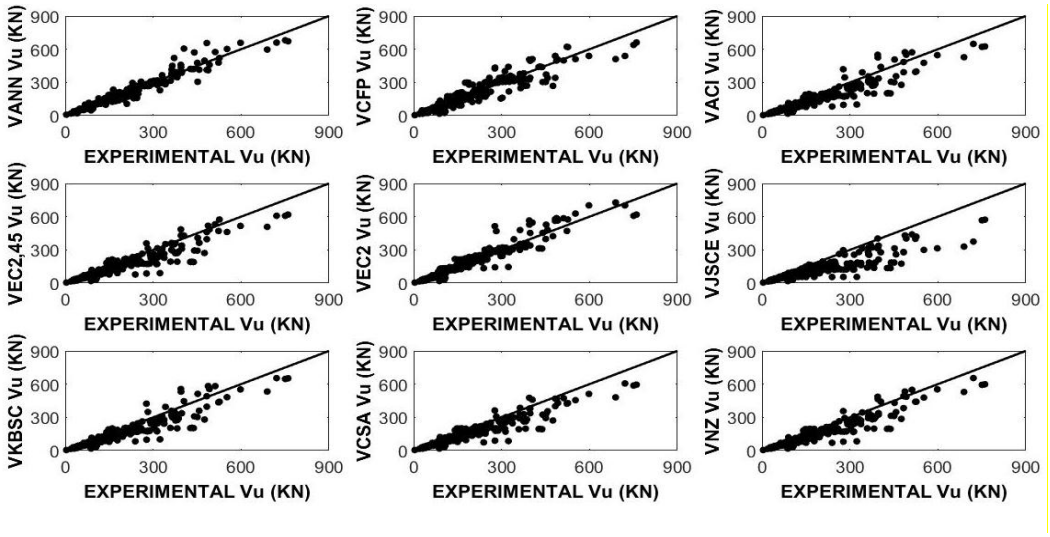


Figure 7: Comparison of predicted vs. experimental shear capacity (V_u) for RC beams with stirrups using the ANN model (BWS-7), the CFP method, and various international design codes.

While methods such as EC2 [7], CFP [12], and other design code-based approaches [8-11] provide reasonable predictions, their accuracy varies across the dataset. Notably, the CFP method, despite relying on non-traditional physical assumptions, delivers predictions that closely align with experimental results. However, the ANN model consistently shows superior agreement with experimental results across the full range of tested values, demonstrating its robustness and adaptability in modeling RC beam behavior beyond the limitations of code-based approximations.

Figure 8 illustrates the distribution of prediction accuracy for all methods, represented as the ratio of experimental to predicted shear capacity (V_{EXP}/V_{PRE}). The results reveal that traditional RC design codes—such as ACI-318 [6], EC2 [7], JSCE [8], NZ [9], KBSC [10], and CSA [11], as well as the CFP method [12], tend to underestimate shear strength. This tendency is evident from the larger number of predictions falling outside the acceptable range of 0.76–1.25.

In comparison, the ANN model (BWS-7) demonstrates superior predictive consistency, with the majority of its results concentrated within the desired accuracy interval. Conversely, the CFP method and most code-based approaches show significantly higher frequencies in the underestimation zones (1.26–2.00 and 2.01–12.00), reinforcing their conservative nature. These findings underscore the variability in prediction performance across methods and high-

light the ANN's advantage in delivering more balanced and accurate shear strength predictions for RC beams with stirrups.

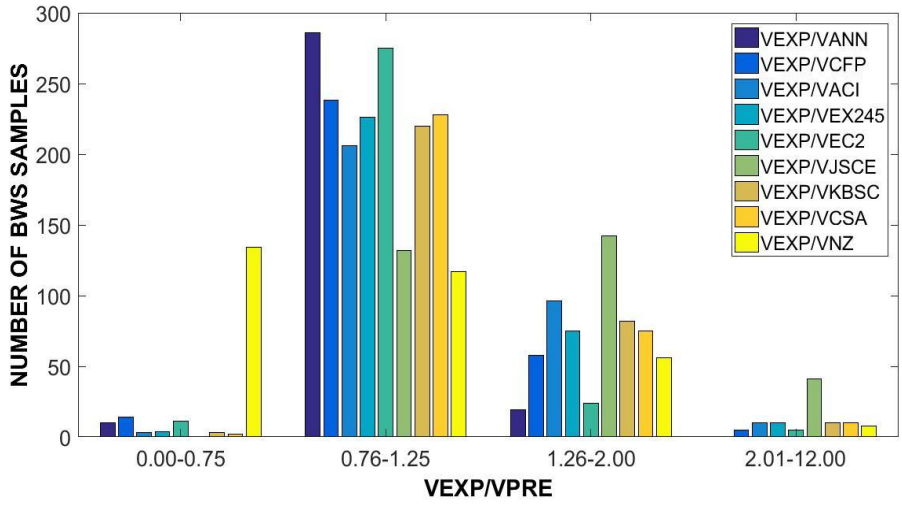


Figure 8: Distribution of prediction-to-experiment ratios (V_{Exp}/V_{PRE}) for RC beams with stirrups (BWS) across different methods.

5 CASE STUDY: ANALYSIS OF INDETERMINATE RC BEAM SPECIMENS

In this study, the proposed ANN–FEA methodology is applied to analyze indeterminate reinforced concrete beams with stirrups (RC-BWS) and conventional cross-sections [32]. The case involves beams subjected to a combination of axial compression (N) and a mid-span transverse load (P) applied to the longer span. The loading and support configuration are shown in Figure 9, which represents a continuous beam tested under realistic loading conditions as reported in [12].

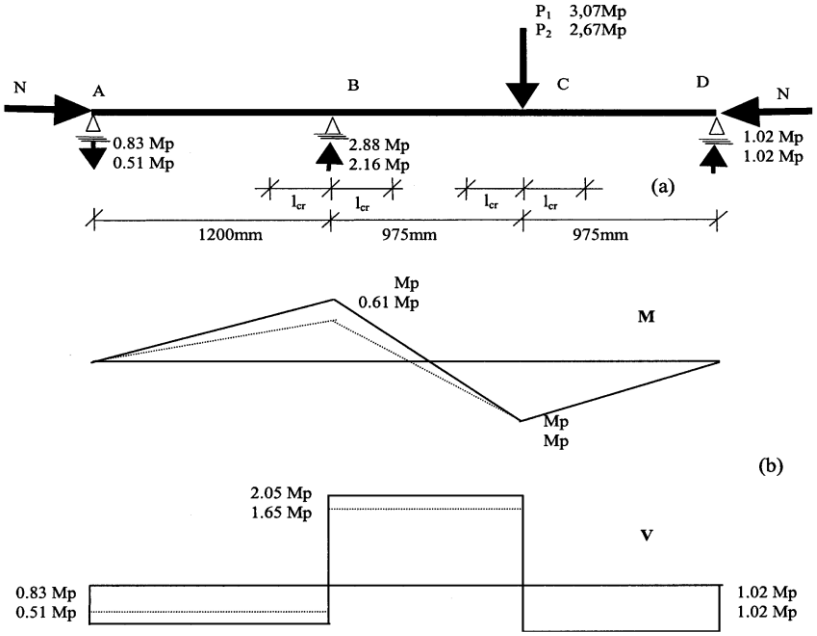


Figure 9: Experimental setup, loading scheme, and internal force diagrams for the indeterminate RC beam specimens [12].

5.1 Specimens with CN=0 and CN=180 kN

To evaluate the structural behavior of statically indeterminate RC beams, three analytical approaches were employed: the proposed ANN-FEA model [32], ABAQUS [13], and SAP2000 [14]. The numerical results obtained from these methods were systematically compared with corresponding experimental data. The experimental study [12] focused on two beam specimens subjected to different axial compression levels: CN = 0 kN and CN = 180 kN. The geometric and reinforcement details of these specimens are presented in Figure 10, which provides a complete overview of the design configuration and loading conditions used in the tests.

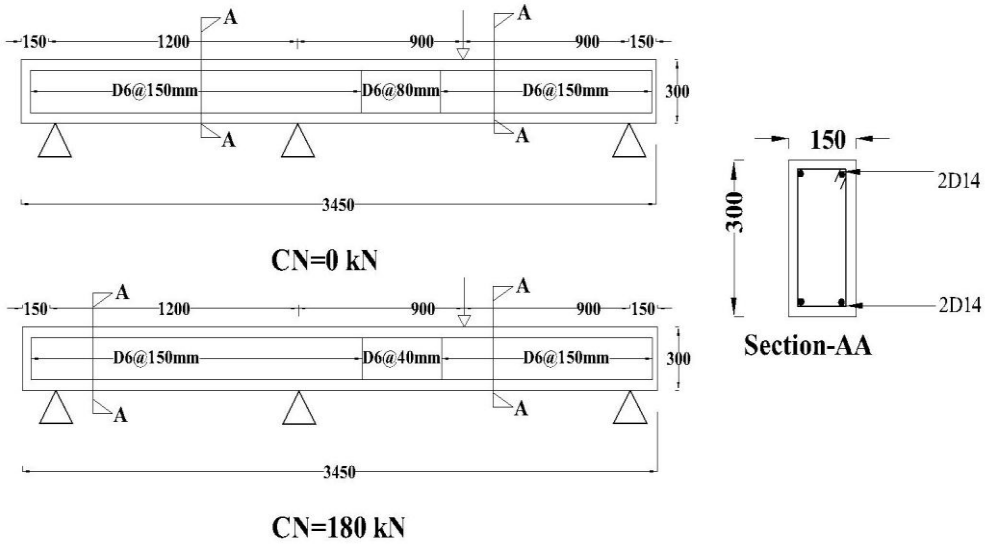


Figure 10: Geometric and reinforcement details of the indeterminate RC beam specimens subjected to axial compression levels of CN = 0 kN and CN = 180 kN [12].

Figure 11 presents a comparative assessment of experimental observations and numerical predictions for indeterminate RC beams subjected to axial loads of (a) CN=0, and (b) CN=180 kN. The results are obtained from three analytical approaches: the ANN–FEA model [32], ABAQUS [13], and SAP2000 [14]. The first row (Experiment) displays the observed cracking patterns at the ULS. The second row (SAP2000) highlights the formation and location of plastic hinges, indicating flexural failure zones as identified through SAP2000 analysis. The third row (ABAQUS) illustrates the distribution and intensity of concrete damage predicted by finite element simulations in ABAQUS as the beam approaches failure. The fourth row (ANN-FE) shows the failure mode predicted by the ANN–FEA methodology.

This visual comparison demonstrates the capacity of the ANN–FEA model to replicate experimental outcomes with high fidelity, while also highlighting the distinct representation of damage and failure in each numerical approach.

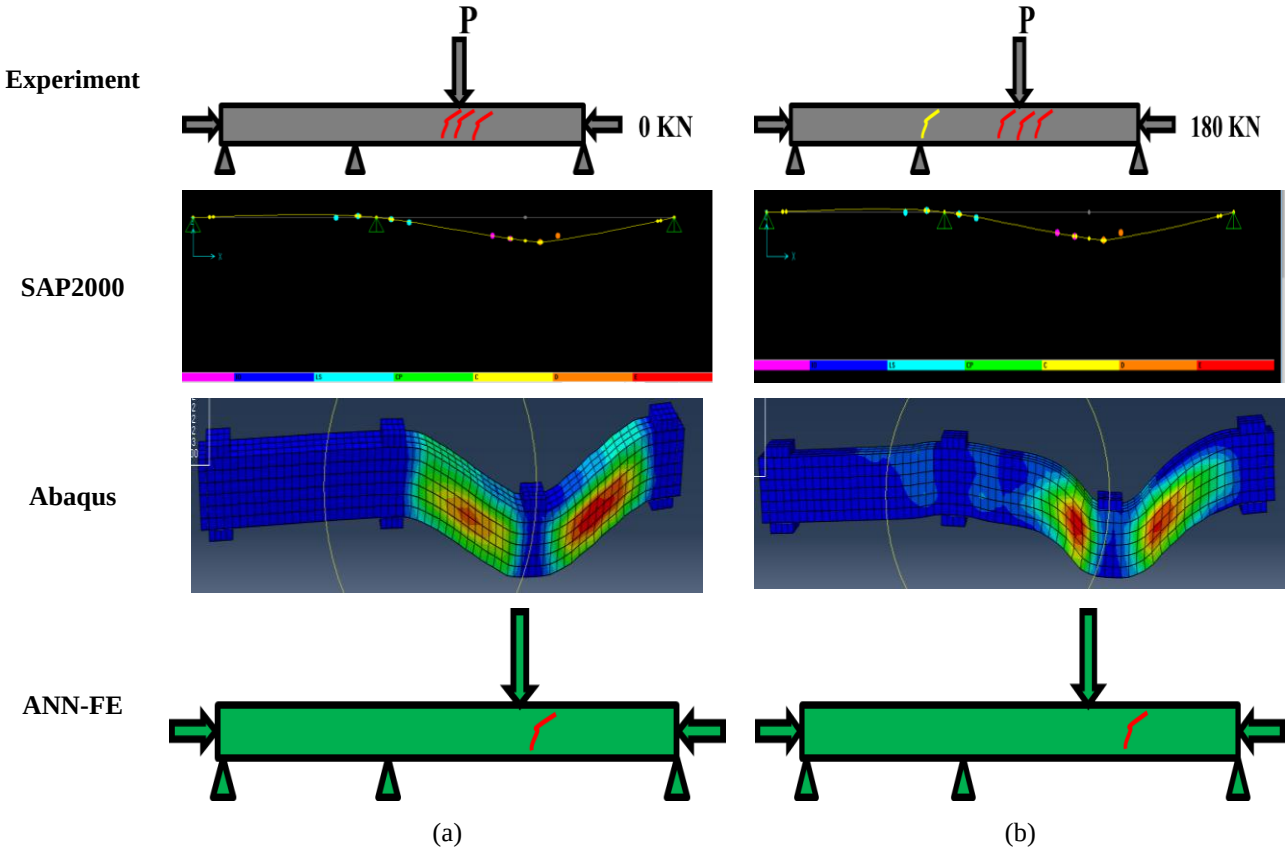
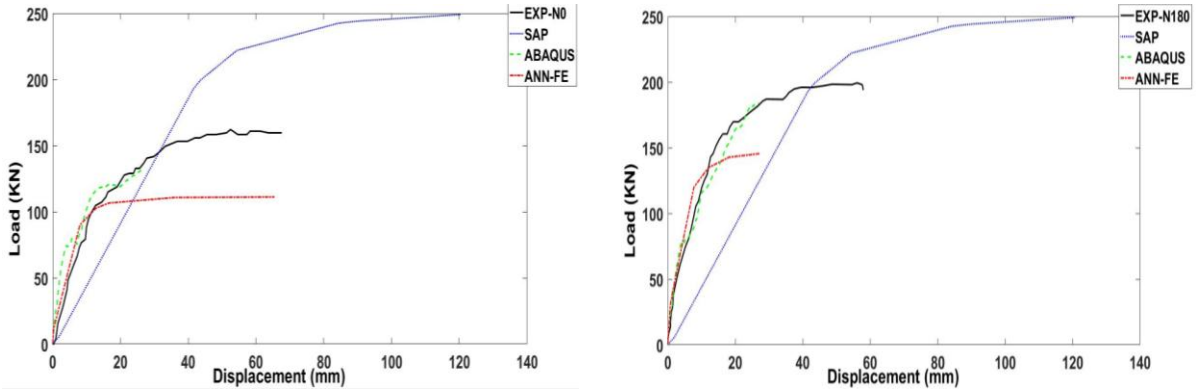


Figure 11: Comparison of experimental and numerical results for indeterminate RC beam specimens under axial loads: (a) CN = 0 kN; (b) CN = 180 kN [12].

Figure 12 presents a comparative analysis of the load–deflection behavior for RC beam specimens under axial loads of (a) CN=0, and (b) CN=180 kN. The curves capture the full loading response up to the ULS, as observed experimentally and predicted by SAP2000, ABAQUS, and the ANN–FEA model [32].

The results show that the ANN–FEA model accurately replicates the experimentally observed failure behavior, including crack initiation and progression, especially in the post-peak region. ABAQUS also exhibits strong alignment with the experimental curves, capturing both stiffness and ultimate capacity effectively. In contrast, SAP2000 tends to overestimate the load-carrying capacity, reflecting its limitations in modeling nonlinear behavior and brittle failure modes.

These findings underscore the enhanced predictive accuracy of both the ABAQUS and ANN–FEA approaches, particularly in capturing complex structural responses under combined axial and transverse loading. Their reliable simulation of load–deflection behavior reinforces their suitability for advanced structural analysis and design of RC elements.



(a) (b)

Figure 12: Comparison of load–deflection curves for indeterminate RC beam specimens under axial loads at ULS: (a) CN = 0 kN; (b) CN = 180 kN.

6 DISCUSSION AND CONCLUSIONS

The proposed ANN–FEA methodology [32] has been specifically developed for the non-linear static analysis of RC frame structures under pushover loading. Based on the results of the four investigated cases, the method has demonstrated strong capability in accurately predicting the structural response of shear-critical RC elements. Its predictive performance is particularly notable in complex scenarios, such as indeterminate RC beams, where conventional tools often fall short.

The following key conclusions can be drawn from the study:

- **Importance of Shear Effects**: Accurately evaluating the strength and ductility of RC frames requires explicit consideration of shear effects. Neglecting these effects can lead to unsafe or overly optimistic assessments—particularly relevant given the continued use of shear-critical frames in real-world applications.
- **Limitations of Common Professional Tools**: Widely used structural analysis programs like SAP2000 [14] tend to oversimplify or ignore shear effects. As a result, they often overestimate both the strength and ductility of RC structures in shear-critical conditions, potentially compromising the accuracy and safety of structural designs.
- **Strengths and Constraints of Research-Grade Tools**: High-fidelity platforms such as ABAQUS [13] provide more accurate simulations that align closely with experimental data. However, their use is hindered by significant computational demands and longer analysis times, which can limit their practicality for routine engineering tasks.
- **Advantages of the ANN–FEA Model**: The proposed ANN–FEA approach offers an efficient, accessible, and accurate alternative for nonlinear analysis. It successfully replicates key aspects of experimental behavior—such as crack initiation, load-bearing capacity, and post-peak response—making it a powerful tool for assessing shear-critical RC frames with high precision.

In conclusion, the ANN–FEA methodology represents a meaningful advancement in the non-linear analysis of RC structures. By addressing the limitations of existing professional and research tools, it provides a practical and reliable framework for structural assessment. Its ability to simulate complex behaviors under combined loading with minimal computational effort makes it a promising solution for safer, more efficient, and realistic RC design and evaluation practices.

Funding: This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska–Curie grant agreement No. 101153307 (https://cordis.europa.eu/project/id/101153307).

Conflict of Interest: The authors declare no conflict of interest.

REFERENCES

[1] Theodossopoulos, D. and B. Sinha, *A review of analytical methods in the current design processes and assessment of performance of masonry structures.* Construction and Building Materials, 2013. **41**: p. 990-1001 DOI: https://doi.org/10.1016/j.conbuildmat.2012.07.095.

[2] Krawinkler, H. and G.D.P.K. Seneviratna, *Pros and cons of a pushover analysis of seismic performance evaluation.* Engineering Structures, 1998. **20**(4): p. 452-464 DOI: https://doi.org/10.1016/S0141-0296(97)00092-8.

[3] Sung, Y.C., T.K. Lin, C.C. Hsiao, and M.C. Lai, *Pushover analysis of reinforced concrete frames considering shear failure at beam-column joints.* Earthquake Engineering and Engineering Vibration, 2013. **12**(3): p. 373-383 DOI: https://doi.org/10.1007/s11803-013-0179-8.

[4] Guner, S. and F.J. Vecchio, *Pushover Analysis of Shear-Critical Frames: Verification and Application.* ACI Structural Journal, 2010. **107**(1) DOI: https://doi.org/10.14359/51663390.

[5] Sharma, A., G.R. Reddy, K.K. Vaze, and R. Eligehausen, *Pushover experiment and analysis of a full scale non-seismically detailed RC structure.* Engineering Structures, 2013. **46**: p. 218-233 DOI: https://doi.org/10.1016/j.engstruct.2012.08.006.

[6] ACI Committee 318, *ACI 318-19: Building Code Requirements for Structural Concrete and Commentary*. 2019.

[7] CEN, *EN 1992-1: Eurocode 2, Design of concrete structures, Part 1*. 1992, European Committee for Standardization (CEN).

[8] Japan Society of Civil Engineers (JSCE), *Standard Specifications for Concrete Structures*. 2007.

[9] Standards New Zealand, *NZS 3101.1:2006 (inc A1, A2, A3) Concrete structures standard - Part 1: The design of concrete structures*. 2017.

[10] Architectural Institute of Korea, *Korean Building Code-Structural (KBCS)*. 2005.

[11] Canadian Standards Association, *CSA A23.3:24. Design of concrete structures*. 2024.

[12] Kotsovos, M.D., *Compressive Force-Path Method*. 2014: Springer Cham, ISBN: 978-3-319-34542-0.DOI: 10.1007/978-3-319-00488-4.

[13] Börgesson, L., *ABAQUS*, in *Developments in Geotechnical Engineering*, O. Stephansson, L. Jing, and C.-F. Tsang, Editors. 1996, Elsevier. p. 565-570. DOI https://doi.org/10.1016/S0165-1250(96)80047-2.

[14] Computers & Structures Inc., *CSI Analysis Reference Manual. For SAP2000, ETABS, SAFE and CSiBridge, version 19*. 2016.

[15] Bathe, K.-J. and G. Larsson, *The Use of ADINA in Engineering Practice*, in *Finite Element Systems: A Handbook*, C.A. Brebbia, Editor. 1982, Springer Berlin Heidelberg: Berlin, Heidelberg. p. 3-18. DOI https://doi.org/10.1007/978-3-662-07229-5_1.

[16] Zakaria, M., T. Ueda, Z. Wu, and L. Meng, *Experimental Investigation on Shear Cracking Behavior in Reinforced Concrete Beams with Shear Reinforcement.* Journal of

Advanced Concrete Technology, 2009. **7**(1): p. 79-96 DOI: https://doi.org/10.3151/jact.7.79.

[17] Stratford, T. and C. Burgoyne, *Shear Analysis of Concrete with Brittle Reinforcement.* Journal of Composites for Construction, 2003. **7**(4): p. 323-330 DOI: https://doi.org/10.1061/(ASCE)1090-0268(2003)7:4(323).

[18] Russo, G., D. Mitri, and M. Pauletta, *Shear strength design formula for RC beams with stirrups.* Engineering Structures, 2013. **51**: p. 226-235 DOI: https://doi.org/10.1016/j.engstruct.2013.01.024.

[19] Campione, G., A. Monaco, and G. Minafò, *Shear strength of high-strength concrete beams: Modeling and design recommendations.* Engineering Structures, 2014. **69**: p. 116-122 DOI: https://doi.org/10.1016/j.engstruct.2014.02.029.

[20] Marí, A., B. Jesús, C. Antoni, O. Eva, and C. and Ribas, *Shear-flexural strength mechanical model for the design and assessment of reinforced concrete beams.* Structure and Infrastructure Engineering, 2015. **11**(11): p. 1399-1419 DOI: https://doi.org/10.1080/15732479.2014.964735.

[21] Plevris, V., *A Glimpse into the Future of Civil Engineering Education: The New Era of Artificial Intelligence, Machine Learning, and Large Language Models.* Journal of Civil Engineering Education, 2025. **151**(2): p. 02525001 DOI: https://doi.org/10.1061/JCEECD.EIENG-2193.

[22] Plevris, V. and G. Papazafeiropoulos, *AI in Structural Health Monitoring for Infrastructure Maintenance and Safety.* Infrastructures, 2024. **9**(12) DOI: https://doi.org/10.3390/infrastructures9120225.

[23] Plevris, V., *AI-Driven Innovations in Earthquake Risk Mitigation: A Future-Focused Perspective.* Geosciences, 2024. **14**(9): p. 244 DOI: https://doi.org/10.3390/geosciences14090244.

[24] Plevris, V., A. Ahmad, and N.D. Lagaros, eds. *Artificial Intelligence and Machine Learning Techniques for Civil Engineering*. 2023, IGI Global. DOI https://doi.org/10.4018/978-1-6684-5643-9.

[25] Kandil, N., R. Wamkeue, M. Saad, and S. Georges, *An efficient approach for short term load forecasting using artificial neural networks.* International Journal of Electrical Power & Energy Systems, 2006. **28**(8): p. 525-530 DOI: https://doi.org/10.1016/j.ijepes.2006.02.014.

[26] Li, K. and J.-X. Peng, *Neural input selection—A fast model-based approach.* Neurocomputing, 2007. **70**(4): p. 762-769 DOI: https://doi.org/10.1016/j.neucom.2006.10.011.

[27] Ahmad, A., N.D. Lagaros, and D.M. Cotsovos, *Neural Network-Based Prediction: The Case of Reinforced Concrete Members under Simple and Complex Loading.* Applied Sciences, 2021. **11**(11) DOI: https://doi.org/10.3390/app11114975.

[28] Solorzano, G. and V. Plevris, *ANN-based surrogate model for predicting the lateral load capacity of RC shear walls*, in *8th European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS 2022)*. 2022: Oslo, Norway. DOI: https://doi.org/10.23967/eccomas.2022.050.

[29] Solorzano, G. and V. Plevris, *Computational intelligence methods in simulation and modeling of structures: A state-of-the-art review using bibliometric maps.* Frontiers in Built Environment, 2022. **8** DOI: https://doi.org/10.3389/fbuil.2022.1049616.

[30] Solorzano, G. and V. Plevris, *DNN-MLVEM: A Data-Driven Macromodel for RC Shear Walls Based on Deep Neural Networks.* Mathematics, 2023. **11**(10): p. 2347 DOI: https://doi.org/10.3390/math11102347.

[31] Kotsovos, G.M. and N.S. Zygouris, *Reinforced concrete frame analysis with mode of failure prediction capability.* Magazine of Concrete Research, 2018. **71**(3): p. 109-125 DOI: https://doi.org/10.1680/jmacr.17.00092.

[32] Ahmad, A., *Reinforced concrete (RC) structures analysis and assessment with artificial neural networks (ANNs)*. 2018, Heriot Watt University.

[33] Solorzano, G. and V. Plevris, *An Open-Source Framework for Modeling RC Shear Walls Using Deep Neural Networks.* Advances in Civil Engineering, 2023. **2023**: p. 7953869 DOI: https://doi.org/10.1155/2023/7953869.

[34] Ahmad, A. and D.M. Cotsovos, *Reliability analysis of models for predicting T-beam response at ultimate limit response.* Proceedings of the Institution of Civil Engineers - Structures and Buildings, 2023. **176**(1): p. 28-50 DOI: https://doi.org/10.1680/jstbu.20.00129.

[35] Ahmad, A., et al., *Knowledge-Based Prediction of Load-Carrying Capacity of RC Flat Slab through Neural Network and FEM.* Mathematical Problems in Engineering, 2021. **2021**(1): p. 4528945 DOI: https://doi.org/10.1155/2021/4528945.

[36] Ahmad, A. and D.M. Cotsovos, *Reliability Analysis of RC Code for Predicting Load-Carrying Capacity of RCC Walls Through ANN*, in *Artificial Intelligence and Machine Learning Techniques for Civil Engineering*, V. Plevris, A. Ahmad, and N.D. Lagaros, Editors. 2023, IGI Global: Hershey, PA, USA. p. 216-246. DOI https://doi.org/10.4018/978-1-6684-5643-9.ch009.

[37] Ahmad, A., D.M. Cotsovos, and N.D. Lagaros, *Framework for the development of artificial neural networks for predicting the load carrying capacity of RC members.* SN Applied Sciences, 2020. **2**(4): p. 1-21 DOI.

[38] Ahmad, A., M. Elchalakani, N. Elmesalami, A. El Refai, and F. Abed, *Reliability analysis of strength models for short-concrete columns under concentric loading with FRP rebars through Artificial Neural Network.* Journal of Building Engineering, 2021. **42**: p. 102497 DOI: https://doi.org/10.1016/j.jobe.2021.102497.

[39] Ahmad, A., G. Kotsovou, D.M. Cotsovos, and N.D. Lagaros, *Assessing the accuracy of RC design code predictions through the use of artificial neural networks.* International Journal of Advanced Structural Engineering, 2018 DOI: https://doi.org/10.1007/s40091-018-0202-4.

[40] Plevris, V., G. Solorzano, N.P. Bakas, and M.E.A. Ben Seghier, *Investigation of performance metrics in regression analysis and machine learning-based prediction models*, in *8th European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS 2022)*. 2022: Oslo, Norway. DOI: https://doi.org/10.23967/eccomas.2022.155.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

UNCERTAINTY QUANTIFICATION IN LASER POWDER BED FUSION USING MULTI-PHYSICS NUMERICAL SIMULATION AND COPULA-BASED ANALYSIS

Yunfan Zhang[1], **Maria Nogal**[1], **Iuri B. C. M. Rocha**[1], **Oswaldo Morales-Nápoles**[1]

[1]Faculty of Civil Engineering and Geosciences, Delft University of Technology
Stevinweg 1, 2628 CN Delft, The Netherlands
{Y.Zhang-46, M.Nogal, I.Rocha, O.MoralesNapoles}@tudelft.nl

Abstract. *Laser powder bed fusion (LPBF), a prominent metal-based additive manufacturing (AM) technique, enables the production of complex, neat-net-shape components with minimal material waste and reduced lead times. However, achieving high final product quality is challenging due to numerous process variables and intricate, nonlinear interactions introduced by LPBF's thermal-mechanical mechanisms. This study employs a copula-based analysis using multi-physics numerical simulations to probabilistically map relationships among process and part quality variables. Dependence and tail-dependence analyses are performed to provide deeper insights into variable interactions, enabling the identification of preferred operational windows with a balanced trade-off between product quality and productivity. The developed methodology advances the understanding of uncertainty propagation in LPBF, contributing toward improved process optimization, repeatability, and reliability.*

Keywords: Additive Manufacturing, Powder Bed Fusion, Uncertainty Quantification, Probabilistic Modeling, Copulas, Tail Dependence

1 INTRODUCTION

Additive manufacturing (AM) is a manufacturing technique that builds 3D components by progressively depositing and fusing materials together in a layer-by-layer pattern along a predefined path. In recent years, a type of metal-based AM technique named laser powder bed fusion (LPBF), also known as selective laser melting, has become increasingly popular and broadly applied in various fields, such as the aerospace industry, automotive industry, and medicine. This is mainly attributed to its ability to manufacture neat-net-shape components with highly complicated geometry, minimal material waste, and reduced lead time, in comparison to conventional subtractive manufacturing techniques such as machining and tooling [6].

In any manufacturing process, the primary goal is to produce high-quality components with accurate geometry and reliable mechanical properties. However, in spite of its merits, guaranteeing the repeatability of the LPBF process with satisfied final part quality has been a recurrent research topic. On the one hand, the more sophisticated and dynamic nature of the LPBF process, compared to conventional techniques, introduces additional process variables as sources of uncertainty, making it more challenging to accurately measure and quantify these uncertainties. Furthermore, a complicated multi-physics mechanism is involved in the LPBF process, showing a high-dimensional non-linear relationship between process and part quality variables. Based on these factors, the propagation of uncertainties in process variables and their effect on the variability of part quality variables are highly unpredictable without a systematic analysis method.

In order to address this problem, uncertainty quantification (UQ) techniques have been applied to gain a better understanding of the relationship between process and part quality variables. This is particularly significant in terms of enhancing reliability, guiding robust design decisions, and reducing risk in production. Important efforts have been done in previous works, most of them focus on the (global) sensitivity analysis method either using experimental data or simulation data to directly quantify the variability in part quality variables by varying one or more process variables [17, 13, 18, 19]. In References [16, 12, 21, 2], Bayesian estimation and calibration methods are used to predict the part quality variables in consideration of the associated uncertainties. Some other methods, such as Bayesian networks [9] and genetic programming [8], have been used to perform the UQ analysis. However, most of the existing methods rely on the assumption of simple correlation structures, largely overlooking the possibility of more complex dependencies between variables. Many of these approaches focus on only a small subset of variables, limiting their capacity to capture the full complexity in the multi-parameter LPBF process.

To address these limitations, an analysis method based on copulas is proposed in this work to establish a deeper understanding of the probabilistic relationship among both process variables and part quality variables. Specifically, the strength and structure of the dependencies of the variables are researched in the UQ analysis to identify the key process variables. The tail dependence is evaluated to investigate the possibility of the co-occurrence of extreme values. Based on these analyses, a preferred operational window is suggested to achieve a balance within the trade-off between the final product quality and productivity.

The paper structure is organized as follows: In Section 2, the mechanism of the LPBF process and the relevant process and part quality variables are introduced. In Section 3, the construction of the multi-physics numerical simulation model for the LPBF process and the copula-based analysis are presented. In Section 4, a case study of the Inconel 625 superalloy is introduced,

including the variability for the Monte Carlo (MC) sampling and the configurations for the simulation. In Section 5, the dependence analysis is performed to investigate both the effect of process variables on part quality and the interdependence among the part quality variables themselves.

2 LASER POWDER BED FUSION PROCESS

LPBF is a common type of metal-based AM technique, which manufactures 3D metallic components from material in powder form. In this process, thin layers of metallic powder are deposited on a build platform and selectively fused together by a high-powered laser to build components in a layer-by-layer pattern. This system is mainly composed of three parts: (i) the powder feed system responsible for powder deposition, (ii) the build chamber containing a build platform, powder bed, and inert gas environment, and (iii) the energy delivery system of a laser source. As shown by the process flowchart in Figure 1, the LPBF process starts with a sliced 3D digital model in computer-aided design (CAD). After an optional preheating operation of the build platform, the powder feed system works to evenly deposit a new layer of metallic powder at the top surface of the build platform using a recoater blade. The laser source is navigated through optical control units to scan across the deposed powder layer, melting and fusing the powder together along a pre-defined path of the digital model. Then, the build piston is lowered down, and the operations above are repeated until the last slice of the layer is successfully printed. At the end, the printed component is cooled down and undergoes optional post-processing procedures to improve mechanical properties or surface quality.

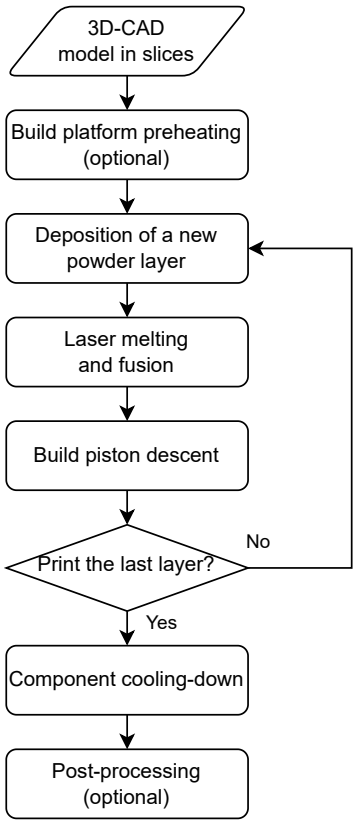


Figure 1: Flowchart of the LPBF process for the manufacturing of a metal component.

The major advantage offered by LPBF is its ability to manufacture near-net-shape components with highly complicated geometry, minimal material waste, and reduced lead times, which is a nearly impossible task for conventional subtractive manufacturing processes. However, the physical process involved in LPBF brings in more complexities. This is mainly reflected in two aspects: (i) The usage of a high-powered laser source and metallic powder feedstock introduces additional process variables as sources of uncertainty, making the accurate quantification of laser and material properties a challenging task. In particular, the powder feedstock is usually more sensitive to the processing environment change due to its loosely distributed state compared to bulk material. Therefore, the variability as well as the change of material properties need to be taken into account before and during the processing cycle. (ii) The LPBF process entails a highly intensive multi-physics mechanism that exhibits a non-linear relationship of high dimensionality between process and part quality variables. This makes a precise prediction of the part quality quite difficult while propagating uncertainty sources through the process model. Therefore, the combined effect of both factors can lead to a final product that does not reach the expected high quality in manufactured components. In order to achieve a stable and reliable LPBF process, a good understanding of the relationship between process variables and part quality variables needs to be established.

There are a large number of process variables that significantly influence the part quality in the LPBF process due to the coexistence of multiple physical models [10]. This work focuses on three types of process variables: variables related to the process itself, the thermal-dependent material properties, and the mechanical material properties. Specifically, these variables are laser power, scan speed, absorptivity, powder density, specific heat capacity, thermal expansion, Young's modulus, and yield stress. The choice of the variables related to the process and thermal properties is primarily based on the existing literature in which dependence has been reported between the melting process and the melt pool evolution [4, 3, 16, 17]. In addition, the mechanical properties are also investigated in this work, as they are playing a crucial role in the formation of distortion and residual strain and stress during the cooling down process. Despite their importance, the effects of thermal expansion, Young's modulus, and yield stress on part quality and the mechanical performance of the final product are largely overlooked in the literature.

After propagating through the non-linear multi-physics model of the LPBF process, the process variables of interest have either a direct or indirect effect on the part quality of manufactured components. To assess the influence of the uncertainty in the process variables, a set representative part quality variables is selected. It consists of the melt pool characteristics during the LPBF process, namely length, height, width, and maximum temperature of the melt pool. These four variables are considered because the melt pool status is one of the most important factors that determines the solidification process, the formation of defects, and consequently, the part quality of manufactured components. For instance, the melt pool with an increased temperature has a weaker surface tension and a smaller dynamic viscosity. This will lead to a recoil effect due to the strong oscillation in melt pool flow and consequently cause the formation of undesirable defects, such as balling, irregularity, distortion, and spatter behind the melt pool track [15]. In addition, other part quality variables that are directly related to the geometric accuracy of manufactured components are also included. The variables are the maximum displacement and maximum residual strain of the solidified material, which are expected to be kept as small as possible. The residual strain here is the L2-normalized equivalent residual strain, which is simply named residual strain in this work for simplicity.

3 METHODOLOGY

To investigate the relationship between process variables and part quality variables, a numerical simulation model is developed to emulate the multi-physics mechanism of the LPBF process in this work. Compared to experimental data, numerical simulation can synthesize an unlimited amount of data without the need for expensive experimental setup and data collection. Moreover, certain part quality variables, such as melt pool dimension and maximum temperature, pose measurement challenges because of the transient manufacturing environment of high temperature. The adoption of numerical simulation enables us to concentrate on the effect of the variability in process variables without the contamination of other sources of uncertainty. In Section 3.1, the mechanism of the multi-physics simulation of the LPBF process is explained.

The synthetic data from the developed model is then analyzed by a copula-based method from a probabilistic perspective. Compared to the sensitivity analysis pattern commonly applied, copulas can capture the complex and joint dependence structures among multiple variables simultaneously, rather than treating each variable's effect in isolation. In addition, copulas can provide more information about the shape and strength of dependence that goes beyond simple correlation measures, offering a deeper understanding of how variables co-vary under different conditions. The theory of the copula and its analysis are introduced in Section 3.2.

3.1 Multi-physics simulation of the LPBF process

The multi-physics simulation model of the LPBF process is constructed based on the work presented in References [20, 11], where the time-dependent thermal-mechanical problem is solved using a one-way coupling approach. In this approach, the solution of the thermal simulation model is taken as the input of the elastoplastic simulation model to solve for the mechanical behavior of the material under the influence of temperature, as illustrated in Figure 2. The simulation assumes isotropic material behavior and perfect plasticity while neglecting factors such as vaporization and powder geometry to focus on the primary thermal and mechanical phenomena.

The thermal simulation shown in Figure 2 is considered a transient heat transfer problem, where the temperature field T_i at any point in the simulation domain is continually changing at each time step i under the effect of external heat energy input. This process is modeled by the heat conduction equation based on Fourier's law in three dimensions

$$\rho C_p \frac{\partial T}{\partial t} = \nabla \cdot (k \nabla T), \tag{1}$$

where T is the temperature distribution as a function of space and time, ρ is the material density, C_p is the specific heat capacity, and k is the thermal conductivity of the heat flux. Notably, the material properties - density, specific heat capacity, and thermal conductivity - are dependent on temperature and typically modeled by linear regression functions to approximate their values in response to temperature.

The elastoplastic simulation shown in Figure 2 is governed by the equilibrium equation in three dimensions, in which the divergence of the stress tensor $\boldsymbol{\sigma}$ must be equal to zero. That is

$$\nabla \cdot \boldsymbol{\sigma} = 0, \tag{2}$$

which assumes there are no body forces in the material.

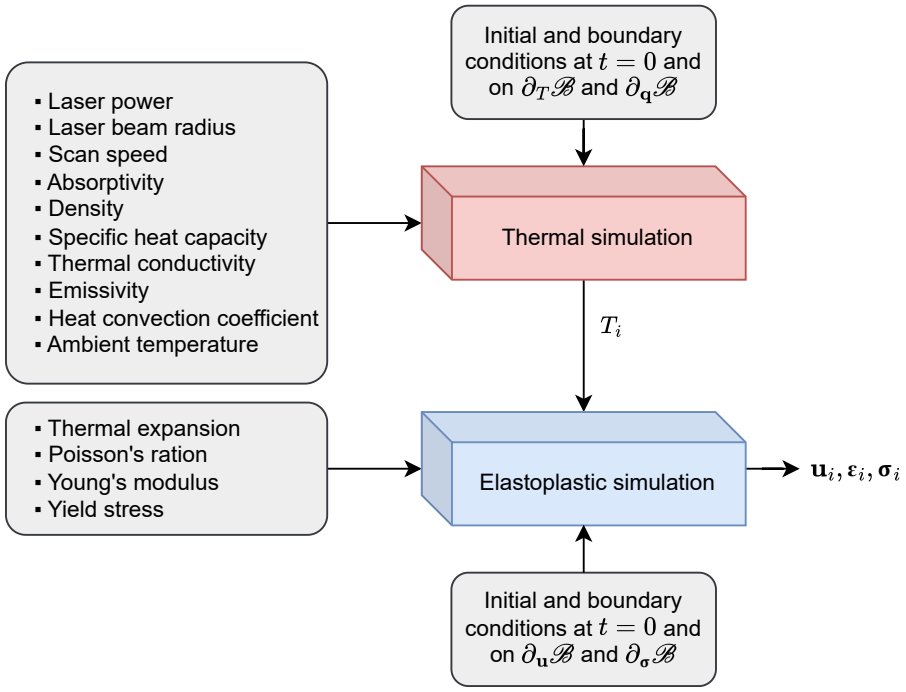


Figure 2: Multi-physics simulation model of the LPBF process using a one-way coupling approach. $(\partial_T \mathcal{B} \cup \partial_{\boldsymbol{q}} \mathcal{B})$ and $(\partial_{\boldsymbol{u}} \mathcal{B} \cup \partial_{\boldsymbol{\sigma}} \mathcal{B})$ are the set of the Dirichlet and Neumann boundaries for the thermal and elastoplastic simulation. T_i, $\boldsymbol{u}_i$, $\boldsymbol{\varepsilon}_i$, and $\boldsymbol{\sigma}_i$ denotes the solution for the temperature, displacement, strain, and stress field at the time step $i = 1, \ldots, n$, respectively.

The constitutive model is defined as J2 perfect plasticity in this case, and the resulting initial boundary value problem (IBVP) is solved using a classic return-mapping algorithm as follows:

$$\begin{aligned} \boldsymbol{\sigma}_{i+1} &= \boldsymbol{\sigma}^{\text{trial}}_{i+1} - \frac{\boldsymbol{s}_{\text{dev}}}{\boldsymbol{s}_{\text{norm}}} \left\langle \boldsymbol{s}_{\text{norm}} - \sigma_y \right\rangle_+ , \\ \boldsymbol{\sigma}^{\text{trial}}_{i+1} &= \boldsymbol{\sigma}_i + \Delta\boldsymbol{\sigma}, \\ \Delta\boldsymbol{\sigma} &= \lambda \text{tr}(\Delta\boldsymbol{\varepsilon}) I + 2\mu \Delta\boldsymbol{\varepsilon}, \end{aligned} \tag{3}$$

where $\boldsymbol{\sigma}^{\text{trial}}$ is the elastic trial stress, $\boldsymbol{s}$ is the deviatoric part of $\boldsymbol{\sigma}^{\text{trial}}$, σ_y is the yield stress, $\boldsymbol{\varepsilon}$ is the strain tensor, and λ and μ are the Lame's parameters. The last term $\left\langle \boldsymbol{s}_{\text{norm}} - \sigma_y \right\rangle_+$ is the yield condition of J2 perfect plasticity, where the operator $\langle \rangle_+$ is defined as a ramp function satisfying $\langle x \rangle_+ = \frac{1}{2}(x + |x|)$. Note that the Lame's parameters are directly derived from the Poisson's ratio ν and the Young's modulus E.

The effect of temperature is taken into account through a thermal strain term $\boldsymbol{\varepsilon}_{\text{th}}$ as follows:

$$\begin{aligned} \Delta\boldsymbol{\varepsilon} &= \boldsymbol{\varepsilon}_{i+1} - \boldsymbol{\varepsilon}_i - \boldsymbol{\varepsilon}_{\text{th}}, \\ \boldsymbol{\varepsilon}_{\text{th}} &= \alpha_V (T_{i+1} - T_i) I, \end{aligned} \tag{4}$$

where α_V is the thermal expansion coefficient. The superscripts in Equations (3) and (4) denote the discretized time step.

As shown in Figure 2, the Dirichlet and Neumann boundary conditions are imposed on both temperature and elastoplastic fields at the set of boundaries $(\partial_T \mathcal{B} \cup \partial_{\boldsymbol{q}} \mathcal{B})$ and $(\partial_{\boldsymbol{u}} \mathcal{B} \cup \partial_{\boldsymbol{\sigma}} \mathcal{B})$,

respectively. Specifically, the heat flux of the laser source q_{laser} is defined at the top surface of the domain and assumed to follow a Gaussian energy distribution as follows:

$$q_{\text{laser}} = \frac{2\eta P}{\pi r_b^2} \exp\left(\frac{-2d^2}{r_b^2}\right), \tag{5}$$

where η is the absorptivity of the material, P is the laser power, r_b is the laser beam radius, and d is the distance between the laser beam center and the material point. Additionally, heat flux terms from convection and radiation are accounted for as described in References [20, 11].

3.2 Copula-based analysis

Copula is a multivariate joint distribution function with uniformly distributed marginals. For a sequence of random variables X_j for $j = 1, \ldots, d$ with a marginal distribution function $F_j(x_j)$, copula is defined by the Sklar's theorem [14] as

$$C(F_1(x_1), ..., F_d(x_d)) = F(x_1, \ldots, x_d), \tag{6}$$

where C is the cumulative distribution function (CDF) of the multivariate copula and F is the joint CDF of the d-dimensional random variables defined on the original scale. In Sklar's theorem, the random variables are standardized by probability integral transform using the marginal distribution function $u_j = F_j(x_j)$, and u_j follows a uniform distribution in the range of $[0, 1]$.

The associated probability density function (PDF) of the multivariate copula can be obtained by calculating its differentiation as

$$c(F_1(x_1), \ldots, F_d(x_d)) = \frac{f(x_1, \ldots, x_d)}{f_1(x_1) \ldots f_d(x_d)}, \tag{7}$$

where c is the copula density and f and f_j are the joint and marginal PDF of the random variables, respectively.

The major advantage offered by the copula lies in its ability to separate the dependence between random variables in spite of the different shapes and ranges of associated marginal distributions. Therefore, it is possible to construct a unique copula for multiple random variables with arbitrary marginal distributions while keeping the dependence structure unchanged. In this work, we primarily focus on bivariate copulas, as they provide a fundamental framework for examining the dependence between two random variables. Numerous families of bivariate copulas - such as Gaussian, Student's t, and various Archimedean types - have been developed. Interested readers are referred to [5] for a detailed overview.

To quantitatively assess the strength of dependence between two random variables, the rank-based correlation coefficient, Kendall's tau τ, is commonly used, which is defined as the probability of concordance minus the probability of discordance of two random variables X_1 and X_2 [5]. Kendall's tau has a range of $[-1, 1]$, with 0 indicating complete independence, 1 for perfect positive correlation, and -1 for perfect negative correlation. Compared to the conventional Pearson correlation coefficient, Kendall's tau is based on the rank and, therefore not dependent on the actual values and ranges of random variables.

In addition to the rank correlation, copula can provide more information with respect to the dependence structure through the tail dependence analysis. The upper and lower tail dependence coefficient of a bivariate copula is defined as

$$\begin{aligned}\lambda^{\text{upper}} &= \lim_{m\to 1^-} P(X_2 > F_2^{-1}(m)|X_1 > F_1^{-1}(m)) = \lim_{m\to 1^-} \frac{1-2m+C(m,m)}{1-m},\\ \lambda^{\text{lower}} &= \lim_{m\to 0^+} P(X_2 \leq F_2^{-1}(m)|X_1 \leq F_1^{-1}(m)) = \lim_{m\to 0^+} \frac{C(m,m)}{m},\end{aligned} \tag{8}$$

where the limiting probability of simultaneous extreme movements in the upper and lower tails are quantified, respectively [5]. The analysis of the tail dependence is of significant importance, particularly for the accurate assessment of extreme risks during manufacturing.

4 APPLICATION CASE

In this work, a numerical simulation focused on single-track scanning is performed to research the thermo-mechanical behavior of the phase-changing material to input process variables in the LPBF process. Specifically, a nickel-based superalloy, Inconel 625, is employed due to the extensive research [20] and industrial experience available on processing this material using AM and LPFB techniques, making it an ideal reference material for the investigation.

An MC simulation is conducted by stochastically sampling the process variables of interest described in Section 2 and using them as inputs to the multi-physics model described in Section 3.1. As a result, the PDFs for the part quality variables are obtained. The mean and standard deviation values for the process variables of Inconel 625 are listed in Table 1. The selection of the mean values is based on previous works and industry standards [1, 19]. A standard deviation of $\pm 10\%$ is assumed to facilitate the analysis of the variables, while the actual variability can be deviated from these values in practical applications. As for the thermal-dependent characteristics modeled by linear regression functions and the values of the constant process variables, we simply adopted the property data from Reference [19] and assumed that the powder material of Inconel 625 shares the same thermal-dependent properties as the bulk material.

Process variable	Units	Mean value	Standard deviation
Laser power, P	W	195	$\pm 10\%$
Scan speed, v_l	mm/s	800	$\pm 10\%$
Density, ρ	kg/m^3	8604	$\pm 10\%$
Specific heat capacity, C_p	J/kg·K	428	$\pm 10\%$
Absorptivity, η	-	0.31	$\pm 10\%$
Thermal expansion coefficient, α	µm/m·K	14	$\pm 10\%$
Young's modulus, E	GPa	150	$\pm 10\%$
Yield stress, σ_y	MPa	650	$\pm 10\%$

Table 1: Mean values and standard deviations of the studied process variables. All the variables are assumed to follow Generalized Beta distributions.

The process variables are assumed to follow Generalized Beta distributions $X_j \sim \text{GBeta}(\alpha, \beta, a, b)$, where X_j denotes the process variable of interest, α and β are the shape parameters, and a and b are the bounds of the support range. The support range is considered as $[0.6\mu_{X_j}, 1.4\mu_{X_j}]$, which corresponds to 4 times the standard deviation from the mean. An advantage of using the Generalized Beta distribution is that the sampled variables are strictly bounded within a support range while keeping a shape resembling a Gaussian distribution, avoiding the sampling of invalid values.

The open-source Python library JAX-FEM [20] is used in this work to construct a numerical simulation model for the LPBF process. The simulation object has a dimension of $3.2 \times 0.8 \times 0.4$ mm^3 in length, width, and depth, with a mesh element size of 20 μm in each direction. The total simulation time is set to three times the laser-on duration, using a time increment of 5 μs. This simulation setting is determined based on a comparison experiment in which various combinations of mesh element sizes and time increments were tested. The combination offering the optimal trade-off between computational time and accuracy was selected. The thermal distribution over the simulation object is computed at every time step, while the mechanical behavior is evaluated every tenth time step in order to reduce computational overhead, as recommended by the library developer [20]. The simulation is run 500 times on TU Delft's supercomputer, DelftBlue [7], producing a sufficient amount of data for further analysis.

5 RESULTS AND DISCUSSION

Based on the MC simulation results, the relationship between (i) each pair of process and part quality variables and (ii) among part quality variables is investigated from a probabilistic perspective. Specifically, a series of bivariate copula functions is fitted for each pair of variables, and the one with the lowest prediction error based on the Akaike information criterion (AIC) is considered the best-fitted copula. Figure 3 shows the rank correlation matrix between the process and part quality variables. Then, the amount and structure of the dependence between the variables are quantified based on the fitted copula and its rank correlation coefficient. Finally, the interaction of the variables in the tail of the fitted copula, that is, the tail dependence, is qualitatively analyzed in terms of its physical implication in the LPBF process.

In Section 5.1, a dependence analysis between the process and part quality variables is performed to investigate the influence of the process variables. In Section 5.2, a dependence analysis solely between the part quality variables is presented to understand how they are correlated to each other and consequently influence the final quality of manufactured components. For the full list of the best-fitted copulas for each pair of the variables, refer to Appendix A.

5.1 Dependence analysis between process and part quality variables

The discussion is structured based on the type of part quality variables involved. Namely, (i) the effect of process variables on melt pool dimension, (ii) maximum temperature, and (iii) maximum displacement and residual strain.

5.1.1 Effect of process variables on melt pool dimension

The stability of the melt pool is one of the most important factors in the LPBF process, as it has a significant impact on the following physical processes, such as microstructure and defect formation, which are directly related to the final quality of manufactured components. Melt pool dimension is a crucial metric to determine the stability of the melt pool development. For instance, a sudden change in the melt pool dimension may imply the occurrence of a mechanical malfunction in operation devices. Moreover, a precise estimation of melt pool dimension in advance can facilitate the optimization of process variables, such as hatch distance and layer thickness, to avoid the occurrence of an unmelted region or porosity due to insufficient or excessive energy input.

Based on the rank correlation coefficient between each process variable and melt pool dimension (last three rows of the matrix shown in Figure 3), laser power and absorptivity are the dominant factors with the largest impact on the length, width, and depth of the melt pool. With

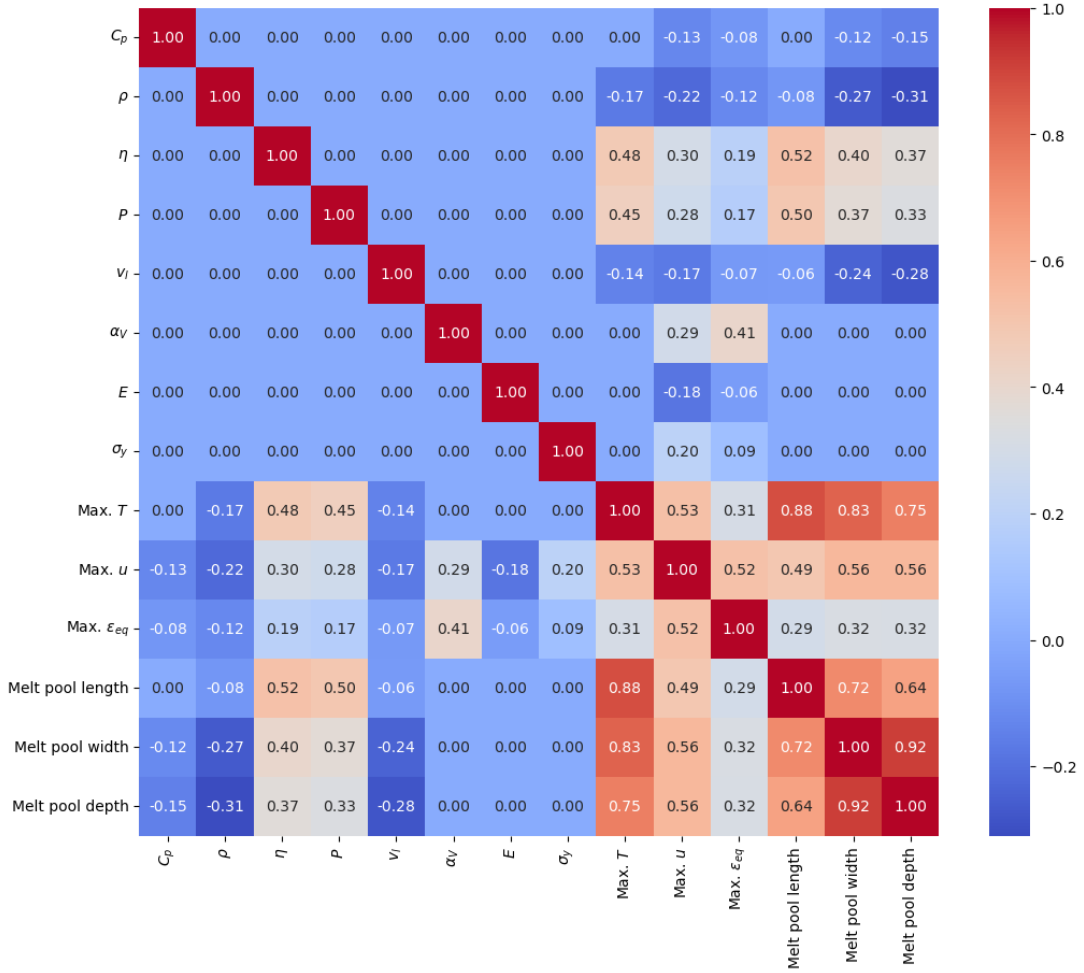


Figure 3: Rank correlation matrix using Kendall's tau between the process and part quality variables and among part quality variables. The rank correlation between process variables to each other is set to 0, as they are sampled from independent GBeta distributions. Most of the variables are defined in Section 2 and 3. u is the magnitude of the displacement and ε_{eq} is the L2-normalized equivalent residual strain.

an increase in absorptivity and laser power, the melt pool size expands correspondingly as a result of a higher energy input. Following that, scan speed, density, and specific heat capacity are less influential, exhibiting a negative correlation to melt pool width and depth. The rest of the process variables (i.e., thermal expansion, Young's modulus, and yield stress) are independent of melt pool dimension. It is noticed that the melt pool length is largely governed by the direct energy input - namely, laser power and absorptivity - while secondary factors such as scan speed, density, and specific heat capacity mostly influence heat conduction in the transverse directions (i.e., width and depth).

The best-fitted copulas for each pair of the process variables and melt pool dimensions are represented in the last three columns of Figure A.3 in Appendix A. It is observed that most of the copulas are Gaussian and elliptically symmetric, and no significant tail dependence is observed. This means that the relationships between the process variables and melt pool dimensions do not exhibit pronounced extreme co-movement at the tails, indicating a consistent correlation structure across the entire data range. The analysis of the tail dependence is of particular importance, as it provides important information about the relationship of associated variables in

extreme conditions to facilitate risk assessment and management.

5.1.2 Effect of process variables on maximum temperature

Similarly, it is observed in Figure 3 that the most influential variables on maximum temperature are laser power and absorptivity, followed by density and scan speed with less impact, and the others are considered independent. This is explained by the heat conduction mechanism introduced in Section 3.1, where more heat energy input, i.e., increased laser power and absorptivity, means a higher maximum temperature and vice versa.

In the first row in Figure A.3, the copulas between the process variables and maximum temperature do not show any tail dependence at both ends, meaning that uniform correlations are maintained across the entire domain.

5.1.3 Effect of process variables on maximum displacement and residual strain

Rows 10 and 11 in Figure 3 show that all the process variables share a similar amount of dependence on maximum displacement with the largest absolute difference of 0.17 between absorptivity and specific heat capacity, whereas only four process variables show evident dependence on maximum residual strain, with thermal expansion as the most dominant factor with a correlation up to 0.41. The process variables that show very low correlations, less than 0.1, are considered independent. Generally speaking, the effect of the process variables on maximum displacement and residual strain demonstrates a similar trend. For instance, an increased heat energy input leads to an increase in both maximum displacement and residual strain. The material that is more sensitive to thermal effects with a greater thermal expansion will accumulate more displacement and residual strain. The main difference is that the residual strain is basically only related to the uncertainties in thermal-dependent properties, while the maximum displacement is influenced by both thermal-dependent and mechanical properties.

By analyzing the tail dependence shown by the best-fitted copulas in the second and third columns of Figure A.3, more information can be extracted in terms of the behavior of maximum displacement and residual strain in extreme conditions. For instance, laser power and absorptivity exhibit a lower tail dependence with respect to maximum displacement. This could potentially be a beneficial property that can be leveraged on purpose to produce geometrically precise components by reducing these process variables by a large margin. On the other hand, the upper tail of laser power, scan speed, and absorptivity is the design space that is expected to avoid as much as possible, as it will lead to a rapid increase in the amount of displacement. The tail dependence of maximum residual strain has a similar but weaker pattern compared to maximum displacement.

Based on the observations above, the synergy effect of both thermal-dependent and mechanical material properties must be delicately considered and designed to control the maximum displacement to a minimum amount. However, there is a trade-off that must be noted in the LPBF process, i.e., the trade-off between final product quality and productivity. Although less heat energy input can produce components with higher accuracy and finer structure, this is usually accompanied by a narrower melt pool and a lower maximum temperature based on the dependent relationships discussed in Section 5.1.1 and 5.1.2, consequently leading to a drop in process stability and productivity.

In addition, materials with a higher Young's modulus and lower yield stress seem to be an appealing choice as they exhibit lower maximum displacement consistently. However, because

the simulation model does not explicitly distinguish between the elastic and plastic strain, the extent of plastic deformation remains unclear. This is potentially detrimental, as a less resilient material is prone to experience more plastic flow rather than staying in an elastic regime, leading to negative impacts on the fatigue behavior of manufactured components. Therefore, further investigation into plastic flow and its response to process parameters is necessary.

5.2 Dependence analysis between part quality variables

Beyond the commonly studied relationship between process and part quality variables, the copula-based analysis allows the investigation of the correlation of part quality variables to each other. This provides an in-depth insight into the LPBF process monitoring from a part quality perspective. In Section 5.2.1, the dependence structure of maximum temperature on other part quality variables is analyzed and discussed. In Section 5.2.2, the focus is mainly on maximum displacement and residual strain. Note that melt pool dimensions are implicitly embedded in the discussion of other part quality variables.

5.2.1 Effect on maximum temperature

In Figure 3, it is observed that the maximum temperature is highly dependent on melt pool dimension (correlation coefficients ranging from 0.75 to 0.88), followed by maximum displacement and residual strain in descending order. This means that an increase in melt pool dimension and maximum displacement and residual strain typically accompanies an increase in maximum temperature, potentially giving rise to unstable melt pool dynamics. In this case, the maximum temperature serves as an effective quality indicator for monitoring melt pool stability in the LPBF process.

By examining the tail dependence structure, we gain further insight into how to maintain a stable operational status with stability in the melt pool dimension and maximum temperature while identifying a preferred operational window with an adequate tradeoff between final product quality and productivity based on qualitative observation. In Figure 4, a strong lower tail dependence is observed between the maximum temperature and other part quality variables, indicating that the part quality variables jointly move toward extreme values. On the other hand, as the part quality variables increase, the maximum temperature exhibits a wider uncertainty range, which poses challenges for process control. Consequently, an operational window close to the lower tail (shown in Figure 4) achieves a favorable balance between final product quality and productivity, providing a reasonable solution for the trade-off problem discussed in 5.1.3. Within this region, the melt pool has a moderate size and is relatively easy to control, while maximum displacement and residual strain remain low.

5.2.2 Effect on maximum displacement and residual strain

In Figure 3, it is observed that maximum displacement and residual strain share a similar pattern of correlation to the part quality variables, with correlation values around 0.50 in the case of maximum displacement and 0.30 in the case of maximum residual strain. The analysis of the part quality variable relationships explains some phenomena that can not be directly derived from their relationship to the process variables. For instance, although the specific heat capacity does not directly affect the maximum temperature, it influences the melt pool dimension through the heat storage and conduction process, which in turn affects the maximum displacement and residual strain.

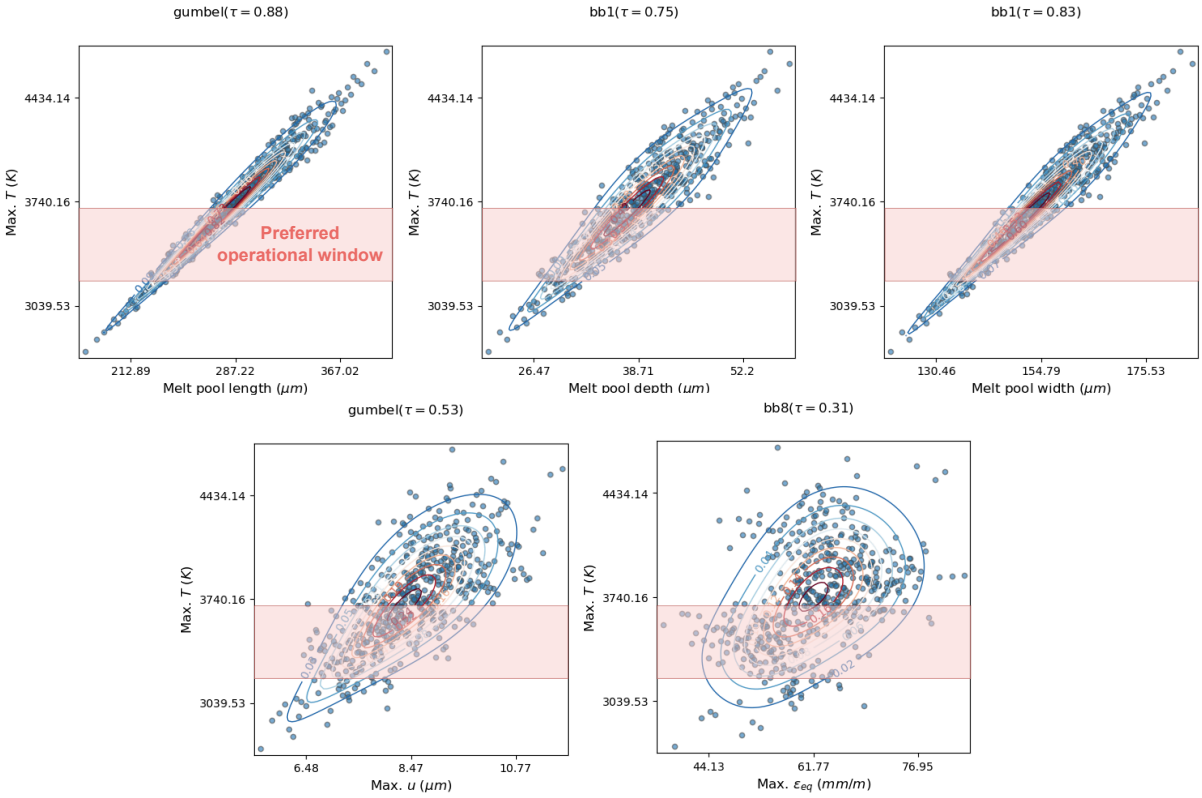


Figure 4: Pair plots of the part quality variables and maximum temperature in a standardized space. The contour curves correspond to the PDF of the best-fitted copula. The type of copula and associated rank correlation coefficient are displayed on the top of the figure. The red box is the preferred operational window that is neither located at the lower tail nor has a large uncertainty range.

The analysis further confirms the existence of the trade-off problem between final product quality and productivity discussed in Section 5.2.2 from a part quality perspective, where a lower degree of distortion is typically accompanied by a narrower and less dynamic melt pool, thus requiring more layers and longer duration for manufacturing the same component. Based on the tail dependence between the part quality variables and maximum displacement and residual strain shown in Figure A.4, it is observed that the pursuit of a very small distortion entails a risk of experiencing a significant drop in melt pool stability. Similar to the analysis procedure discussed in Section 5.2.1, the preferred operational window is located close to the lower tail, as shown in Figure 5, where it is possible to achieve a combination of low distortion but a relatively active melt pool.

6 CONCLUSIONS

In this study, a multi-physics numerical simulation model was used for the LPBF process to enable a UQ analysis of key process variables and their effects on the final product quality. By applying a copula-based approach using simulation data with randomly sampled uncertainties, the nonlinear dependent relationship and tail dependence among both process variables (e.g., laser power, absorptivity, and thermal expansion) and part quality variables (e.g., melt pool dimension, maximum temperature, and maximum displacement and residual strain) are captured. The results revealed that variations in laser power and absorptivity have the most significant impact on the melt pool dimension and maximum temperature, while thermal expansion primarily

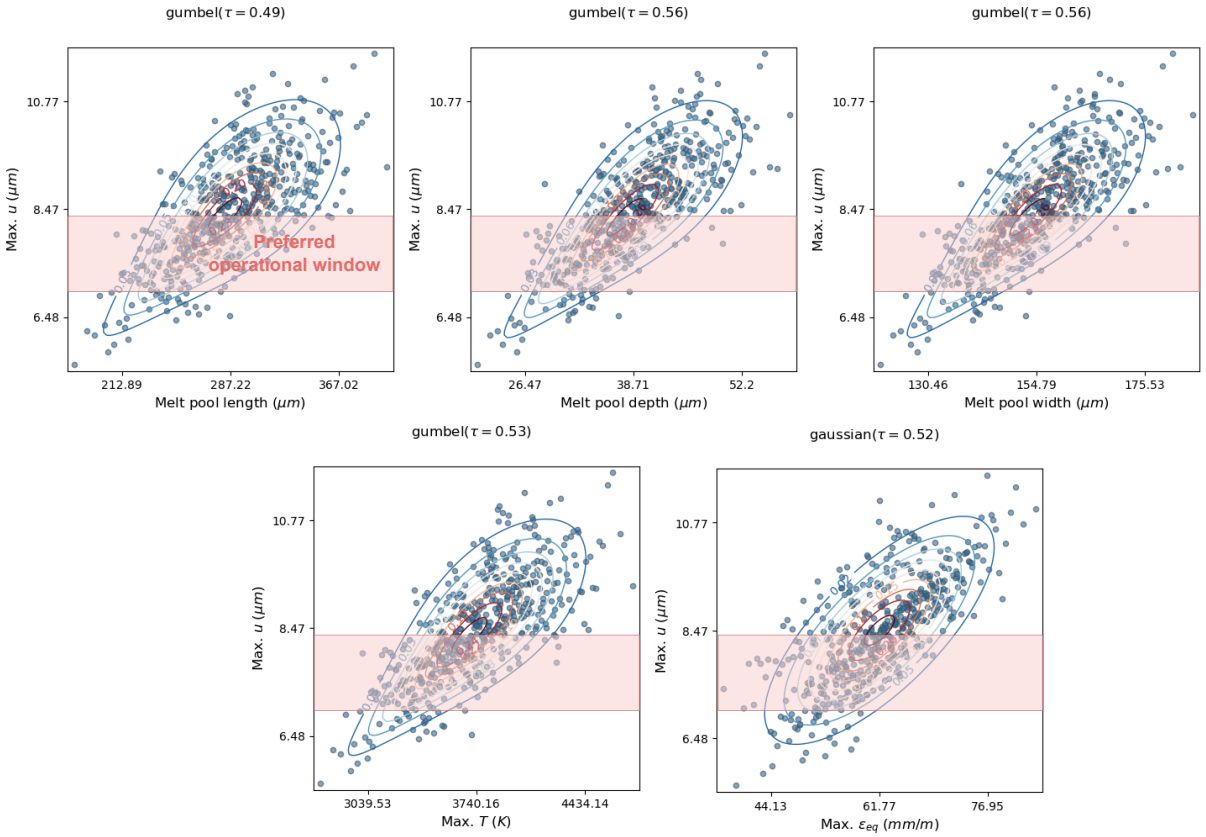


Figure 5: Pair plots of the part quality variables and maximum displacement in a standardized space. The contour curves correspond to the PDF of the best-fitted copula. The type of copula and associated rank correlation coefficient are displayed on the top of the figure. The red box is the preferred operational window that is neither located at the lower tail nor has a large uncertainty range.

drives the variability in maximum residual strain. These findings are in line with previous works [4, 3] and validated from a probabilistic perspective. The maximum displacement is controlled by a synergy effect of both thermal-dependent and mechanical material properties, with each contributing to a similar degree. This is, to our knowledge, the first work to highlight the effect of process variables on maximum displacement. Importantly, tail-dependent relationships support the fact that extreme values in maximum displacement often coincide with extreme thermal or mechanical responses. Based on these analyses, a trade-off problem between the final product quality and productivity was defined in the LPBF process, and a preferred operational window was suggested based on a qualitative observation.

By identifying the dominant process variables that cause significant deviation in part quality from design values, this research advances toward the optimization and control of various parameters in LPBF to enhance repeatability and reliability. For instance, this approach has shown the potential to define operational windows that maximize both the final product quality and productivity. Further efforts are needed in reverse modeling to project these qualitatively observed operational windows back to specific process variables. Together with the relevant physical and technical constraints, the process variable settings that optimize the trade-off between the final product quality and productivity can be determined.

This study has potential limitations, including theoretical assumptions in the numerical simulations, simplified uncertainty ranges that may differ from realistic industrial conditions, and

restriction to bivariate copulas. Future work will address these limitations by incorporating experimental observations through a dependency modeling method of more complexities.

7 ACKNOWLEDGMENTS

We gratefully acknowledge the European Commission for its support of the Marie Sklodowska Curie program through the Horizon Europe DN APRIORI project (GA 101073551).

REFERENCES

[1] Inconel 625. *Wikipedia*, Nov. 2024.

[2] M. Chiappetta, C. Piazzola, L. Tamellini, A. Reali, F. Auricchio, and M. Carraturo. Data-informed uncertainty quantification for laser-based powder bed fusion additive manufacturing. *International Journal for Numerical Methods in Engineering*, 125(17):e7542, 2024.

[3] L. E. Criales, Y. M. Arısoy, B. Lane, S. Moylan, A. Donmez, and T. Özel. Laser powder bed fusion of nickel alloy 625: Experimental investigations of effects of process parameters on melt pool size and shape with spatter analysis. *International Journal of Machine Tools and Manufacture*, 121:22–36, Oct. 2017.

[4] L. E. Criales, Y. M. Arısoy, and T. Özel. Sensitivity analysis of material and process parameters in finite element modeling of selective laser melting of Inconel 625. *The International Journal of Advanced Manufacturing Technology*, 86(9):2653–2666, Oct. 2016.

[5] C. Czado. *Analyzing Dependent Data with Vine Copulas: A Practical Guide With R*, volume 222 of *Lecture Notes in Statistics*. Springer International Publishing, Cham, 2019.

[6] T. DebRoy, H. L. Wei, J. S. Zuback, T. Mukherjee, J. W. Elmer, J. O. Milewski, A. M. Beese, A. Wilson-Heid, A. De, and W. Zhang. Additive manufacturing of metallic components – Process, structure and properties. *Progress in Materials Science*, 92:112–224, Mar. 2018.

[7] Delft High Performance Computing Centre (DHPC). DelftBlue Supercomputer (Phase 2). `https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase2`, 2024.

[8] I. Gholaminezhad, H. Assimi, A. Jamali, and D. A. Vajari. Uncertainty quantification and robust modeling of selective laser melting process using stochastic multi-objective approach. *The International Journal of Advanced Manufacturing Technology*, 86(5):1425–1441, Sept. 2016.

[9] N. Hertlein, S. Deshpande, V. Venugopal, M. Kumar, and S. Anand. Prediction of selective laser melting part quality using hybrid Bayesian network. *Additive Manufacturing*, 32:101089, Mar. 2020.

[10] Z. Hu and S. Mahadevan. Uncertainty quantification and management in additive manufacturing: Current status, needs, and opportunities. *The International Journal of Advanced Manufacturing Technology*, 93(5):2855–2874, Nov. 2017.

[11] S. Liao, A. Golgoon, M. Mozaffar, and J. Cao. Efficient GPU-accelerated thermomechanical solver for residual stress prediction in additive manufacturing. *Computational Mechanics*, 71(5):879–893, May 2023.

[12] F. Lopez, P. Witherell, and B. Lane. Identifying Uncertainty in Laser Powder Bed Fusion Additive Manufacturing Models. *Journal of Mechanical Design*, 138(114502), Sept. 2016.

[13] P. Nath, Z. Hu, and S. Mahadevan. Modeling and Uncertainty Quantification of Material Properties in Additive Manufacturing. In *2018 AIAA Non-Deterministic Approaches Conference*, AIAA SciTech Forum. American Institute of Aeronautics and Astronautics, Jan. 2018.

[14] M. Sklar. Fonctions de répartition à N dimensions et leurs marges. *Annales de l'ISUP*, VIII(3):229–231, 1959.

[15] J. Wang, R. Zhu, Y. Liu, and L. Zhang. Understanding melt pool characteristics in laser powder bed fusion: An overview of single- and multi-track melt pools for process optimization. *Advanced Powder Materials*, 2(4):100137, Oct. 2023.

[16] Z. Wang, C. Jiang, P. Liu, W. Yang, Y. Zhao, M. F. Horstemeyer, L.-Q. Chen, Z. Hu, and L. Chen. Uncertainty quantification and reduction in metal additive manufacturing. *npj Computational Materials*, 6(1):1–10, Nov. 2020.

[17] Z. Wang, P. Liu, Y. Ji, S. Mahadevan, M. F. Horstemeyer, Z. Hu, L. Chen, and L.-Q. Chen. Uncertainty Quantification in Metallic Additive Manufacturing Through Physics-Informed Data-Driven Modeling. *JOM*, 71(8):2625–2634, Aug. 2019.

[18] Z. Wang, P. Liu, Y. Xiao, X. Cui, Z. Hu, and L. Chen. A Data-Driven Approach for Process Optimization of Metallic Additive Manufacturing Under Uncertainty. *Journal of Manufacturing Science and Engineering*, 141(081004), June 2019.

[19] S. Wells, A. Plotkowski, and M. J. M. Krane. Propagation of Input Uncertainties in Numerical Simulations of Laser Powder Bed Fusion. *Metallurgical and Materials Transactions B*, 52(5):3016–3031, Oct. 2021.

[20] T. Xue, S. Liao, Z. Gan, C. Park, X. Xie, W. K. Liu, and J. Cao. JAX-FEM: A differentiable GPU-accelerated 3D finite element solver for automatic inverse design and mechanistic data science. *Computer Physics Communications*, 291:108802, Oct. 2023.

[21] J. Ye, M. Mahmoudi, K. Karayagiz, L. Johnson, R. Seede, I. Karaman, R. Arroyave, and A. Elwany. Bayesian Calibration of Multiple Coupled Simulation Models for Metal Additive Manufacturing: A Bayesian Network Approach. *ASCE-ASME J Risk and Uncert in Engrg Sys Part B Mech Engrg*, 8(011111), Oct. 2021.

A FITTED BIVARIATE COPULA FUNCTIONS

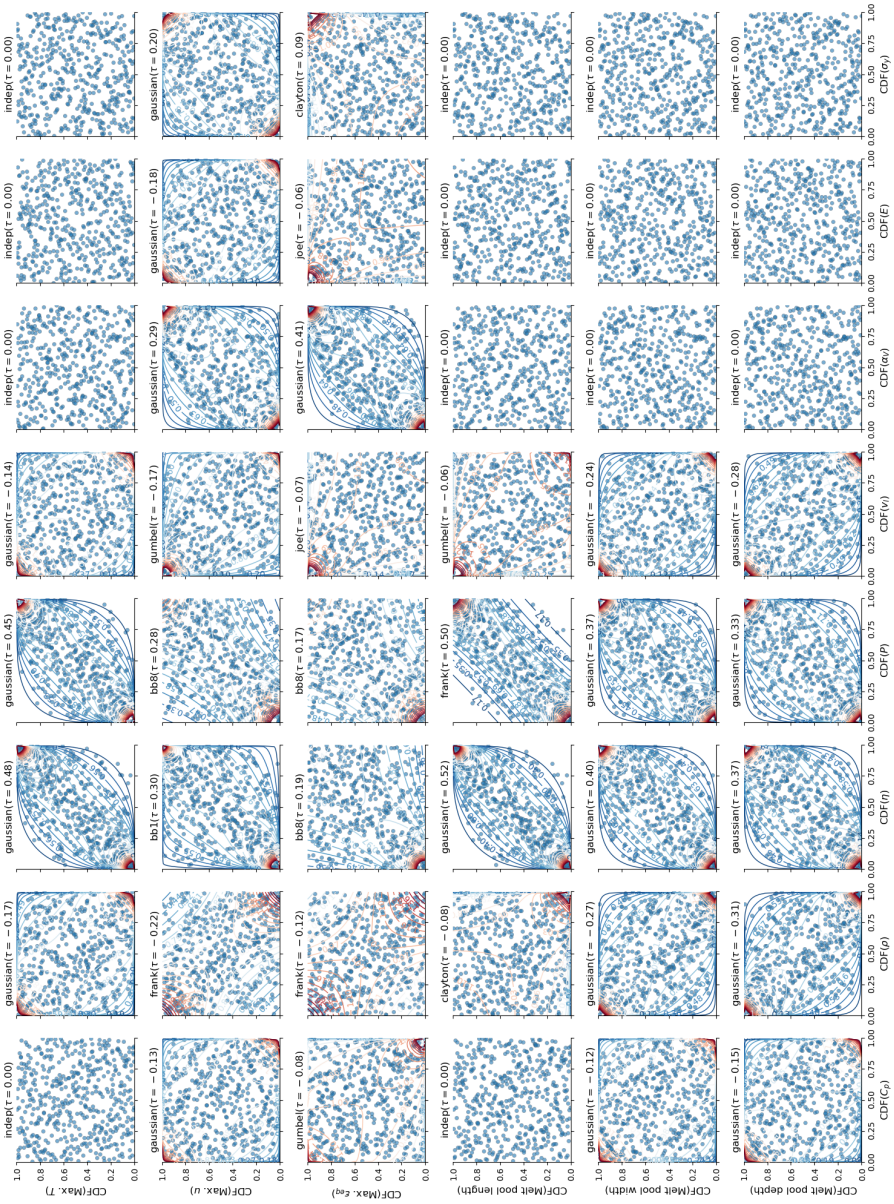


Figure A.1: Pair plots between process and part quality variables, where the contour of the best-fitted copula density function is displayed. Each variable is transformed to uniform space using empirical marginal CDFs. The type and rank correlation coefficient of the best-fitted copula is displayed on the left-hand side of each pair plot.

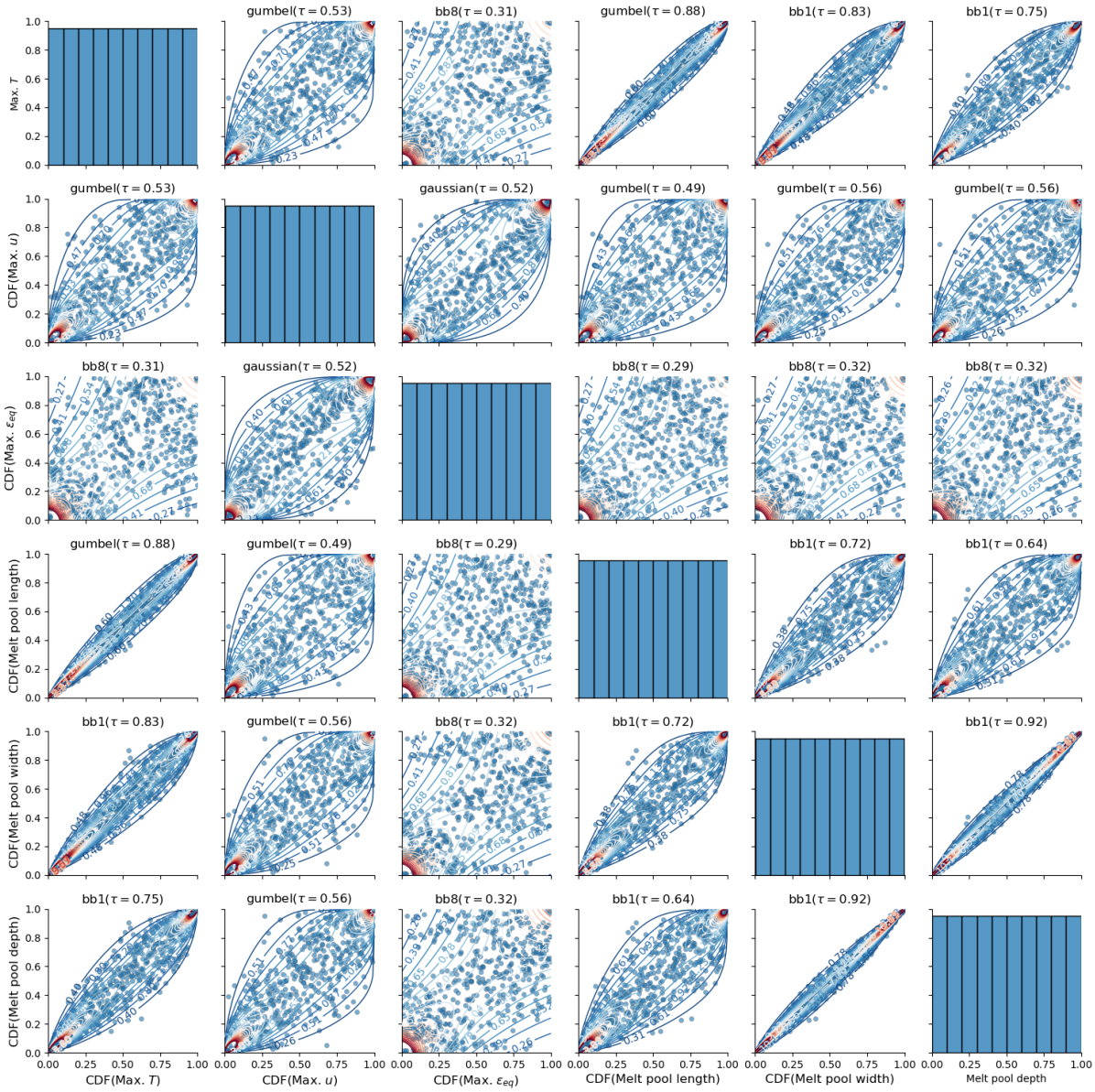


Figure A.2: Pair plots between part quality variables, where the contour of the best-fitted copula density function is displayed. Each variable is transformed to uniform space using empirical marginal CDFs. The type and rank correlation coefficient of the best-fitted copula is displayed on the top of each pair plot. The histogram corresponding to the marginal PDF of each variable is shown in the diagonal.

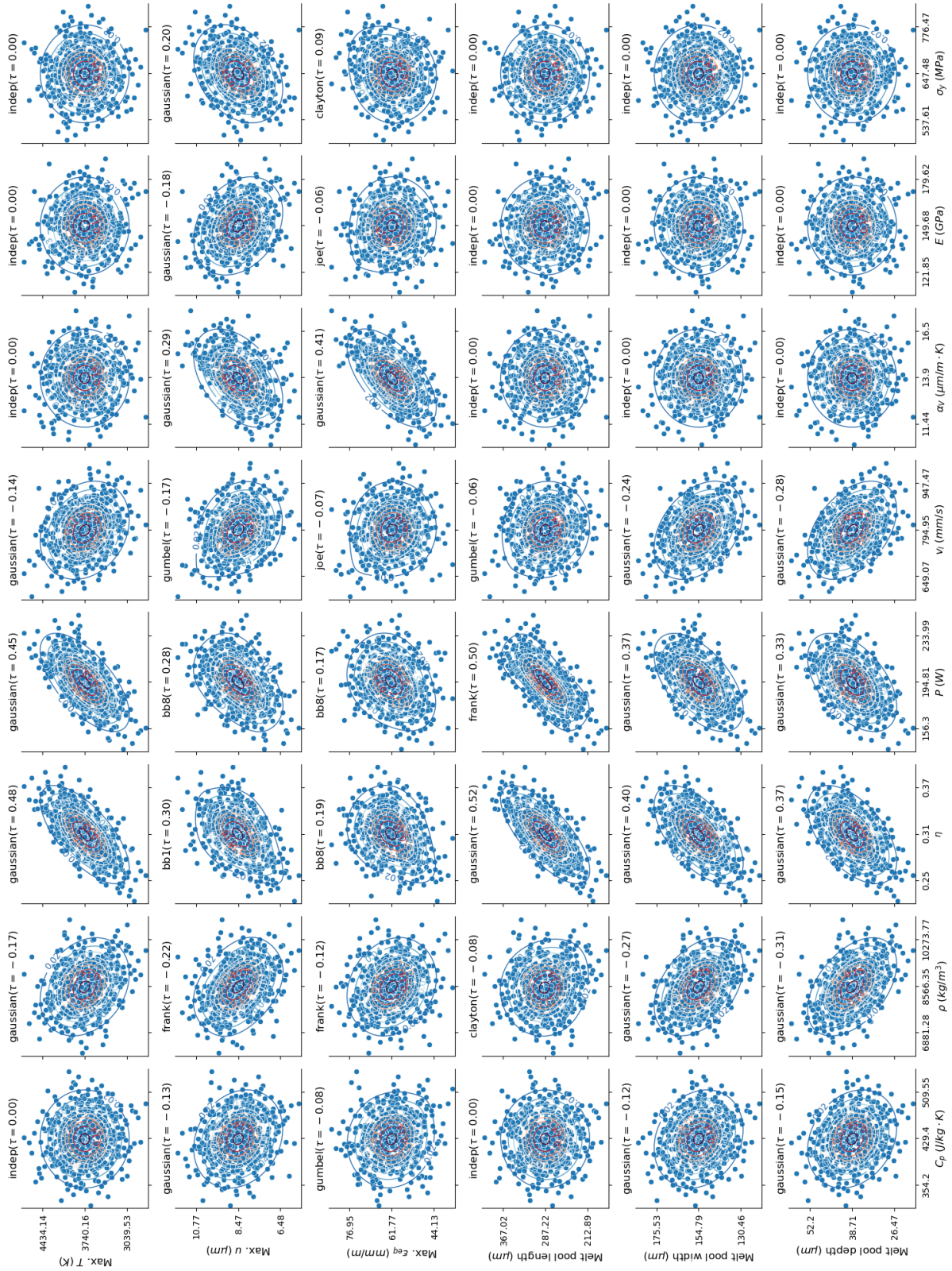


Figure A.3: Pair plots between process and part quality variables, where the contour of the best-fitted copula density function is displayed. Each variable is re-scaled to standardized space by performing inverse univariate Gaussian PDF on the uniform marginal $X_i = \phi^{-1}(u_i)$ and mapped to the original scale. The type and rank correlation coefficient of the best-fitted copula is displayed on the left-hand side of each pair plot.

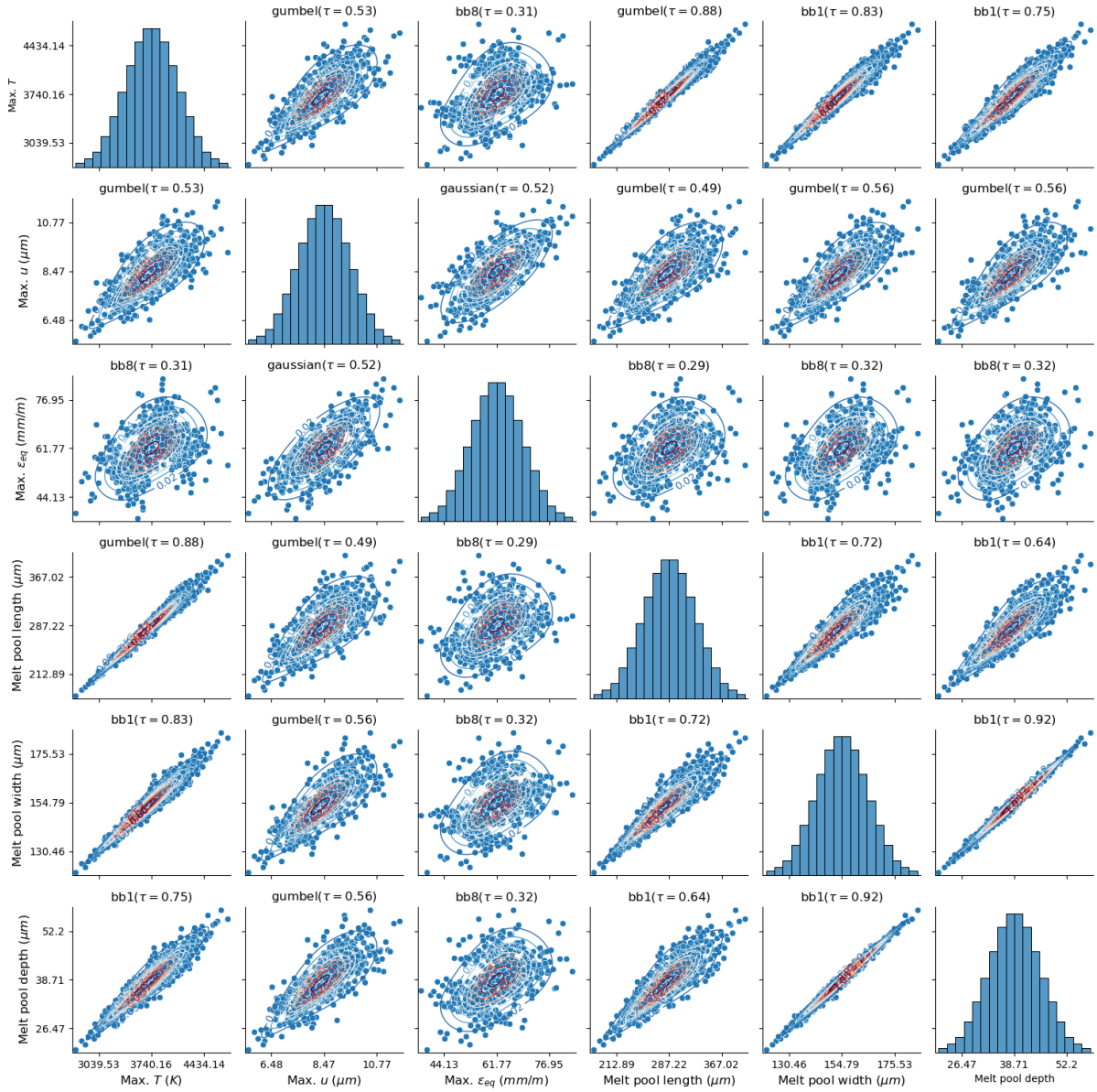


Figure A.4: Pair plots between part quality variables, where the contour of the best-fitted copula density function is displayed. Each variable is re-scaled to standardized space by performing inverse univariate Gaussian PDF on the uniform marginal $X_i = \phi^{-1}(u_i)$ and mapped to the original scale. The type and rank correlation coefficient of the best-fitted copula is displayed on the top of each pair plot. The histogram corresponding to the marginal PDF of each variable is shown in the diagonal.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

UNCERTAINTY QUANTIFICATION FOR THE SIMULATION OF AUTOMOTIVE COMPONENT LOADS USING STOCHASTIC SURROGATE MODELS

Paul Straehle[1,2], Steve Wolff-Vorbeck[1], Thomas Leyendecker[1] and Carsten Proppe[2]

[1]Robert Bosch GmbH
Robert-Bosch-Campus 1, 71272 Renningen, Germany
e-mail: {paul.straehle, steve.wolff-vorbeck, thomas.leyendecker}@de.bosch.com

[2] Institute of Engineering Mechanics, Karlsruhe Institute of Technology
76131 Karlsruhe, Germany
e-mail: carsten.proppe@kit.edu

Abstract. *To design reliable automotive components, it is important to understand the expected operational loads. Early in development, virtual load generation helps derive these loads realistically. Many loads depend on vehicle speeds, making velocity simulation an essential part of virtual load generation. The velocity profiles are determined using a stochastic simulator, which employs a stochastic process with variable parameters to model the variation in driving behavior and generate random velocity profiles for a given route and driver. The velocity profiles are propagated to component loads using use case specific system simulations. The derived velocities and loads are subject to model and data uncertainty. To address this, we propose a framework that models input parameters as random variables in a Bayesian framework, allowing for the handling of uncertainties. To enable the identification of these uncertainties and sampling from the posterior distribution, we apply Polynomial Chaos Expansion and Generalized Lambda Models to derive stochastic surrogate models as a substitute for the velocity simulation. We show that the parameter distributions derived with the surrogate provide valid descriptions of the velocity simulation uncertainty and validate the results with respect to measured velocities. As a result, our study yields a broad probabilistic framework for data-driven reliability prediction of automotive components and enables the consideration of load estimation uncertainties in an early stage of component development.*

Keywords: Uncertainty Quantification, stochastic surrogate model, Bayesian inference, velocity simulation

1 Introduction

The established approach to estimate the loads on automotive components is to perform test drives under certain assumptions of the actual usage of the car by a customer. From these assumptions, a test route is derived which is then driven in a specific manner by a test driver. The loads measured on this route are then extrapolated to the targeted lifetime of the component. However, due to the high uncertainties associated with the assumptions and the very limited number of kilometers driven, this approach can only give a limited view into the loads acting on the component over its actual lifetime. In addition, existing physical components (e.g. prototypes or components from previous generations) are required on which the measurements can be carried out. This aspect makes the approach unfeasible for the development of new, innovative components as currently developed for electric and fuel cell electric vehicles. In this case, load measurements must be "virtualized", meaning that they are fully generated by simulations. This requires the modeling of realistic profiles of the speed of the vehicle in the driving direction, referred to in the following as the velocity of the vehicle, along an arbitrary route taking into account influences such as traffic and driving behavior.

In the following, the velocity simulation [1, 2] is a stochastic simulator $\mathcal{M}_v(s, \boldsymbol{r}(s), \boldsymbol{\theta}_v)$ and considers various random influences to generate random velocity profiles for a given route. It produces a stochastic process

$$\{V_s = \mathcal{M}_v(s, \boldsymbol{r}(s), \boldsymbol{\theta}_v) : s \in S\} \tag{1}$$

where s is a position along the route contained in the set of positions S. Moreover, we consider different properties $\boldsymbol{r}(s)$, such as the legal speed limit, curvature, slope, etc., of the road as a function of the position s along the route and $\boldsymbol{\theta}_v$ is the parameter vector describing driver behavior and vehicle properties. To assess component loads the simulated velocities are fed into a system simulation (i.e., transfer function) $\mathcal{M}_g(v(s), \boldsymbol{p}(s), \boldsymbol{\theta}_g)$ which simulates component loads based on the velocity as well as other inputs $\boldsymbol{p}(s)$ and parameters $\boldsymbol{\theta}_g$. Due to the random velocity profiles, the component load is thus also described by a stochastic process

$$\{T_s = \mathcal{M}_g(V_s, \boldsymbol{p}(s), \boldsymbol{\theta}_g) : s \in S\}. \tag{2}$$

In Section 2 we describe a comprehensive Uncertainty Quantification (UQ) framework to handle uncertainties of the velocity simulation in the context of virtual load derivation. To enable the quantification of uncertainty of the velocity simulation we incorporate a Generalized Lambda Model (GLaM) based on the model derived in [3], where the authors used the Generalized Lambda Distribution (GLD) to surrogate the response distribution of a stochastic simulator. In Section 3, the fitting and validation of the stochastic surrogate model is described. Furthermore, in Section 4 a Sobol' indices based sensitivity analysis [4] of the velocity simulation parameters is performed to identify the most important parameters. The posterior distribution of the most important parameters is identified within a Bayesian framework, see Section 5. Finally, in Section 6 an evaluation of the simulation results under consideration of the identified uncertainties is performed.

2 Uncertainty Quantification framework

The UQ framework provides the structure and connects the needed parts of a model to identify uncertainties in the velocity simulation. An overview of the framework is given in Fig. 1. We focus on a use case where we want to identify the uncertainties in the parameter vector $\boldsymbol{\theta}_v$

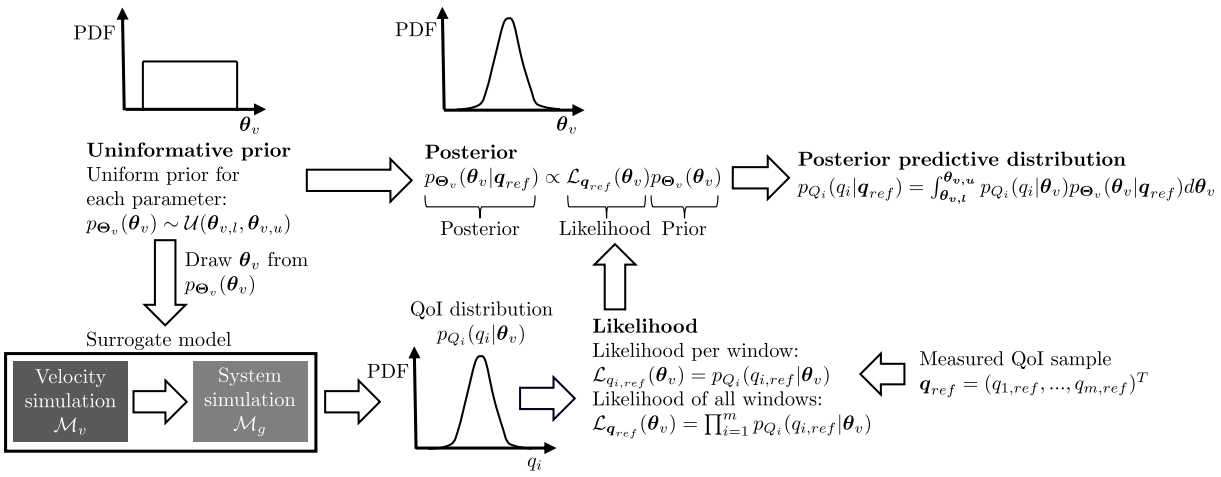


Figure 1: UQ framework

describing the driving behavior of a single driver on a set of routes. The available dataset consists of velocity measurements on $216\,\mathrm{km}$ of city, country, and motorway roads. We consider one specific Quantity of Interest (QoI) which is given by the torque load on the drivetrain of an electric vehicle. However, we note that the method developed in the following can be applied to any other load case. Thus, the system simulation $\mathcal{M}_g(v(s), \boldsymbol{p}(s), \boldsymbol{\theta}_g)$ is a model of the vehicle and its drivetrain which, under consideration of the vehicle dynamics, outputs the respective torque in the drivetrain component. As outlined in [2] the torque series are split into windows of length and shift $s_{w,l} = s_{w,s} = 5000\,\mathrm{m}$ to asses the component load. For each window i the gear damage sum $d_{G,i}$ as it could be used for gear dimensioning according to [5] is calculated. As we only investigate the influence of the load generation on the damage sum $d_{G,i}$ and do not consider a particular component and its properties the damage sum is normalized with the mean damage of all windows $\bar{d}_G$ and the window length $s_{w,l}$

$$d'_{G,i} = \frac{d_{G,i}}{s_{w,s}\bar{d}_G}. \tag{3}$$

Further, due to the exponential form of the damage sum we consider its logarithm as the QoI q_i, i.e.,

$$q_i = \log d'_{G,i}. \tag{4}$$

As the focus is on the uncertainty in the velocity simulation we obtain a measured sample of the QoI by applying the system simulation to measured velocities instead of using a measured torque series directly. This way the reference damage $q_{i,ref}$ for each window is obtained from the measured velocities on the given set of routes. If there were a second set of velocity measurements from the same driver on the same set of routes the damage per window would change as a result of random changes in driving behavior, traffic, etc. It is thus necessary to model the damage per window as a random variable with associated Probability Density Function (PDF)

$$Q_i \sim p_{Q_i}(q_i) \tag{5}$$

and, for multiple windows $\boldsymbol{Q} = (Q_1, ..., Q_m)^T$, obtain the joint distribution $p_{\boldsymbol{Q}}(\boldsymbol{q})$. As the properties of the damage distribution can change in each window depending on the driving and road situation in the window, the distributions Q_i are not identically distributed and the damage distributions between windows are assumed as independent. Using the stochastic velocity simulation

$\mathcal{M}_v(s, \boldsymbol{r}(s), \boldsymbol{\theta}_v)$ in combination with the deterministic torque simulation $\mathcal{M}_g(v(s), \boldsymbol{p}(s), \boldsymbol{\theta}_g)$ the damage distribution $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$ can be estimated as a function of the velocity simulation parameters $\boldsymbol{\theta}_v = (\theta_{v,1}, ..., \theta_{v,k})^T$. The parameters of the torque simulation $\boldsymbol{\theta}_g$ are assumed to be known exactly and are not considered. The measured sample $\boldsymbol{q}_{ref} = (q_{1,ref}, ..., q_{m,ref})^T$ is drawn from the true, unknown distribution $\tilde{p}_{\boldsymbol{Q}}(\boldsymbol{q})$. The velocity simulation parameters $\boldsymbol{\theta}_v$ are defined as a random variable $\boldsymbol{\Theta}_v$ in a Bayesian interpretation of probability, meaning that we aim to find the posterior distribution $p_{\boldsymbol{\Theta}_v}(\boldsymbol{\theta}_v|\boldsymbol{q}_{ref})$ using the sample $\boldsymbol{q}_{ref}$ such that the true damage distribution $\tilde{p}_{\boldsymbol{Q}}(\boldsymbol{q})$ is well approximated and the uncertainty in the simulated distribution $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$ is expressed through the uncertain parameters $\boldsymbol{\Theta}_v$.

For the uncertain velocity simulation parameters a uniform, uninformative prior

$$p_{\boldsymbol{\Theta}_v}(\boldsymbol{\theta}_v) \sim \mathcal{U}(\boldsymbol{\theta}_{v,l}, \boldsymbol{\theta}_{v,u}) \tag{6}$$

is defined. The lower $\boldsymbol{\theta}_{v,l}$ and upper $\boldsymbol{\theta}_{v,u}$ bounds are set to threshold values estimated from a database with velocity simulation parameters from numerous drivers to ensure a physically plausible meaning for them. Table 1 lists all parameters in the vector $\boldsymbol{\theta}_v$ and their bounds. Based on the prior a Design of Experiment (DoE) is performed using Latin Hypercube Sampling (LHS) [6]. The DoE then provides the data needed to fit a stochastic surrogate model, see Section 3.

Parameters $\boldsymbol{\theta}_v$	$\boldsymbol{\theta}_{v,l}$	$\boldsymbol{\theta}_{v,u}$	Description
$\bar{a}_{x,max,dr}$	1	5	Mean of maximum driver acceleration in $\mathrm{m/s^2}$
$\bar{a}_{x,min,dr}$	-5	-1	Mean of maximum driver deceleration in $\mathrm{m/s^2}$
$o_{v,ci}$	0.5	1.3	Velocity offset on city roads
$o_{v,co}$	0.5	1.3	Velocity offset on country roads
$o_{v,mo}$	0.5	1.3	Velocity offset on motorways
$\bar{\sigma}_{\zeta,log,ci}$	-5	-2	Mean log standard deviation of velocity fluctuations on city roads
$\bar{\sigma}_{\zeta,log,co}$	-6	-2	Mean log standard deviation of velocity fluctuations on country roads
$\bar{\sigma}_{\zeta,log,mo}$	-7	-3	Mean log standard deviation of velocity fluctuations on motorways
$\bar{v}_{max,dr}$	110	150	Mean maximum driver speed in $\mathrm{km/h}$

Table 1: Velocity simulation parameters $\boldsymbol{\theta}_v$ and their bounds

Based on the simulated damage distribution per window $p_{\boldsymbol{Q_i}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$ we define the likelihood of the measurement per window i by

$$\mathcal{L}_{q_{i,ref}}(\boldsymbol{\theta}_v) = p_{Q_i}(q_{i,ref}|\boldsymbol{\theta}_v). \tag{7}$$

Due to the independence of the windows, the likelihood of all windows being their product

$$\mathcal{L}_{\boldsymbol{q}_{ref}}(\boldsymbol{\theta}_v) = \prod_{i=1}^{m} p_{Q_i}(q_{i,ref}|\boldsymbol{\theta}_v), \tag{8}$$

and the log-likelihood $\ell_{\boldsymbol{q}_{ref}}(\boldsymbol{\theta}_v)$ follows in the same manner as the sum of their logarithms.

Through Bayes' rule, the posterior distribution of the velocity simulation parameters is obtained

$$p_{\boldsymbol{\Theta}_v}(\boldsymbol{\theta}_v|\boldsymbol{q}_{ref}) \propto \mathcal{L}_{\boldsymbol{q}_{ref}}(\boldsymbol{\theta}_v) p_{\boldsymbol{\Theta}_v}(\boldsymbol{\theta}_v). \tag{9}$$

Evaluation of the damage distribution under consideration of the posterior gives the posterior predictive distribution of the damage per window given by

$$p_{Q_i}(q_i|\boldsymbol{q}_{ref}) = \int_{\boldsymbol{\theta_{v,l}}}^{\boldsymbol{\theta_{v,u}}} p_{Q_i}(q_i|\boldsymbol{\theta}_v) p_{\boldsymbol{\Theta}_v}(\boldsymbol{\theta}_v|\boldsymbol{q}_{ref}) d\boldsymbol{\theta}_v. \tag{10}$$

3 Stochastic surrogate model

A surrogate model approximates a simulation model to reduce costly evaluations of the full simulation. As the used velocity simulation in combination with the torque simulation and the damage evaluation is a stochastic simulator this requires a stochastic surrogate model which allows for faster evaluations of the damage distribution $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$ by the approximate surrogate distribution $\hat{p}_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$. Different stochastic surrogate models are described in [3, 7, 8]. For our problem, we choose the GLaM [3] which is built on the GLD [9].

A GLD can well approximate most unimodal distributions and has four parameters $\{\lambda_l : l = 1, ..., 4\}$. The lambda parameters are given by a function of the parameters $\boldsymbol{\theta}_v$ and thus the GLaM gives the approximate QoI distribution per window as a function of $\boldsymbol{\theta}_v$

$$\hat{Q}_i(\boldsymbol{\theta}_v) \sim \hat{p}_{Q_i}(q_i|\boldsymbol{\theta}_v) = \mathrm{GLD}(\lambda_1(\boldsymbol{\theta}_v), \lambda_2(\boldsymbol{\theta}_v), \lambda_3(\boldsymbol{\theta}_v), \lambda_4(\boldsymbol{\theta}_v)). \tag{11}$$

Moreover as the distributions per window are assumed as independent they can be combined to the joint distribution

$$\hat{p}_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v) = \prod_{i=1}^{m} \hat{p}_{Q_i}(q_i|\boldsymbol{\theta}_v). \tag{12}$$

Each lambda parameter and its dependence on $\boldsymbol{\theta}_v$ is modeled with a Polynomial Chaos Expansion (PCE)

$$\begin{aligned} \lambda_l(\boldsymbol{\theta}_v) &\approx \lambda_l^{PC}(\boldsymbol{\theta}_v; \boldsymbol{c}) = \sum_{\boldsymbol{\alpha} \in \boldsymbol{\mathcal{A}}_l} c_{l,\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{\theta}_v), \quad l = 1, 3, 4 \\ \lambda_2(\boldsymbol{\theta}_v) &\approx \lambda_2^{PC}(\boldsymbol{\theta}_v; \boldsymbol{c}) = \exp\left(\sum_{\boldsymbol{\alpha} \in \boldsymbol{\mathcal{A}}_2} c_{2,\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{\theta}_v) \right) \end{aligned} \tag{13}$$

where $\lambda_2(\boldsymbol{\theta}_v)$ is transformed to $\log(\lambda_2(\boldsymbol{\theta}_v))$ to ensure positivity.

The model is built on an experimental design of $N = 1000$ samples without replications, i.e., for each sample from the parameter distribution $p_{\boldsymbol{\Theta}_v}(\boldsymbol{\theta}_v)$ one sample from the conditional damage distribution $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$ is drawn. Using the Maximum Likelihood Estimation (MLE) procedure described in [3], the PCE coefficients $c_{l,\boldsymbol{\alpha}}$ for the lambda parameters are fitted. The polynomial degrees for the four lambdas are $p_1 = 2, p_2 = p_3 = p_4 = 1$ with $q_1 = q_2 = q_3 = q_4 = 1$ for the full polynomial basis in the truncation sets $\{\boldsymbol{\mathcal{A}}_l : l = 1, ..., 4\}$.

For validation, a second DoE with $N = 100$ samples is done where for each parameter sample $\boldsymbol{\theta}_v$ we run 1000 repetitions to approximate $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$. Fig. 2 compares how the distributions

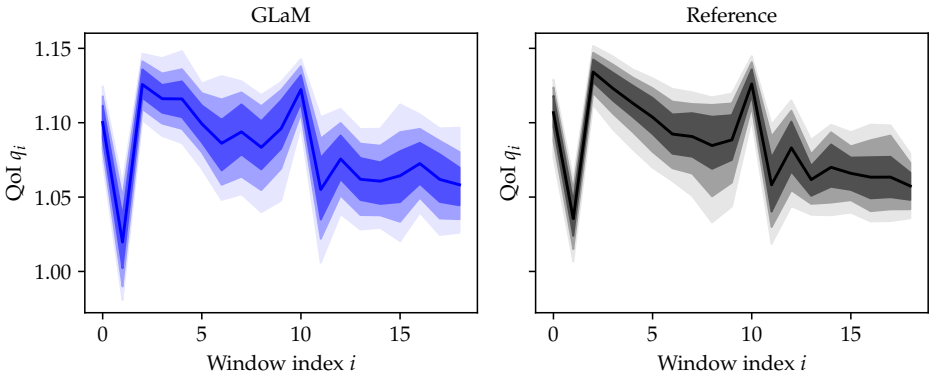


Figure 2: Comparison of reference $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$ and GLaM $\hat{p}_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$ distributions for different windows along a route with fixed parameter vector $\boldsymbol{\theta}_v$. The 3 color shades indicate regions that contain $50, 75$, and 90% of the distributions.

$p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$ and $\hat{p}_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$ deviate for different windows along a route with the parameter vector $\boldsymbol{\theta}_v$ being fixed.

Between the GLaM distribution $\hat{p}_{Q_i}(q_i|\boldsymbol{\theta}_v)$ and the distribution $p_{Q_i}(q_i|\boldsymbol{\theta}_v)$ the empirical Wasserstein distance (e.g. [10]) $d_{WS}(Q_i(\boldsymbol{\theta}_v), \hat{Q}_i(\boldsymbol{\theta}_v))$ is calculated for each of the 100 parameter samples and each window i. The empirical Wasserstein distance is computed with the 1000 samples obtained from $p_{Q_i}(q_i|\boldsymbol{\theta}_v)$ through repeated sampling and an equal amount of samples drawn from $\hat{p}_{Q_i}(q_i|\boldsymbol{\theta}_v)$. Each computed Wasserstein distance is normalized by the standard deviation of the empirical distribution $\sigma(Q_i(\boldsymbol{\theta}_v))$

$$d(Q_i(\boldsymbol{\theta}_v), \hat{Q}_i(\boldsymbol{\theta}_v)) = \frac{d_{WS}(Q_i(\boldsymbol{\theta}_v), \hat{Q}_i(\boldsymbol{\theta}_v))}{\sigma(Q_i(\boldsymbol{\theta}_v))}. \tag{14}$$

The standard deviation $\sigma(Q_i(\boldsymbol{\theta}_v))$ can be seen as the Wasserstein distance between the distribution $p_{Q_i}(q_i|\boldsymbol{\theta}_v)$ and the mean $\mathbb{E}[Q_i(\boldsymbol{\theta}_v)]$, see [3] for more details. The normalized Wasserstein distance can thus be interpreted as the ratio between the Wasserstein distances $d_{WS}(Q_i(\boldsymbol{\theta}_v), \hat{Q}_i(\boldsymbol{\theta}_v))$ and $d_{WS}(Q_i(\boldsymbol{\theta}_v), \mathbb{E}[Q_i(\boldsymbol{\theta}_v)])$, i.e., how much improvement is obtained by using the stochastic surrogate distribution $\hat{p}_{Q_i}(q_i|\boldsymbol{\theta}_v)$ compared to a degenerate distribution at the true mean.

For each window the mean of the normalized Wasserstein distance for all parameter samples from $p_{\boldsymbol{\Theta}_v}(\boldsymbol{\theta}_v)$ is taken

$$\epsilon_i = \mathbb{E}[d(Q_i(\boldsymbol{\Theta}_v), \hat{Q}_i(\boldsymbol{\Theta}_v))]. \tag{15}$$

Fig. 3 depicts the distribution of the error measure ϵ_i across all windows. For most windows, an error $\epsilon_i < 0.5$ is observed where the mean error is $\bar{\epsilon} = 0.55$. When the GLaM is slightly modified by assuming normality of the distributions $\hat{p}_{Q_i}(q_i|\boldsymbol{\theta}_v)$, i.e., $\lambda_3 = \lambda_4 = 0.13$ the mean error measure increases slightly to $\bar{\epsilon} = 0.58$.

A second metric for the validation is the accuracy with which the likelihood of a measurement can be predicted with the surrogate model. To calculate the likelihood a reference sample is defined as the mean of the prior predictive distribution (see also (10) for the posterior predictive distribution)

$$\boldsymbol{q}_{ref,Pr} = \mathbb{E}[\boldsymbol{Q}(\boldsymbol{\Theta}_v)] \text{ with } p_{\boldsymbol{\Theta}_v}(\boldsymbol{\theta}_v) \sim \mathcal{U}(\boldsymbol{\theta}_{v,l}, \boldsymbol{\theta}_{v,u}). \tag{16}$$

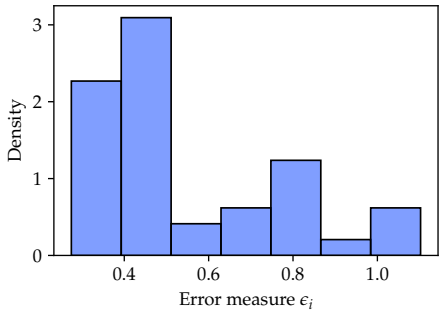
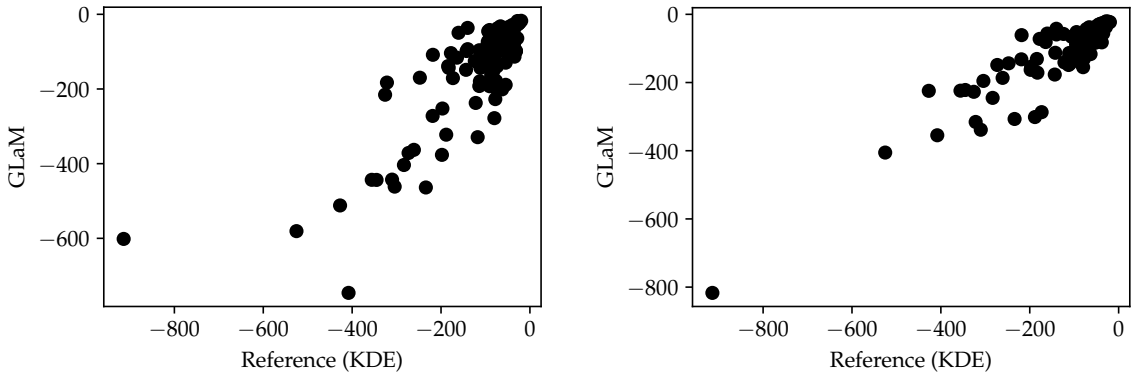


Figure 3: Distribution of error measure ϵ_i

The log-likelihood is thus calculated according to (8). To calculate the likelihood with the reference distribution a Kernel Density Estimation (KDE) is applied to the repeated samples from $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$ to approximate the distribution. Fig. 4a depicts the log-likelihood for each of the 100 parameter samples derived with the KDE approximation of $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$ and the surrogate model $\hat{p}_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$. The coefficient of determination equals $R^2 = 0.51$. When we simplify the GLaM by using the normality assumption and setting $\lambda_3 = \lambda_4 = 0.13$ the prediction is improved with $R^2 = 0.81$, see Fig. 4b. As shown in the analysis of the normalized Wasserstein distance there is only a slightly better approximation of the distribution $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$ when using the full GLD. Additionally, through the normality assumption, the model is regularized and less prone to overfitting, contributing to the improved prediction of the log-likelihood. Due to these reasons, all further analyses are carried out with the simplified GLaM which assumes a normal distribution of the surrogate $\hat{p}_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$.



(a) Full GLaM ($R^2 = 0.51$)

(b) GLaM with $\lambda_3 = \lambda_4 = 0.13$ ($R^2 = 0.81$)

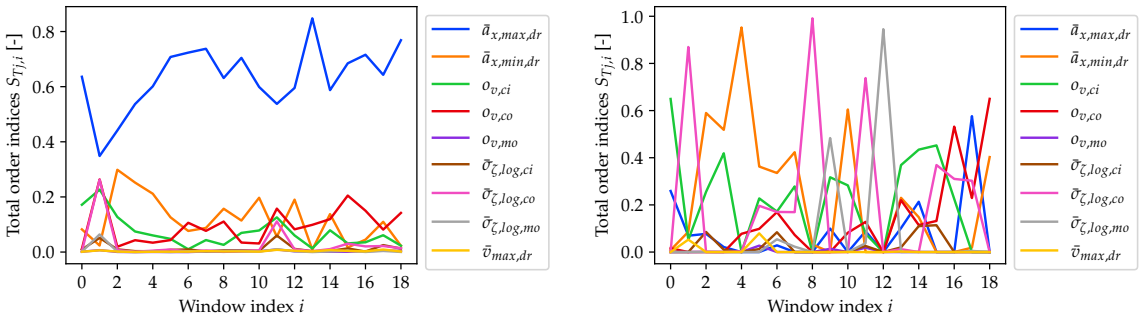
Figure 4: Comparison of log-likelihood $\ell_{\boldsymbol{q}_{ref,Pr}}(\boldsymbol{\theta}_v)$ between reference solution obtained by KDE approximation of repeated samples and the GLaM surrogate model

4 Sensitivity analysis

Based on the derived GLaM surrogate a global sensitivity analysis using Sobol' indices [4] of the input parameters is conducted. In [11] different approaches to global sensitivity analysis of stochastic simulators are described. We focus on sensitivity analysis based on deterministic QoIs derived from the damage distribution $\hat{p}_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_v)$.

For each window i the Sobol' indices of the mean $\mathbb{E}[Q_i|\boldsymbol{\theta}_v]$ and standard deviation $\sigma(Q_i|\boldsymbol{\theta}_v)$ are estimated by Monte Carlo simulations [12, 13]. Mean and standard deviation are directly obtained from the description of the GLD in the GLaM. The parameter space for the Monte Carlo simulations is defined by the prior $p_{\boldsymbol{\Theta}_v}(\boldsymbol{\theta}_v)$ from (6) where $N = 10^4$ samples are drawn. Fig. 5 depicts the total order Sobol' indices $S_{Tj,i}$ for the parameters $\theta_{v,j}$ along a route calculated for $\mathbb{E}[Q_i|\boldsymbol{\theta}_v]$ and $\sigma(Q_i|\boldsymbol{\theta}_v)$, see Tab. 1 for a description of the parameters $\theta_{v,j}$. For this route, the mean $\mathbb{E}[Q_i|\boldsymbol{\theta}_v]$ of most windows is strongly influenced by $\bar{a}_{x,max,dr}$. The standard deviation $\sigma(Q_i|\boldsymbol{\theta}_v)$ of the windows is influenced by a greater variety of parameters. To analyze the overall sensitivity of the QoIs to the parameters the mean first and total order Sobol' indices

$$\begin{aligned} \bar{S}_j &= \frac{1}{m}\sum_{i=1}^{m} S_{j,i} \\ \bar{S}_{Tj} &= \frac{1}{m}\sum_{i=1}^{m} S_{Tj,i} \end{aligned} \tag{17}$$

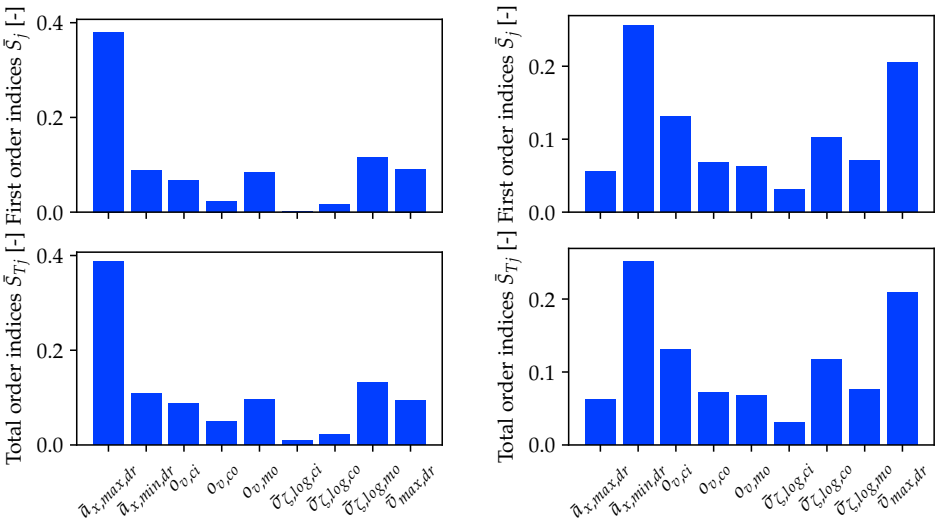


(a) $\mathbb{E}[Q_i]$ (b) $\sigma(Q_i)$

Figure 5: Total order Sobol' indices $S_{Tj,i}$ along a route for $\mathbb{E}[Q_i]$ and $\sigma(Q_i)$

are defined. Overall, the mean $\mathbb{E}[Q_i|\boldsymbol{\theta}_v]$ is mainly sensitive to $\bar{a}_{x,max,dr}$ while the standard deviation $\sigma(Q_i|\boldsymbol{\theta}_v)$ is primarily sensitive to $\bar{a}_{x,min,dr}$ and $\bar{v}_{max,dr}$, see Fig. 6. For both QoIs there is little difference between the mean first $\bar{S}_j$ and total order $\bar{S}_{Tj}$ indices, indicating no significant couplings of the velocity simulation parameters $\boldsymbol{\theta}_v$. The parameters $\bar{a}_{x,max,dr}$ and $\bar{a}_{x,min,dr}$ influence the acceleration and deceleration behavior of the driver which is strongly related to the torque in the drivetrain and thus the investigated gear damage sum. Both parameters are therefore expected to strongly influence the mean $\mathbb{E}[Q_i|\boldsymbol{\theta}_v]$ and standard deviation $\sigma(Q_i|\boldsymbol{\theta}_v)$ of the damage distribution. The parameter $\bar{v}_{max,dr}$ influences the mean speed of the driver (especially on motorways) which changes the duration and frequency of the driver accelerating and decelerating and thus the torque and gear damage sum.

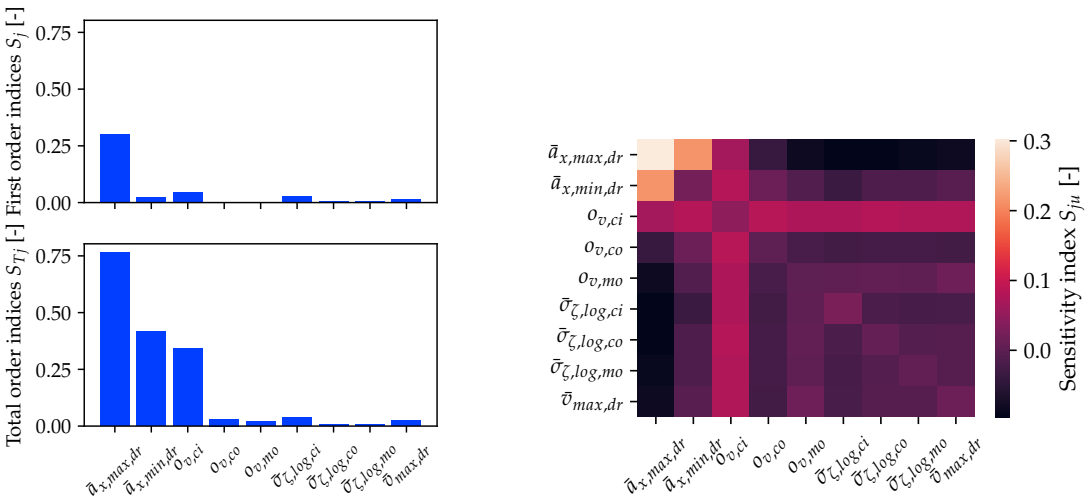
To globally analyze the importance of the parameters the sensitivity of the log-likelihood $\ell_{\boldsymbol{q}_{ref,Pr}}(\boldsymbol{\theta}_v)$ to the velocity simulation parameters $\boldsymbol{\theta}_v$ is calculated. The likelihood is calculated with the reference sample $\boldsymbol{q}_{ref,Pr}$ defined as the mean of the prior predictive distribution, see (16). The Sobol' indices thus give an indication of how sensitive the likelihood of the reference sample is to changes in the parameters. The first S_j and total order S_{Tj} Sobol' indices show a strong influence of the three parameters $\bar{a}_{x,max,dr}$, $\bar{a}_{x,min,dr}$ and $o_{v,ci}$, see Fig. 7a, the first two of



(a) $\mathbb{E}[Q_i|\boldsymbol{\theta}_v]$ (b) $\sigma(Q_i|\boldsymbol{\theta}_v)$

Figure 6: Mean first $\bar{S}_j$ and total order $\bar{S}_{Tj}$ indices for $\mathbb{E}[Q_i|\boldsymbol{\theta}_v]$ and $\sigma(Q_i|\boldsymbol{\theta}_v)$

which were also the strongest influence factors in the sensitivity analysis of mean $\mathbb{E}[Q_i|\boldsymbol{\theta}_v]$ and standard deviation $\sigma(Q_i|\boldsymbol{\theta}_v)$. The offset $o_{v,ci}$ influences the mean velocity driven in the city and has similar effects on the gear damage sum as the parameter $\bar{v}_{max,dr}$ mentioned before but with a focus on city streets. The values of the first and second-order indices are significantly different, indicating higher-order coupling effects. Fig. 7b depicts the second-order Sobol' indices S_{ju} with a strong coupling effect between $\bar{a}_{x,max,dr}$ and $\bar{a}_{x,min,dr}$.



(a) First S_j and total order S_{Tj} Sobol' indices (b) Second order Sobol' indices S_{ju}

Figure 7: Sobol' indices for log-likelihood $\ell_{\boldsymbol{q}_{ref,Pr}}(\boldsymbol{\theta}_v)$

5 Bayesian parameter inference

A Bayesian framework is applied to identify the posterior distribution $p_{\boldsymbol{\Theta}_v^*}(\boldsymbol{\theta}_v^*|\boldsymbol{q}_{ref})$ of the parameters in

$$\mathrm{A}_v = \{\bar{a}_{x,max,dr}, \bar{a}_{x,min,dr}, o_{v,ci}\} \tag{18}$$

with the most influence on the log-likelihood. These parameters $\boldsymbol{\theta}_v^* = (\theta_{v,j} : j \in \mathrm{A}_v)^T$ are modeled as a random variable $\boldsymbol{\Theta}_v^*$. The Markov Chain Monte Carlo (MCMC) algorithm with slice sampling (e.g. [14]) is employed to sample from the posterior distribution in (9), with the measured values of the QoI for each window $\boldsymbol{q}_{ref}$. As a result the joint posterior distribution $p_{\boldsymbol{\Theta}_v^*}(\boldsymbol{\theta}_v^*|\boldsymbol{q}_{ref})$ is obtained. A KDE of the distribution marginalized to one and two dimensions is depicted in Fig. 8. The black diamonds and lines indicate the default values of the parameters $\boldsymbol{\theta}_{v,0}$ identified with a heuristic from the measured velocities of the driver (e.g. $\bar{a}_{x,max,dr}$ and $\bar{a}_{x,min,dr}$), as done in [1], or taken as a guess value from experience (e.g. $o_{v,ci}$). For $\bar{a}_{x,min,dr}$ and $o_{v,ci}$ these values are within the posterior distributions. The posterior distribution of $\bar{a}_{x,max,dr}$ is bounded to the left due to the lower bound of the prior distribution $\boldsymbol{\theta}_{v,l}$ to keep the parameter values within a physically plausible range.

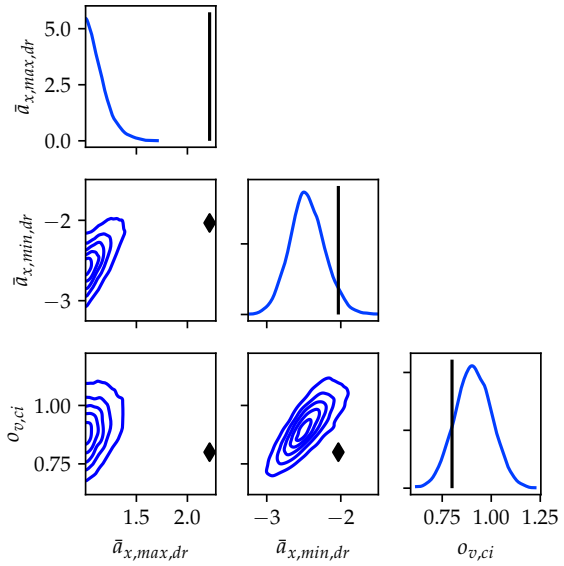


Figure 8: Joint posterior distribution $p_{\boldsymbol{\Theta}_v^*}(\boldsymbol{\theta}_v^*|\boldsymbol{q}_{ref})$ marginalized over 1 and 2 dimensions. Black diamonds and lines indicate the default values of the parameters $\boldsymbol{\theta}_{v,0}$

6 Evaluation of results

For evaluation of the identified posterior distribution $p_{\boldsymbol{\Theta}_v^*}(\boldsymbol{\theta}_v^*|\boldsymbol{q}_{ref})$ the posterior predictive $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{q}_{ref})$ distribution in (10), is evaluated and compared to the distribution obtained with the default parameter vector $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_{v,0})$. The posterior predictive $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{q}_{ref})$ is evaluated for the parameters in A_v while the other parameters are set to their default values. It should be noted that the default parameter vector $\boldsymbol{\theta}_{v,0}$ is partly also a function of the measured sample $\boldsymbol{q}_{ref}$ as some of the parameters in it are derived with heuristics from the measured velocities which are used to calculate $\boldsymbol{q}_{ref}$. Both distributions $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{q}_{ref})$ and $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_{v,0})$ are obtained by sampling $N = 1000$ times with the full simulation model.

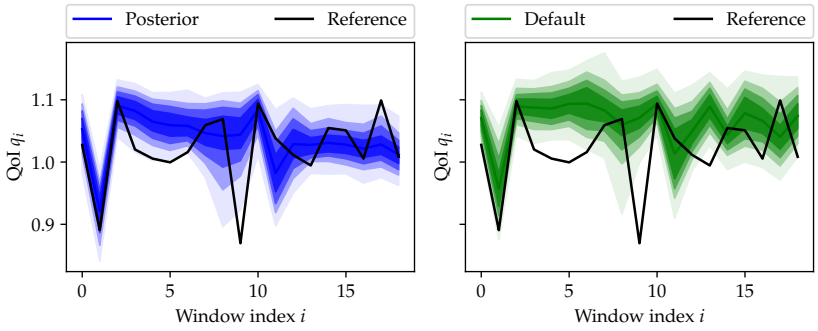


Figure 9: Comparison of posterior predictive distribution $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{q}_{ref})$ and distribution obtained with default parameters $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_{v,0})$ for multiple windows along a route. The center line is the mean of the distributions, the different color shades indicate the $50\ \%, 75\ \%, 90\ \%$, and $99\ \%$ credible intervals.

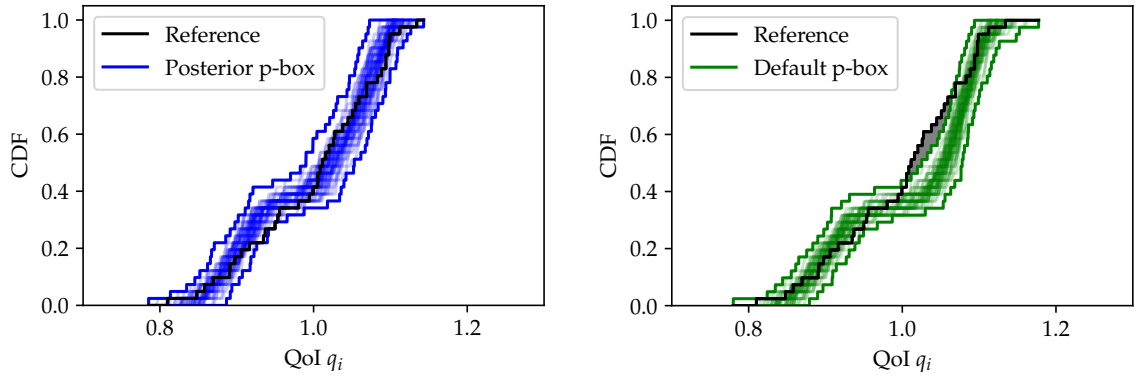
Fig. 9 depicts the $\gamma = 50\ \%, 75\ \%, 90\ \%$ and $99\ \%$ credible intervals as well as the mean of $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{q}_{ref})$ and $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_{v,0})$ compared with $\boldsymbol{q}_{ref}$ for the windows along a route. The credible intervals are defined from the quantiles $[q_{(1-\gamma)/2}, q_{(1+\gamma)/2}]$ such that γ is the probability of a value being in the interval. Overall the intervals are shifted towards the measured reference solution when using the posterior parameter distribution $p_{\boldsymbol{\Theta}_v^*}(\boldsymbol{\theta}_v^*|\boldsymbol{q}_{ref})$ compared to the default parameters $\boldsymbol{\theta}_{v,0}$. When evaluating the $99\ \%$ credible interval for all windows of all routes the posterior predictive distribution covers $83\ \%$ of all measured windows in $\boldsymbol{q}_{ref}$ compared to $76\ \%$ of the default distribution.

The density $p_{\tilde{Q}}(\tilde{q})$ describes the distribution of the QoI across multiple windows. In the considered use case this describes how often a certain load and damage sum $d_{G,i}$ occur in the considered set of routes. The available windows $i \in 1, ..., m$ are a sample from the set of all windows obtained from all routes driven over the lifespan of a vehicle. The uncertainty regarding the measured sample from the distribution $p_{\tilde{Q}}(\tilde{q})$ is thus reducible through more measurements, i.e., it is epistemic. From the simulation single samples from the window distributions $p_{Q_i}(q_i)$ for $i \in 1, ..., m$ can be combined into the distribution $p_{\tilde{Q}}(\tilde{q})$. Repeating this sampling multiple times leads to multiple empirical distributions for $p_{\tilde{Q}}(\tilde{q})$ which can be combined into a probability-box (p-box) giving a quantification regarding the epistemic uncertainty caused by the size of the route sample. The p-boxes are obtained by taking 1000 samples from all windows, i.e. by "driving" 1000 times over the given set of routes. Fig. 10 compares the p-boxes obtained from the default distribution $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_{v,0})$ and the posterior predictive distribution $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{q}_{ref})$. In the case of the posterior predictive distribution the p-box not only includes the uncertainty regarding the route set but also regarding the simulation parameters. When taking samples from $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{q}_{ref})$ the simulated p-box fully encloses the measured sample from $\tilde{p}_{\tilde{Q}}(\tilde{q})$, while for the default distribution $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_{v,0})$ there is a significant deviation to the measurement.

Moreover, Fig. 11 compares the mean

$$\bar{\tilde{q}} = \mathbb{E}[\tilde{Q}] = \frac{1}{m}\sum_{i=1}^{m} q_i \tag{19}$$

obtained from the measurement $\boldsymbol{q}_{ref}$ and the distribution $p_{\bar{\tilde{Q}}}(\bar{\tilde{q}})$ of it derived from the 1000 samples taken from $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{q}_{ref})$ and $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_{v,0})$ where each sample is the mean $\bar{\tilde{q}}$ of a single draw from $\boldsymbol{q} \sim \boldsymbol{Q}$. As q_i is the logarithm of the normalized damage sum (see (4)) the mean



(a) Obtained with $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{q}_{ref})$

(b) Obtained with $p_{\boldsymbol{Q}}(\boldsymbol{q}|\boldsymbol{\theta}_{v,0})$

Figure 10: Simulated p-boxes of the distribution $p_{\tilde{Q}}(\tilde{q})$ compared to the measured reference sample

$\bar{\tilde{q}}$ of it is linearly related to the accumulated log damage sum across all windows $d_{log,sum} = \sum_{i=1}^{m} \log(d_{G,i})$

$$\begin{aligned} \bar{\tilde{q}} &= \frac{1}{m} \sum_{i=1}^{m} \log\left(\frac{d_{G,i}}{s_{w,s}\bar{d}_G}\right) \\ \bar{\tilde{q}} &= \frac{1}{m} \left(\sum_{i=1}^{m} \log(d_{G,i}) - m(s_{w,s}\bar{d}_G)\right) \\ d_{log,sum} &= m\bar{\tilde{q}} + m(s_{w,s}\bar{d}_G) = \sum_{i=1}^{m} \log(d_{G,i}) \end{aligned} \tag{20}$$

making it especially relevant for the presented gear dimensioning use case. By using the posterior distribution of the parameters $p_{\boldsymbol{\Theta}_v^*}(\boldsymbol{\theta}_v^*|\boldsymbol{q}_{ref})$ the distribution of the mean $p_{\bar{\tilde{Q}}}(\bar{\tilde{q}})$ includes the measurement with higher probability.

7 Conclusion

In this contribution, we presented a UQ framework which allows to quantify parametric uncertainties in the virtual estimation of automotive component loads derived from simulated velocities. The framework is applied to a use case where the piecewise gear damage sum is calculated from the torque load in the drivetrain. The torque series and gear damage sum are derived from the simulated velocity on a given set of routes. It is demonstrated that the GLaM presented in [3] is suitable to build a surrogate for the stochastic velocity simulation. With the help of the GLaM a global sensitivity analysis is performed and Sobol' indices are derived for the different parameters of the velocity simulation. With the sensitivity analysis, the set of relevant parameters for the use case can be significantly reduced. Gear damage sums derived from measured velocities of a single driver are incorporated into a Bayesian framework to inversely quantify the uncertainty of the relevant velocity simulation parameters for the driver regarding the gear damage sum. When forwarding the identified parameter uncertainties through the simulation chain an improved match with the measured gear damage sums is observed taking into account the parametric uncertainties.

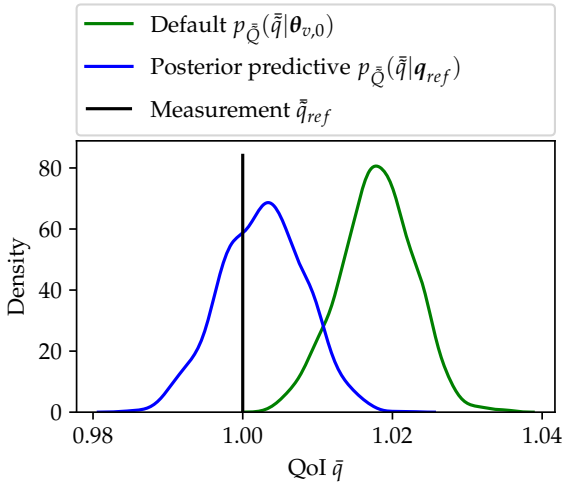


Figure 11: Mean distribution $p_{\bar{\tilde{Q}}}(\bar{\tilde{q}})$

References

[1] K. Sandmann, T. Leyendecker, M. Burger, and M. Speckert. *Ableitung von feldrelevanten Lastkollektiven mittels Stochastischer Verkehrssimulation*. 2018. URL: https://publica.fraunhofer.de/handle/publica/403187.

[2] P. Strähle, S. Wolff-Vorbeck, T. Leyendecker, and C. Proppe. "Modeling Velocity Fluctuations for the Derivation of Automotive Component Loads". In: *ECCOMAS 2024*. 9th European Congress on Computational Methods in Applied Sciences and Engineering. 2024. URL: https://www.scipedia.com/public/Strahle_et_al_2024a.

[3] Xujia Zhu and Bruno Sudret. "Emulation of Stochastic Simulators Using Generalized Lambda Models". In: *SIAM/ASA Journal on Uncertainty Quantification* 9.4 (Jan. 2021), pp. 1345–1380. ISSN: 2166-2525. DOI: 10.1137/20M1337302. URL: https://epubs.siam.org/doi/10.1137/20M1337302 (visited on 11/06/2024).

[4] I.M Sobol. "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates". In: *Mathematics and Computers in Simulation* 55.1 (Feb. 2001), pp. 271–280. ISSN: 03784754. DOI: 10.1016/S0378-4754(00)00270-6. URL: https://linkinghub.elsevier.com/retrieve/pii/S0378475400002706 (visited on 05/22/2024).

[5] ISO 6336-6. *Calculation of load capacity of spur and helical gears — Part 6: Calculation of service life under variable load*. Version Second edition 2019-11. Geneva, 2019.

[6] M. D. McKay, R. J. Beckman, and W. J. Conover. "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code". In: *Technometrics* 21.2 (May 1979), p. 239. ISSN: 00401706. DOI: 10.2307/1268522. URL: https://www.jstor.org/stable/1268522?origin=crossref (visited on 01/17/2025).

[7] Xujia Zhu and Bruno Sudret. "Stochastic polynomial chaos expansions to emulate stochastic simulators". In: *International Journal for Uncertainty Quantification* 13.2 (2023), pp. 31–52. ISSN: 2152-5080. DOI: 10.1615/Int.J.UncertaintyQuantification.2022042912. URL: https://www.dl.begellhouse.com/journals/

52034eb04b657aea,629ce0e12931a3e6,49d88de9790d7920.html (visited on 01/29/2025).

[8] Nora Lüthen, Stefano Marelli, and Bruno Sudret. "A spectral surrogate model for stochastic simulators computed from trajectory samples". In: *Computer Methods in Applied Mechanics and Engineering* 406 (Mar. 2023), p. 115875. ISSN: 00457825. DOI: 10.1016/j.cma.2022.115875. URL: https://linkinghub.elsevier.com/retrieve/pii/S0045782522008313 (visited on 01/29/2025).

[9] Marshall Freimer, Georgia Kollia, Govind S. Mudholkar, and C Thomas Lin. "A study of the generalized Tukey lambda family". In: *Communications in Statistics - Theory and Methods* 17.10 (Jan. 1988), pp. 3547–3567. ISSN: 0361-0926, 1532-415X. DOI: 10.1080/03610928808829820. URL: http://www.tandfonline.com/doi/abs/10.1080/03610928808829820 (visited on 01/27/2025).

[10] L. V. Kantorovich. "Mathematical Methods of Organizing and Planning Production". In: *Management Science* 6.4 (July 1960), pp. 366–422. ISSN: 0025-1909, 1526-5501. DOI: 10.1287/mnsc.6.4.366. URL: https://pubsonline.informs.org/doi/10.1287/mnsc.6.4.366 (visited on 10/03/2024).

[11] X. Zhu and B. Sudret. "Global sensitivity analysis for stochastic simulators based on generalized lambda surrogate models". In: *Reliability Engineering & System Safety* 214 (Oct. 2021), p. 107815. ISSN: 09518320. DOI: 10.1016/j.ress.2021.107815. arXiv: 2005.01309[stat]. URL: http://arxiv.org/abs/2005.01309 (visited on 01/20/2025).

[12] Andrea Saltelli. "Making best use of model evaluations to compute sensitivity indices". In: *Computer Physics Communications* 145.2 (May 2002), pp. 280–297. ISSN: 00104655. DOI: 10.1016/S0010-4655(02)00280-1. URL: https://linkinghub.elsevier.com/retrieve/pii/S0010465502002801 (visited on 01/24/2025).

[13] Andrea Saltelli, Paola Annoni, Ivano Azzini, Francesca Campolongo, Marco Ratto, and Stefano Tarantola. "Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index". In: *Computer Physics Communications* 181.2 (Feb. 2010), pp. 259–270. ISSN: 00104655. DOI: 10.1016/j.cpc.2009.09.018. URL: https://linkinghub.elsevier.com/retrieve/pii/S0010465509003087 (visited on 01/24/2025).

[14] Radford M. Neal. "Slice sampling". In: *The Annals of Statistics* 31.3 (June 1, 2003). ISSN: 0090-5364. DOI: 10.1214/aos/1056562461. URL: https://projecteuclid.org/journals/annals-of-statistics/volume-31/issue-3/Slice-sampling/10.1214/aos/1056562461.full (visited on 01/28/2025).

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

COMBINING NEURAL NETWORKS WITH GAUSSIAN PROCESSES FOR ROBUST AND INTERPRETABLE PREDICTION

Amalia Macali, Georgios Soimoiris, Nathan Eskue, Vahid Yaghoubi

[1]

Q-VAIbe Lab, Department of Aerospace Structures and Materials,
Faculty of Aerospace Engineering,
Delft University of Technology Kluyverweg 1, Delft, 2629 HS, the Netherlands
a.macali@tudelft.nl
g.soimoiris@tudelft.nl

Abstract. *This paper presents a hybrid model that combines Artificial Neural Networks (ANN) and Gaussian Processes (GP). The goal is to achieve high prediction accuracy while quantifying uncertainty. The proposed structure is a simple ANN used as the trend of the GP, particularly emphasizing the joint training of the two models. The ANN+GP exploits the ability of the ANN to capture complex, non-linear relationships in the data. At the same time, the GP provides an approach to uncertainty estimation, thus improving the accuracy of the predictions. This paper emphasizes the importance of concurrent training, which can improve the accuracy of the prediction model. The algorithm is suitable for any application where both accurate, robust predictions and uncertainty estimates are critical to enhance the interpretability of the model. The proposed method has been successfully applied to the frequency response functions of a simple structure.*

Keywords: Uncertainty Quantification, Artificial Neural Network, Gaussian Process

1 Introduction

Machine learning has become widely used for predictive modeling, from engineering and finance to the natural sciences. Two of the most commonly used techniques are Artificial Neural Networks (ANN) and Gaussian Processes (GP). ANNs identify complex, non-linear relationships in large datasets, making them ideal for classification, regression, and feature extraction [1]. The weakness of ANNs is that even if they are highly effective at making predictions, they do not naturally provide uncertainty estimates, a critical component in different fields like healthcare, finance, and engineering, where decision-making depends on knowing how confident one can be in a prediction [2].

While the GP is a probabilistic model, rather than providing a single point prediction, GPs furnish the underlying function as a distribution, allowing them to generate confidence intervals around their predictions [3]. This probabilistic capability is beneficial in scenarios where understanding a prediction's uncertainty is just as important as the prediction itself. However, GP scalability is challenging; in fact, the complexity increases with the size of the data, which limits their applicability to more minor data sets [4].

Given these models' complementary strengths and weaknesses, combining ANN and GP holds significant potential. Hybrid models have been proposed before, but the ANN and GP are continuously trained in different steps, optimized independently [5, 6, 7, 8]. Inspired by [9], this paper presents a combination technique in which ANN is used as the mean function of a GP and applied to two real-world spatiotemporal data sets. Their model uses stochastic variational inference and sparse GP approximations to manage scalability and uncertainty estimation. This integration brings a more effective interaction between the two models, leveraging their strengths in a unified framework.

2 Related Work

While hybrid ANN-GP models have been previously explored, few methods enable a fully unified training procedure where both components influence each other throughout optimization. Most approaches optimize ANN and GP separately or rely on variational approximations that decouple the learning process.

In contrast, the method proposed in this work integrates a simple ANN as a trend of an Exact GP. This architecture promotes tighter coupling between feature extraction and uncertainty modeling, improving calibration and prediction performance.

Deep Gaussian Processes (DGPs) enhance model capacity by increasing the depth within the GP framework, stacking multiple GP layers to capture hierarchical uncertainty [11]. Other approaches such as Gaussian Process Regression Networks (GPRNs) [12] aim to model complex, multi-output relationships by combining GPs with latent neural network structures. However, these methods introduce considerable inference complexity and treat the GP as the central component. On the other hand, several methods focus on introducing uncertainty directly into neural networks. Bayesian Neural Networks (BNNs) [13] adopt a probabilistic treatment of the network weights, while Monte Carlo Dropout [14] estimates uncertainty by performing stochastic forward passes at test time. These techniques maintain the neural architecture as the core model but add uncertainty through approximate inference mechanisms.

In contrast, the method proposed in this work avoids sequential or independent optimization by enabling joint training of both components, ANN and GP. This unified structure improves generalization, ensures stable convergence, and provides well-calibrated uncertainty estimates.

By optimizing both algorithms simultaneously, the generalization ability of the ANN is en-

hanced while providing robust uncertainty quantification through the GP. This unified optimization improves both predictive accuracy and uncertainty estimation. The results are obtained using frequency response prediction (FRF) data for mechanical systems [10].

The proposed method is compared with baseline models, including a standalone Gaussian Process and a standalone Artificial Neural Network. Although inspired by this architecture [9], the implementation adopted in this work is deliberately simplified, relying on an exact GP and avoiding variational inference or sparse approximations.

3 Methodology

3.1 GP with Neural Network Mean Function (ANN+GP)

This configuration uses the neural network as a parametric mean function within a Gaussian Process. The GP models the residual structure around the predictions made by the ANN, allowing for a probabilistic treatment of the output. An ANN captures the underlying data trend, while the GP accounts for local variations and provides uncertainty estimates. However, unlike their implementation, which relies on sparse approximations and variational inference for scalability, an exact GP model was chosen in the following experiments. A Gaussian Process is defined as:

$$f(x) \sim \mathcal{GP}(m(x), k(x, x')) \tag{1}$$

where $m(x)$ is the mean function and $k(x, x')$ is the covariance function.

In this configuration, the mean function $m(x)$ is modeled by a neural network:

$$m(x) = NN(x) = W_2\, \sigma(W_1 x + b_1) + b_2 \tag{2}$$

where $\sigma(\cdot)$ denotes a non-linear activation function, such as ReLU.

The covariance function is defined using a Matérn kernel:

$$k_\nu(r) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu} r}{\ell} \right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu} r}{\ell} \right) \tag{3}$$

where $r = \|x - x'\|$, ℓ is the lengthscale, and K_ν is the modified Bessel function of the second kind.

The resulting model combines the expressiveness of the neural network with the probabilistic nature of the GP:

$$f(x) \sim \mathcal{GP}(NN(x), k(x, x')) \tag{4}$$

This formulation allows the GP to model uncertainty around a learned trend. This work implements the model using an exact GP, avoiding variational approximations or sparse methods.

4 Results

4.1 Structural Dynamics Dataset

The structural dynamics application is based on a two-degree-of-freedom (2-DOF) mechanical system composed of two masses connected by linear springs and dampers. A harmonic excitation is applied to the first mass, and the system response is measured at both degrees of freedom, resulting in two complex-valued output channels. Uncertainty is introduced in the stiffness parameter, modeled as $k = k_0(1 + \delta_k \xi)$, where $\delta_k = 5\%$ and $\xi \sim \mathcal{N}(0, 1)$.

FRFs are computed over a frequency range of 10–35 Hz with a resolution of 0.01 Hz, resulting in 2501 frequency points per output for 100 samples. The resulting dataset has dimensions

$2 \times 2501 \times 100$. Each complex-valued output is decomposed into real and imaginary components, which are then normalized using a MinMax scaler.

Principal Component Analysis (PCA) is applied separately to the real and imaginary parts, retaining 99% of the variance. This reduces the dimensionality to 18 components for the real part and 19 for the imaginary part, maintaining computational efficiency while preserving relevant information.

4.2 Neural Architecture and Gaussian Process Configuration

To ensure a fair comparison between models, the same neural architecture is used for both the standalone ANN and the hybrid ANN+GP models. The architecture consists of a feedforward neural network with two hidden layers with 8 and 16 neurons, respectively, each followed by a ReLU activation function.

For the ANN+GP model, this neural network serves as the mean function of the Gaussian Process. The GP component uses a Multitask Matern kernel with $\nu = 2.5$, allowing it to jointly model all output components while capturing smooth variations in the signal. For the standalone GP, the same kernel and likelihood structure are used. In both cases, the model is trained using the exact GP formulation with a multitask Gaussian likelihood.

This architectural alignment ensures that any performance differences can be attributed to the modeling strategy (pure GP, pure ANN, or hybrid) rather than differences in network capacity or training procedure.

4.3 Results

The performance of the proposed ANN+GP architecture is evaluated against two baseline models: a standalone Artificial Neural Network (ANN) and a standalone Gaussian Process (GP).

The evaluation includes the following metrics: Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), coefficient of determination (R^2), and execution time.

It is important to highlight that the R^2 values reported in this work are directly obtained from the training pipeline, as computed within the PCA-transformed space (i.e., the reduced latent representation). In contrast, MSE, RMSE, and MAE are computed after reconstructing the full FRF signal from PCA, and thus reflect prediction accuracy on the original, full-resolution dataset.

Figure 1 shows a comparison of MSE, RMSE, and MAE values across the two datasets. The ANN+GP model achieves a test MSE of approximately 11.2 (real) and 11.6 (imag), clearly outperforming the ANN baseline (23.6 real, 24.1 imag) and closely matching the performance of the standalone GP model (11.5 real, 11.0 imag).

These results show that the GP model achieves the best overall performance regarding both error metrics. However, the ANN+GP model closely matches the GP regarding MSE, RMSE, and MAE on the reconstructed signal while significantly outperforming the standalone ANN across all metrics. Although not as expressive as GP in the latent space, ANN+GP still provides consistent predictions with lower error and better generalization than ANN.

Figure 2 presents the R^2 scores obtained on the test sets. The GP model achieves the highest values with $R^2 = 0.61$ (real) and $R^2 = 0.59$ (imag), followed by ANN+GP with $R^2 = 0.48$ (real) and $R^2 = 0.35$ (imag). The ANN model performs worst, with $R^2 = 0.20$ (real) and $R^2 = 0.11$ (imag).

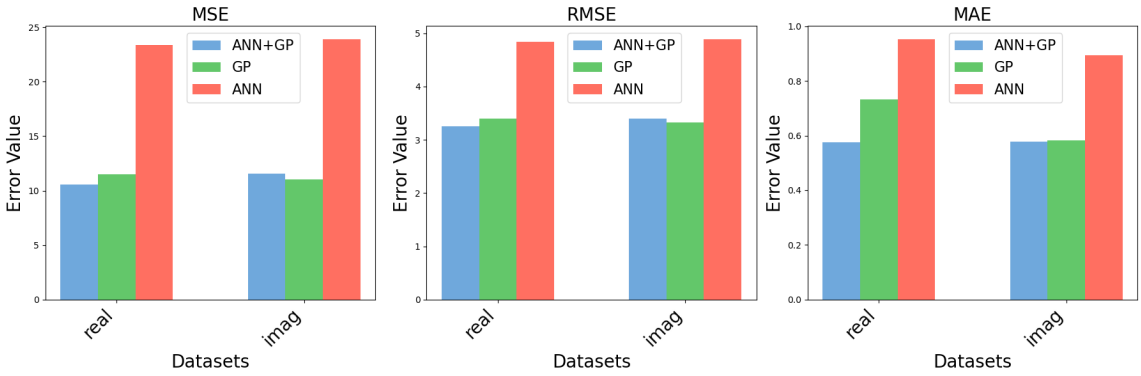


Figure 1: Comparison of MSE, RMSE, and MAE values for each model on real and imaginary components of the FRF.

This suggests that GP best captures the latent dynamics in the reduced space, while ANN+GP maintains a strong balance between accuracy and robustness—significantly outperforming the ANN.

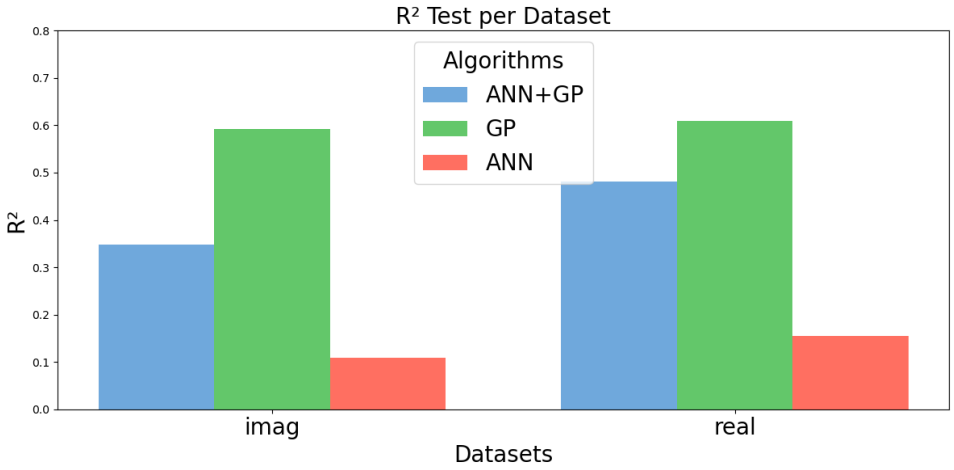


Figure 2: R^2 scores on the test sets for the real and imaginary components. GP achieves the highest latent-space performance, while ANN+GP offers a solid compromise between expressiveness and generalization.

Figures 3 and 4 provide a qualitative visualization of the reconstructed signals for the imaginary and real components, respectively. The top panel in each figure shows the true signal, followed by the predicted signals for each model.

Both ANN+GP and GP accurately capture the key dynamics of the FRF, especially around resonance peaks. The ANN+GP results appear smoother and more consistent with the ground truth, with well-calibrated uncertainty. In contrast, the ANN model struggles with signal complexity and produces unstable predictions, particularly around sharp transitions.

5 Conclusion

This work presents a hybrid modeling framework that integrates Artificial Neural Networks (ANNs) with Gaussian Processes (GPs) by using the ANN as the mean function of an exact GP. The proposed ANN+GP architecture is jointly trained and aims to combine the expressiveness of neural networks with the uncertainty quantification of GPs.

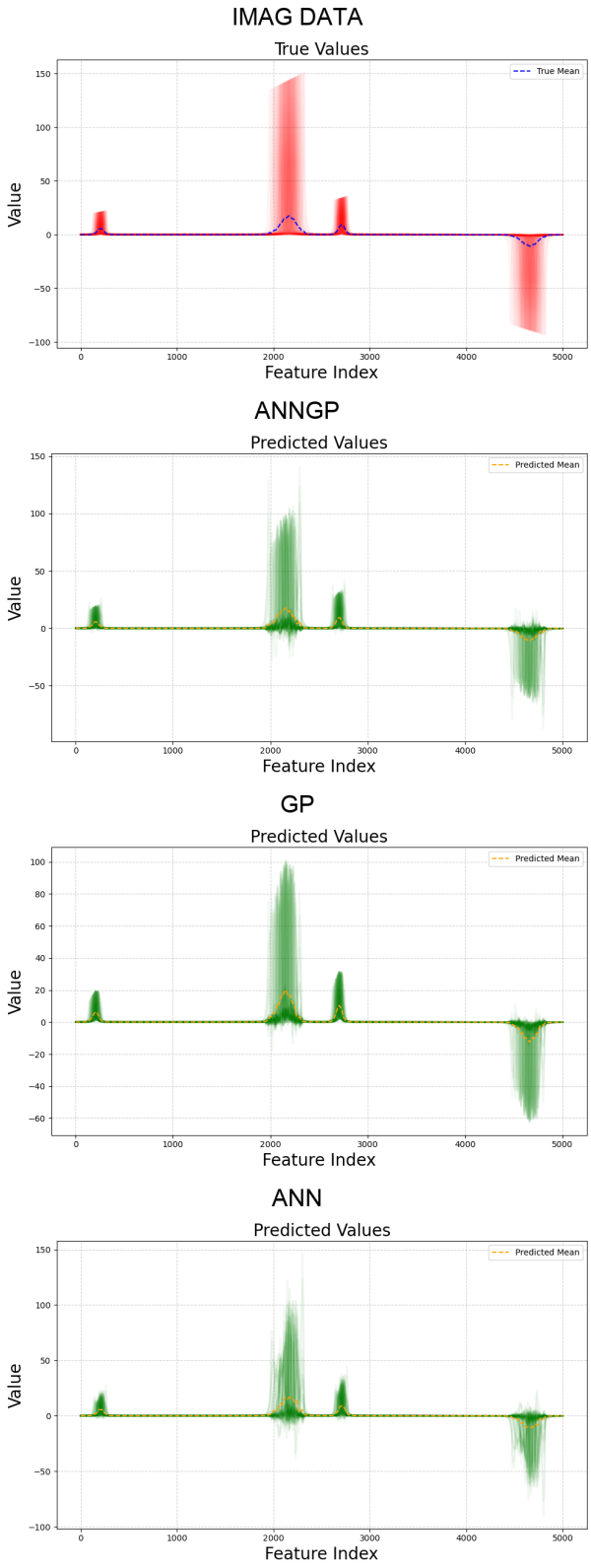


Figure 3: Predicted vs true values for the imaginary part of the FRF. ANN+GP and GP closely match the ground truth; ANN introduces artifacts and underestimates uncertainty.

The methodology was evaluated using a structural dynamics application involving predicting frequency response functions (FRFs) under parametric uncertainty. Results show that the

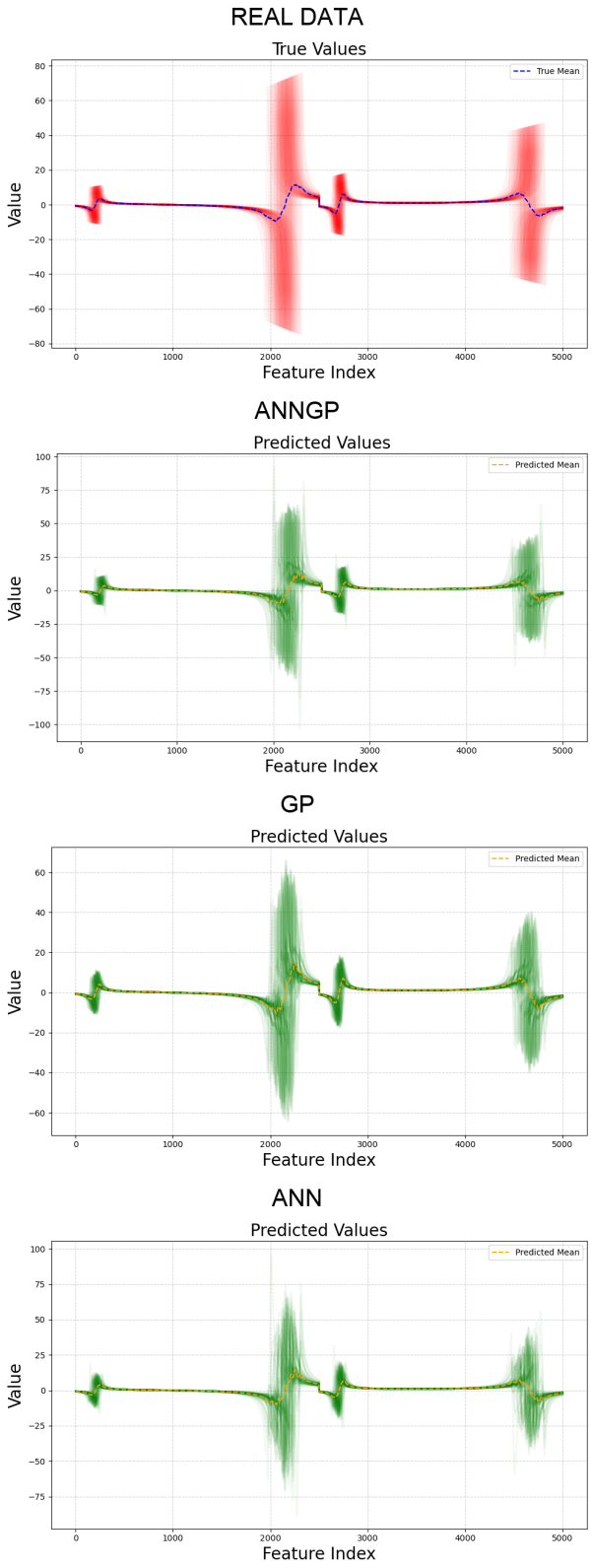


Figure 4: Predicted vs true values for the real part of the FRF. ANN+GP achieves the best visual match to the ground truth, with stable predictions and well-modeled uncertainty.

standalone GP consistently achieves the highest R^2 scores in the latent PCA space and slightly outperforms the hybrid model regarding MSE and other error metrics. However, the ANN+GP

model performs comparably to the GP while significantly improving over the standalone ANN across all evaluation metrics.

Although the ANN+GP framework has shown promising results in spatio-temporal datasets, its performance in structural dynamics remains limited. This suggests that alternative modeling strategies may be required to fully capture the underlying physics of such systems. As a potential remedy, future work will explore architectures where the Gaussian Process is used together with the neural network, allowing for more flexible uncertainty-aware predictions tailored to the dynamic behavior of structural systems.

Acknowledgements

This project is made possible in part by a contribution from the National Growth Fund program NXTGEN HIGHTECH, which is about to be launched. It is program will invest as much as 1 billion euros until 2030 with over 330 partners, in more than 60 projects and in six essential domains. I doing so, NXTGEN HIGHTECH will make a significant contribution to the structural and sustainable economic growth in the Netherlands and offer solutions for the major societal challenges in the areas of energy transition, health, safety and food. For more information, please visit NXTGEN hightech.

REFERENCES

[1] Giulia Vilone and Luca Longo, *Explainable Artificial Intelligence: a Systematic Review*, arXiv, 2020. https://arxiv.org/abs/2006.00093

[2] Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U Rajendra Acharya, and others, *A review of uncertainty quantification in deep learning: Techniques, applications and challenges*, **Information Fusion**, 76:243–297, 2021. Elsevier.

[3] Matthias Seeger, *Gaussian Processes for Machine Learning*, **International Journal of Neural Systems**, 14(02):69–106, 2004. https://doi.org/10.1142/S0129065704001899. PMID: 15112367.

[4] Alexander C. Ogren, Berthy T. Feng, Katherine L. Bouman, and Chiara Daraio, *Gaussian process regression as a surrogate model for the computation of dispersion relations*, **Computer Methods in Applied Mechanics and Engineering**, 420:116661, 2024. Elsevier.

[5] Ruzicka, J., Koza, J., Tumpach, J., Pitra, Z., & Holena, M. (2021). Combining Gaussian Processes with Neural Networks for Active Learning in Optimization. I *IAL@ PKDD/ECML* (pp. 105–120).

[6] Gao, J., Wang, J., Xu, Z., Wang, C., & Yan, S. (2023). Multiaxial fatigue prediction and uncertainty quantification based on back propagation neural network and Gaussian process regression. *International Journal of Fatigue*, 168, 107361.

[7] Farid, M. (2022). Data-driven method for real-time prediction and uncertainty quantification of fatigue failure under stochastic loading using artificial neural networks and Gaussian process regression. *International Journal of Fatigue*, 155, 106415.

[8] Y. Ide, S. Ozaki, S. Izutsu, T. Kotoura, M. Yamashiro, and M. Kodama, "Hybrid real-time wave forecasting model combining Gaussian process regression and neural networks," *Ocean Engineering*, vol. 312, p. 119028, 2024.

[9] Iwata, T., & Ghahramani, Z. (2017). Improving output uncertainty estimation and generalization in deep learning via neural network Gaussian processes. *arXiv preprint arXiv:1707.05922*.

[10] V. Yaghoubi, S. Marelli, B. Sudret, and T. Abrahamsson, "Sparse polynomial chaos expansions of frequency response functions using stochastic frequency transformation," *Probabilistic Engineering Mechanics*, vol. 48, pp. 39–58, 2017, doi: https://doi.org/10.1016/j.probengmech.2017.04.003.

[11] Andreas Damianou and Neil D. Lawrence, *Deep Gaussian Processes*, **Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics (AISTATS)**, PMLR 31:207–215, Scottsdale, Arizona, USA, 2013.

[12] Andrew Gordon Wilson, David A. Knowles, and Zoubin Ghahramani, *Gaussian Process Regression Networks*, arXiv preprint arXiv:1110.4411, 2011. Available at: https://arxiv.org/abs/1110.4411.

[13] Igor Kononenko, *Bayesian neural networks*, **Biological Cybernetics**, 61(5):361–370, 1989. Springer.

[14] Daily Milanés-Hermosilla, Rafael Trujillo Codorniú, René López-Baracaldo, Roberto Sagaró-Zamora, Denis Delisle-Rodriguez, John Jairo Villarejo-Mayor, and José Ricardo Núñez-Álvarez, *Monte Carlo Dropout for Uncertainty Estimation and Motor Imagery Classification*, **Sensors**, 21(21):7241, 2021. MDPI.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

PHYSICS-INFORMED NEURAL NETWORKS COMBINED WITH POLYNOMIAL CHAOS EXPANSION TO PROPAGATE RANDOM FIELD PROPERTIES OF COMPOSITES

Damien Bonnet-Eymard[1], Augustin Persoons[2], Panagiotis Gavallas[3], Matthias G. R. Faes[4], George Stefanou[3] and David Moens[1]

[1] KU Leuven, Department of Mechanical Engineering,
Jan Pieter de Nayerlaan 5, 2860 Sint-Katelijne-Waver, Belgium
damien.bonnet-eymard@kuleuven.be, david.moens@kuleuven.be

[2] Université de Technologie de Troyes, Laboratory of Mechanical & Material Engineering (LASMIS),
12 rue Marie Curie- CS 42060 10010 TROYES Cedex, France
augustin.persoons@utt.fr

[3] Aristotle University of Thessaloniki, Department of Civil Engineering,
54124 Thessaloniki, Greece
pgavallas@civil.auth.gr, gstefanou@civil.auth.gr

TU Dortmund University, Chair for Reliability Engineering,
Leonhard-Euler-Strasse 5, 44227 Dortmund, Germany
matthias.faes [at] tu-dortmund [dot] de

Abstract. *Composite materials exhibit inherent structural uncertainties, particularly in the spatial distribution of their constituents, which leads to randomness in their mechanical response. If a known random variable represents the spatial distribution, a common approach to uncertainty propagation is Monte Carlo simulation. This involves computing the response for numerous samples of the random variable to approximate the output distribution. However, this method is often computationally prohibitive due to the high cost of evaluating the response for each sample. To address this challenge, surrogate models are frequently used to establish the relationship between input and output random variables. Among these, Polynomial Chaos Expansion (PCE) is a widely adopted technique, as it significantly reduces computational costs while maintaining high accuracy. However, conventional PCE frameworks are designed for problems with single output variables, making them unsuitable for predicting spatially dependent random outputs without further adaptation. This work uses a novel framework combining Physics-Informed Neural Networks (PINNs) with PCE to approximate space-dependent random outputs. This approach requires only the governing partial differential equations (PDEs) and associated boundary/initial conditions, eliminating the need for expensive forward model evaluations such as Finite Element Models. The proposed framework is applied to a composite*

Keywords: Separable Physics-Informed Neural Networks, Polynomial Chaos Expansion, Neural Chaos, Surrogate modeling

---

plate under a uniform side load, where the stiffness coefficients are treated as random fields. Our results demonstrate the successful propagation of these random properties to predict the material's uncertain mechanical response.

1 INTRODUCTION

Due to their complex microstructure, composite materials exhibit inherent spatially varying properties, which can introduce randomness in their mechanical response. This uncertainty must be quantified to ensure reliable design and analysis. By studying the microstructure, it is possible to calibrate random variables that represent the spatially varying material properties [1]. Here, we address the challenge of propagating this microscale uncertainty to the macroscopic mechanical response of the material.

If a model is available to compute the response of the material (e.g., a Finite Element Model), the most straightforward approach is to use Monte Carlo simulation, i.e., to compute the response for a large number of samples of the random variables in order to approximate the output distribution. This approach, nonetheless, is often computationally prohibitive, motivating the use of surrogate models to establish the relationship between the input and output random variables.

One of the most widely used surrogate modeling techniques is Polynomial Chaos Expansion (PCE) [2], which allows for a significant reduction in the number of model evaluations while maintaining high accuracy. The main idea is to approximate the model's output as a polynomial expansion of the input random variables.

The standard PCE framework is designed for problems with a single output variable and is not directly applicable to predict spatially dependent random outputs, like the displacement field of a composite material. A possible solution is to discretize the output field into a finite number of parameters (e.g., using Principal Component Analysis) and to use a separate PCE for each parameter. Another solution employs a PCE with spatially dependent coefficients, removing the need for discretization approximations. The challenge then becomes determining these coefficient functions, which can no longer be fitted using regression techniques as done with scalar PCE coefficients.

An interesting approach to solve this problem was proposed by Zhang et al. (2019) [3], who combined the PCE framework with Physics-Informed Neural Networks (PINNs) [4]. A neural network is used to learn the spatially dependent PCE coefficients while enforcing the governing PDEs and boundary conditions through a physics-informed loss function. It is important to note that this approach no longer requires the use of FEM simulations, as the PINN is directly used as the physics solver. In previous work [5], we replicated the results of Zhang et al. (2019) on a simple 1D Poisson equation; nonetheless, we faced difficulties scaling the method to a more complex 2D composite plate problem. The slow convergence of the PINN training, combined with the high computational cost due to the number of spatial coefficients to be learned, made the method intractable for this problem.

Indeed, for the 2D composite plate problem, approximatly 50 random variables are needed to represent the 2 spatially dependent random parameters. This leads to a total of $2 \times 50 + 2 = 102$ coefficient functions to be learned, considering only the first-order PCE. Although first-order might suffice for that problem, higher-order PCEs are often needed for a more accurate approximation. This high dimensionality is a common difficulty in PCE; techniques can be used to reduce the number of basis functions to obtain a sparse PCE expansion [6].

To reduce the computational cost using sparsity, the present work combines two recently developed techniques: Neural Chaos [7] and Separable Physics-Informed Neural Networks (SPINN) [8]. Neural Chaos is a recent framework named by analogy with Polynomial Chaos Expansion, which uses a neural network instead of polynomials to construct the expansion. The random variables networks are trained alongside the coefficient functions, allowing for both

sparse basis functions and corresponding coefficients to be learned simultaneously.

A key feature of PCE is the orthogonality of its basis functions, which allows the analytical computation of output distribution moments. This orthogonality is therefore also desirable for the neural basis functions. This is achieved here using a specific loss function that penalizes the correlation between the neural basis functions. Additionally, the SPINN architecture is used to reduce the computational cost of the training. The key idea behind SPINN is to use a separate neural network for each input dimension, which are then combined through a generalized dot product. This allows faster training and fits naturally with the random variables expansion, as detailed in Section 2.4.

Hence, the overall framework can be summarized as a combination of Separable PINNs with Neural Chaos (referred to as SPINN-NC hereafter). The SPINN-NC architecture will be detailed in the methodology section by gradually introducing each component. A brief description of the PINN and SPINN architectures will be followed by Polynomial and Neural Chaos Expansions. The combination of these two components will then be presented, leading to the SPINN-NC framework. In the second part of the paper, we first validate the SPINN-NC approach on a simple 1D Poisson equation problem, and then apply SPINN with polynomial basis (SPINN-PC) to propagate the uncertain composite material properties.

2 Methodology

2.1 (Separable) Physics-Informed Neural Networks

The concept of utilizing neural networks to solve partial differential equations (PDEs) dates back to early works [9], later gaining significant traction through the introduction of Physics-Informed Neural Networks (PINNs) by Raissi et al. [4]. PINNs offer a framework for numerically solving boundary value problems (BVPs) by employing a neural network as the function approximator, where the network's weights and biases (θ) are the parameters to be determined.

Consider a general boundary value problem for a function $u(x)$ defined on a domain Ω with boundary $\partial\Omega$:

$$\mathcal{F}_\lambda[u](x) = f(x) \quad \text{in } \Omega, \tag{1}$$

$$\mathcal{B}_\lambda[u](x) = g(x) \quad \text{on } \partial\Omega, \tag{2}$$

where x represents coordinates (potentially including time), f and g are source terms and boundary/initial conditions, and $\mathcal{F}_\lambda$ and $\mathcal{B}_\lambda$ are differential operators, possibly parameterized by λ.

PINNs approximate the solution $u(x)$ with a neural network $\hat{u}_\theta(x)$. The core idea is to integrate the governing physical laws directly into the learning process. This is achieved by defining a loss function that includes terms penalizing the extent to which $\hat{u}_\theta$ violates the PDE and the boundary conditions. These physics-based loss terms are evaluated at specific locations called collocation points, distributed within the domain Ω and on its boundary $\partial\Omega$. If measurement data $\{(x_i, u(x_i))\}$ is available (e.g., for inverse problems to find λ), a data fidelity term is also included. The overall loss function is typically a weighted sum:

$$\mathcal{L}_{total} = \underbrace{\alpha_{\text{data}} \sum_i \left(u(x_i) - \hat{u}_\theta(x_i)\right)^2}_{\mathcal{L}_{\text{data}}} + \underbrace{\alpha_{\text{PDE}} \sum_{x\in\Omega} \left(\mathcal{F}[\hat{u}_\theta](x) - f(x)\right)^2}_{\mathcal{L}_{\text{PDE}}} + \underbrace{\alpha_{\text{BC}} \sum_{x\in\partial\Omega} \left(\mathcal{B}[\hat{u}_\theta](x) - g(x)\right)^2}_{\mathcal{L}_{\text{BC}}}. \tag{3}$$

The hyperparameters α_i balance the contribution of each term [10]. The differential operators $\mathcal{F}$ and $\mathcal{B}$ applied to the network $\hat{u}_\theta$ are calculated efficiently using automatic differentiation (AD), a

standard feature in deep learning libraries. Boundary conditions can be enforced "softly" via the loss term $\mathcal{L}_{\text{BC}}$ or "hardly" by designing the network architecture to satisfy them by construction, which can sometimes improve convergence [4]. A schematic of the standard PINN framework is shown in Figure 1.

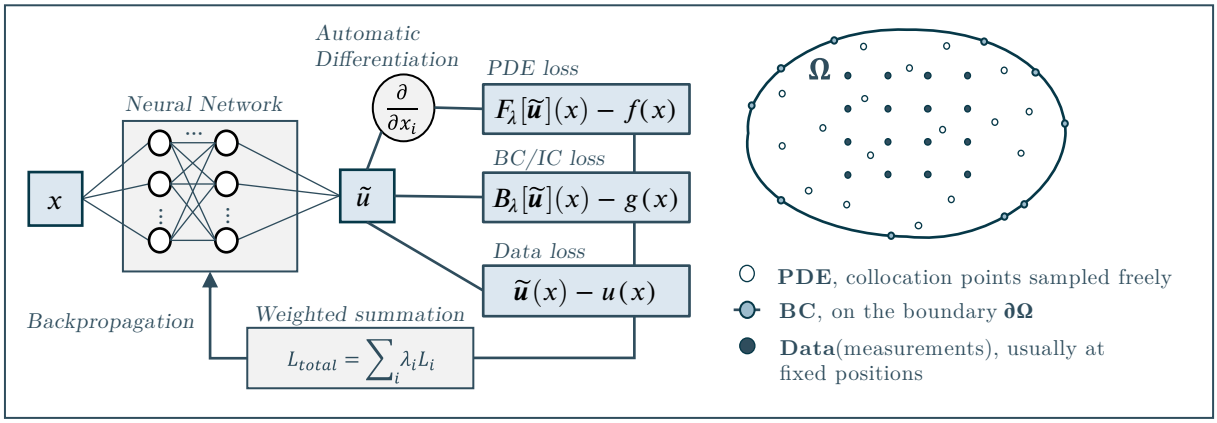


Figure 1: Physics-Informed Neural Network framework (PINN).

Training involves finding the optimal parameters θ that minimize $\mathcal{L}_{total}$, typically using gradient-based optimization algorithms. However, PINNs are known to face training challenges, and their convergence is not guaranteed, motivating ongoing research into improving their robustness and efficiency [10, 11].

To address these convergence issues, Separable Physics-Informed Neural Networks (SPINNs) were introduced by Cho et al. (2023) [8]. This architecture, illustrated in Figure 2, employs a separable structure where each component of the d-dimensional input $x = (x_1, \ldots, x_d)$ is fed into a separate "body-network". Each body-network produces an output vector of size r (the rank of the model). These d output vectors are then combined through a generalized dot product to produce the final output $\hat{u}_\theta(x)$. This structure can be interpreted as a low-rank tensor approximation of the solution, and SPINNs possess universal approximation capabilities [8].

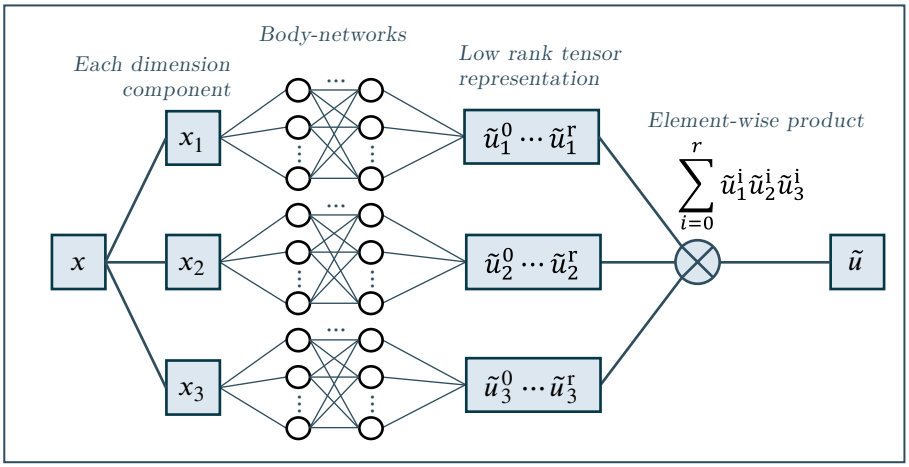


Figure 2: Separable PINN architecture for a 3D problem.

The primary advantage of SPINN lies in its computational efficiency during training. To compute the network output at N^d points arranged on a Cartesian product grid (formed by

taking N points along each of the d dimensions), only N evaluations of each of the d body-networks are required, totaling $d \times N$ evaluations. This is a significant reduction compared to the N^d evaluations needed for a standard feedforward network. This benefit, which extends to gradient computations via forward-mode AD, allows the physics loss terms to be evaluated at a much larger set of collocation points for a comparable computational cost, often leading to substantially faster convergence and improved accuracy, with reported wall-clock time reductions exceeding an order of magnitude [8].

However, this efficiency comes with the constraint that evaluation points must lie on a factorizable, grid-like structure. This can complicate applications involving complex domain geometries [8], but is not restrictive for the simple domains studied here.

Furthermore, the tensor product structure inherent in the SPINN architecture, resulting from the combination of outputs from dimension-specific networks, aligns naturally with the structure of random variables expansions.

2.2 Polynomial Chaos Expansion

Polynomial Chaos Expansion (PCE) is a popular surrogate modeling technique that has been widely used in uncertainty quantification. [12] This framework will be only briefly introduced here, the interested reader is referred to the UQLab user manual [13] for more details. The goal of PCE is to propagate uncertain input parameters through the output of a model. The main idea is to represent this ouptut as a polynomial expansion of the random input parameters. Consider a model $\mathbf{Y} = \mathcal{M}(\xi)$ where $\xi \in \mathbb{R}^M$ is a random vector with independent components and $\mathbf{Y}$ is the output of the model. The PCE surrogate model of $\mathcal{M}$ is given by

$$\mathbf{Y} \approx \mathcal{M}^{\text{PCE}}(\xi) = \sum_{\alpha \in \mathcal{A}} y_\alpha \Psi_\alpha(\xi), \tag{4}$$

where $\mathcal{A} \subset \mathbb{N}^M$ is the set of multi-indices, y_α are the PCE coefficients, and $\Psi_\alpha(\xi)$ are the multivariate polynomials. The polynomials are constructed to be orthogonal with respect to a probability distribution of the input parameters (see [13] for more details). The set of multi-indices $\mathcal{A}$ encodes the truncation of the expansion, which is a trade-off between accuracy and computational cost. Many strategies exist in the literature to optimally choose this truncation (e.g. sparse PCE [6]).

Once the PCE basis functions $\Psi_\alpha(\xi)$ are defined, calibrating the model consists of finding the PCE coefficients y_α. This step is usually data-based: the model $\mathcal{M}$ is evaluated at a set of points and the PCE coefficients are determined by regression.

Classic PCE models, as written in Eq. 4, only take random variables as input. However, in our case, the space coordinates (and eventually time) should also be considered as input. There are different ways to handle space variables in the PCE framework. The main one is to make the coefficients of the PCE model dependent on the space variables: $y_\alpha = y_\alpha(x)$

2.3 Neural Chaos

Inspired by the spectral expansion formalism of PCE, Neural Chaos (NC) was originally introduced as a data-driven alternative to classical polynomial chaos methods [7]. The main idea in Neural Chaos is to replace the pre-defined polynomial basis functions $\Psi_p(\xi)$ with basis functions parameterized by neural networks, $\Psi_p(\xi; \beta_p)$, where β_p represents the network parameters.

The expansion then takes a form analogous to that of PCE:

$$\mathbf{Y} \approx \sum_{i=0}^{r} y_i(x;\theta_i)\Psi_i(\xi;\beta_i), \tag{5}$$

with the deterministic coefficients $y_i(x;\theta_i)$ also represented by neural networks.

In the original implementation, calibration was performed solely using data, with the stochastic basis functions constructed sequentially through iterative algorithms to ensure orthogonality. In our approach, orthogonality is enforced by incorporating a loss term that compares the covariance matrix of the network outputs with the identity matrix, thereby maintaining uncorrelated learned stochastic basis functions $\Psi_i(\xi;\beta_i)$.

Furthermore, we depart from the data-based methodology by integrating Neural Chaos within a Physics Informed Neural Network (PINN) framework, directly leveraging the underlying physics.

2.4 PINN with Polynomial and Neural Chaos

The PINN-PC framework, proposed by Zhang et al. (2019) [3], uses a neural network to learn the spatially dependent Polynomial Chaos (PC) coefficients $y_\alpha(x)$. As depicted in Figure 3, the network receives the spatial coordinates x as input and outputs the set of coefficients $\{y_\alpha(x)\}_{\alpha\in\mathcal{A}}$. These coefficients then multiply the known PC basis functions $\{\Psi_\alpha(\xi)\}$ from Eq. 4 to form the overall approximation $u_\theta(x,\xi)$. Because the PCE expansion is simply a linear combination and the neural network is differentiable, the combined model remains differentiable with respect to x. Automatic differentiation can thus be used to compute the PDE and boundary condition residuals needed for the physics-informed loss terms. However, since one must sample not only over spatial points x but also over random variables $\{\xi_s\}_{s=1}^{N_\xi}$, the computational load can become prohibitive, as observed in our previous work [5].

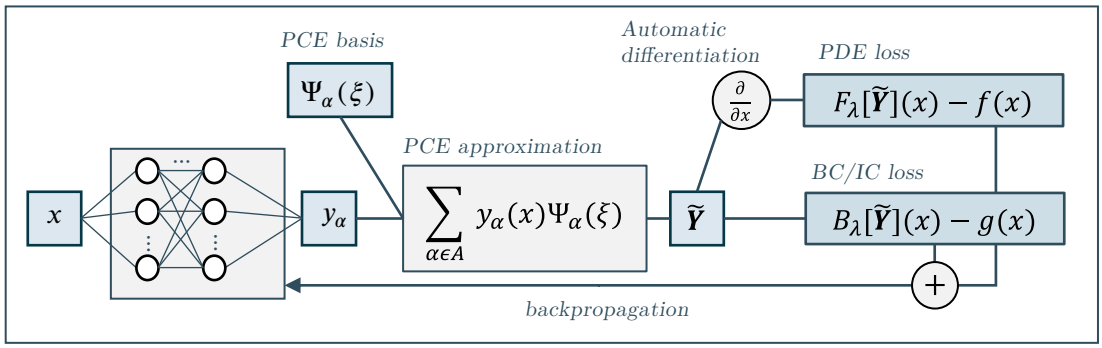


Figure 3: Physics-Informed Polynomial Chaos framework (**PINN-PC**).

To alleviate these challenges, we propose a Separable Physics-Informed Neural Chaos framework (see Figure 4). Instead of relying on fixed PC basis functions, we use Neural Chaos (NC) to learn the random basis functions $\{\Psi_i(\xi)\}_{i=0}^{r}$ directly. Simultaneously, the SPINN architecture processes spatial coordinates x to produce corresponding deterministic coefficients $\{y_p(x)\}_{i=0}^{r}$. The final approximation is formed by the dot product of these two sets of outputs:

$$\mathbf{Y} \;=\; \sum_{i=0}^{r} y_i(x)\;\Psi_i(\xi).$$

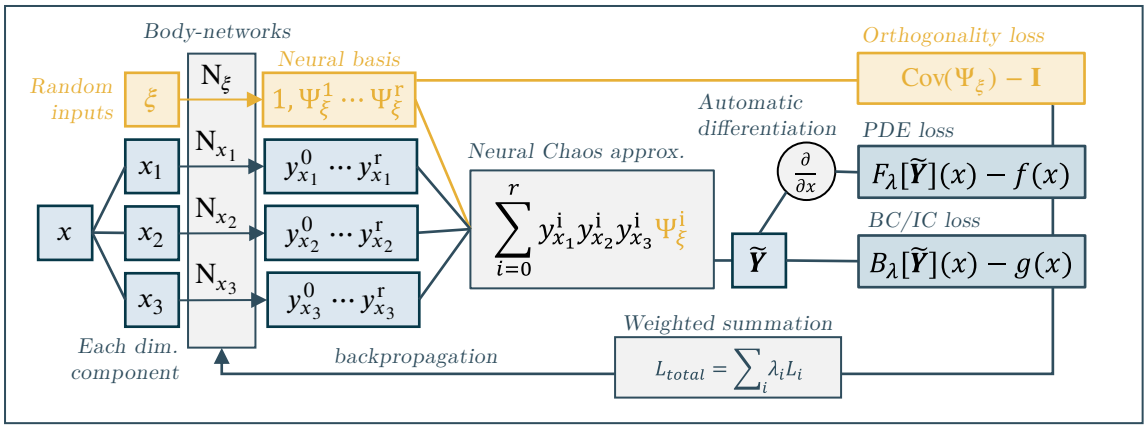


Figure 4: Separable Physics-Informed Neural Chaos framework (**SPINN-NC**).

Separable architecture. Each dimension (space or random) is handled by its own body network, improving computational efficiency when training over both spatial grids and random samples. This structure enforces separability at the level of each i-mode, potentially limiting the capacity of the model. This can be mitigate using higher-rank expansions (r).

Zero-order mode. A dedicated unit vector is included for the zero-order mode, representing the mean component of the solution. This ensures that $\Psi_0(\xi) \equiv 1$, and $y_0(x)$ captures the overall mean field.

Orthogonality loss. Orthogonality among the learned random basis functions is enforced through a loss term that compares their covariance matrix (computed over a batch of ξ_s) to the identity matrix. To ensure these basis functions remain centered (i.e., orthogonal to the mean mode), no bias is used in the random body network, and an odd activation function (e.g., $\tanh$) is selected.

By unifying Neural Chaos for the random dimension with a SPINN-based approach for the spatial dimension, we can expect significant gains in computational speed compared to traditional PINN-PC. Coefficients predicted by a SPINN can also be combined with a polynomial basis, corresponding to a model called SPINN-PC hereafter.

3 Application

In this section, we first validate the Neural Chaos approach on a simple 1D Poisson equation problem. We then address the propagation of uncertain composite plate material properties. All the simulations are performed using the open-source library `DeepXDE` [14], on a single NVIDIA GeForce RTX 4080 Ti GPU. The code will be made available on GitHub: `www.github.com/bonneted/PINN-PC`.

3.1 1D Poisson Equation

For initial validation, we reproduce the results from Zhang et al. [3] using Neural Chaos instead of Polynomial Chaos.

3.1.1 Problem Statement

We consider the one-dimensional stochastic Poisson equation with Dirichlet boundary conditions:

$$\begin{aligned} -\frac{d^2u}{dx^2} &= f(x,\xi), \quad x \in [0,1], \\ u(0) &= u(1) = 0. \end{aligned} \tag{6}$$

The source term $f(x,\xi)$ is modeled as a Gaussian random field, $\mathcal{GP}(f_0(x), k(x,x'))$, with a mean function given by

$$f_0(x) = 10\sin(2\pi x),$$

and a squared exponential covariance:

$$k(x,x') = \sigma^2 \exp\left(-\frac{(x-x')^2}{l^2}\right), \tag{7}$$

with $\sigma = 1$ and $l = 0.5$. To represent the randomness, $f(x,\xi)$ is discretized using a Karhunen-Loève expansion with $N_\xi = 6$ terms, retaining 99% of the variance.

3.1.2 Results

The SPINN-NC method is trained using $N_x = 13$ collocation points (denoted as f sensor on Figure 5 as it corresponds to points where the forcing term f is known) and $N_\xi = 10\,000$ samples of the random variables. We choose a rank $r = 3$ which is half the number of random variables. Training is carried out with the Adam optimizer at a learning rate of 10^{-3} for 5 000 iterations (approximately one minute). Orthogonality of the neural basis functions is enforced during training, ensuring that the first coefficient provided by the model represents the mean of the solution and the subsequent coefficients capture the uncertainties.

The covariance matrix, M_{cov}, of the basis functions at the end of the training is approximated using 10^5 samples of the random variables:

$$M_{cov} = \begin{bmatrix} 0.991 & 0.006 & -0.000 \\ 0.006 & 1.002 & -0.006 \\ -0.000 & -0.006 & 1.004 \end{bmatrix} \tag{8}$$

As it is close to the identity matrix, the mean and standard deviation of the solution can directly be computed from the coefficients of the Neural Chaos expansion. They are compared to the reference solution obtained from Monte Carlo simulations in Figure 5.

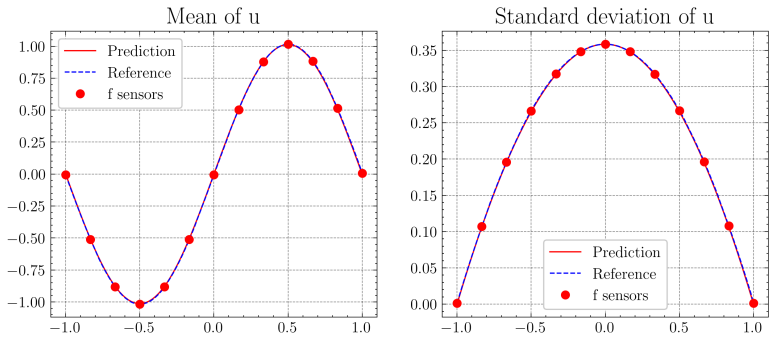


Figure 5: Mean and standard deviation of the SPINN-NC approximation compared with Monte Carlo reference solution (10^4 samples)

To further validate the accuracy of the approximation, we computed the mean squared error (MSE) between the Neural Chaos solution and a reference solution obtained using finite difference. This comparison was performed on a set of samples used during training (Train) and on a new, independently sampled set (Test) to assess potential overfitting. The density plots illustrating these MSE distributions are presented in Figure 6. The resulting mean squared error for the Neural Chaos solution was consistently below (10^{-3}) and are very similar between the training and test datasets. This indicates an accurate approximation, even with a rank lower than the number of random variables, thereby encouraging the further exploration of Neural Chaos for dimension reduction.

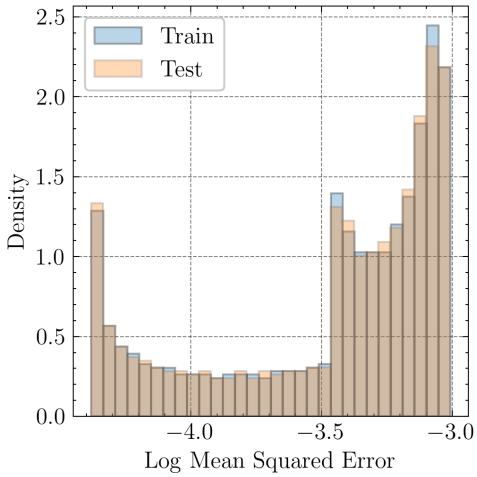


Figure 6: Density plots of the mean square error of the SPINN-NC approximation.

3.2 Composite plate

Building upon the work of Stefanou et al. (2022) [1], we investigate the propagation of random spatial variations in the stiffness matrix components of a composite plate, where these components are modeled as random fields calibrated from microstructure simulations.

3.2.1 Problem statement

We consider a 2D composite plate of dimensions 100μm $\times$ 100μm. A tensile side load $p = 0.1\text{N}/\mu\text{m}$ is applied on the right side of the plate. The boundary conditions are illustrated

in Figure 7.

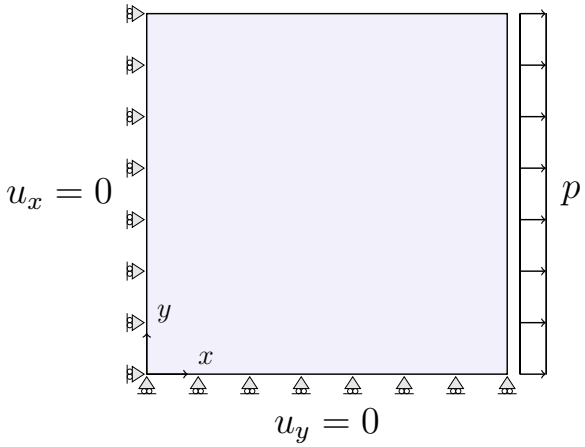


Figure 7: Boundary value problem for the composite plate.

The material is considered to be linear elastic and isotropic. The stiffness matrix can be expressed in terms of stiffness components C_{11} (axial) and C_{33} (shear) as follows:

$$\mathbf{C} = \begin{bmatrix} C_{11} & C_{11} - 2C_{33} & 0 \\ C_{11} - 2C_{33} & C_{11} & 0 \\ 0 & 0 & C_{33} \end{bmatrix}. \tag{9}$$

The stiffness components C_{11} and C_{33} are modeled as lognormal random fields with realistic mean and standard deviation calibrated from simulations of the microstructure, after Stefanou et al. (2022) [1]. To limit the number of terms in the KL expansion, we first consider larger values for the correlation length L_x and L_y (in μm). The parameters are summarized in Table 1.

Table 1: Parameters of the underlying Gaussian random fields modeling the stiffness components.

Parameter	**Mean**	**Std**	**Lx**	**Ly**
C_{11}	1.55	0.129	75	100
C_{33}	0.316	0.027	100	75

For illustration, a realization of both C_{11} and C_{33} random fields is shown in Figure 8.

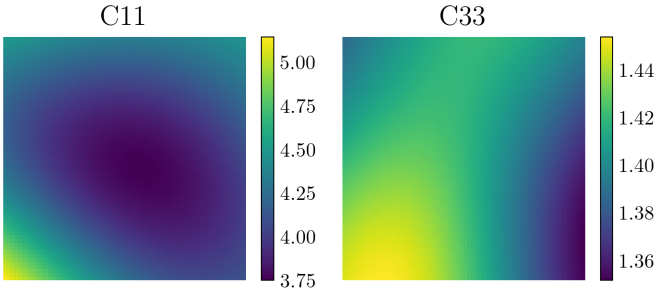


Figure 8: Realization of the random fields C_{11} and C_{33}.

Similarly to the Poisson equation example, the random fields are discretized using the Karhunen-Loève expansion. We choose here a squared exponential covariance function for the random fields to keep the number of basis functions low. The problem is discretized on a grid of $N = 100 \times 100$ points. Eigendecomposition of the covariance matrix is performed, and truncation of the expansion is chosen to keep 95% of the variance, resulting in $M = 4$ basis functions for each stiffness component.

3.2.2 Results

On this first attempt with larger correlation lengths, the number of random variables is small enough to directly use them as basis functions (no dimension reduction with Neural Chaos). The spatial coefficients are approximated using a Separable-PINN and the SPINN-PC model is trained using $N_x = 100^2$ collocation points and $N_\xi = 10$ samples of the random variables. The training is performed with the Adam optimizer at a learning rate of 10^{-3} for 50 000 iterations (approximately 5 minutes). The mean and standard deviation of the displacement field are computed by sampling the random variables (Figure 10). The results are compared to the reference solution obtained from 10^3 Monte Carlo simulations shown in Figure 9.

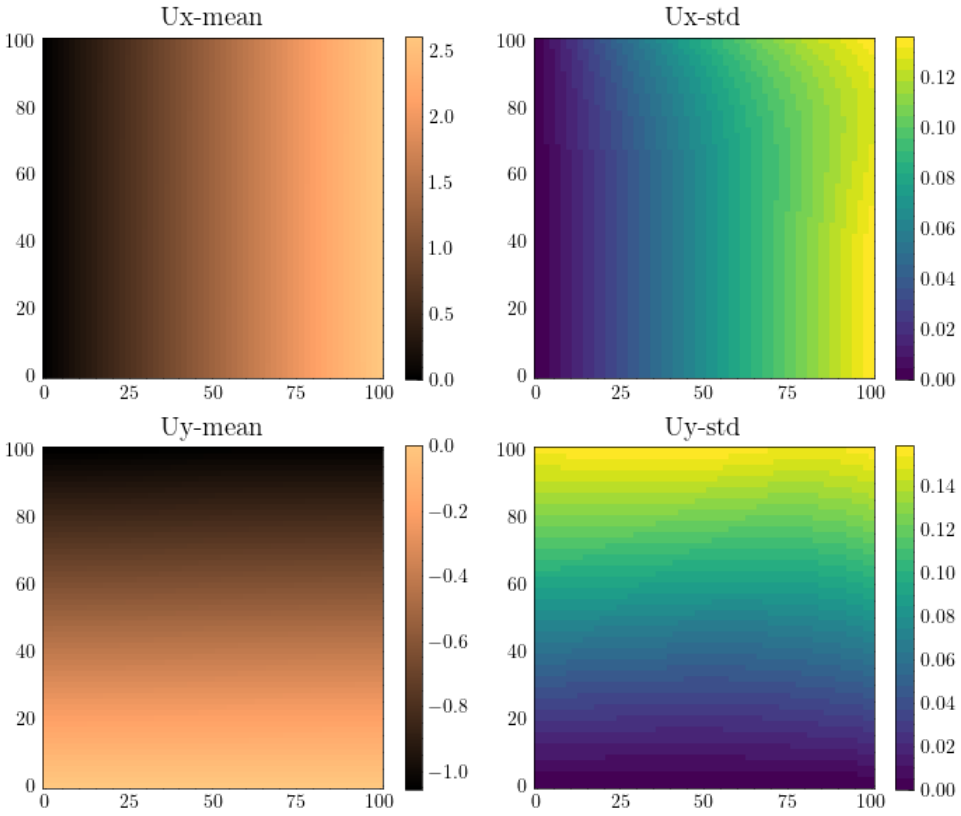


Figure 9: Mean and standard deviation of the displacement field for the composite plate problem using the 10^3 Monte Carlo samples.

The mean and standard deviation of the displacement field are found to be in good agreement with the reference solution, validating the use of Physics-Informed Neural Networks combined with Polynomial Chaos Expansion to propagate the random properties of the composite material.

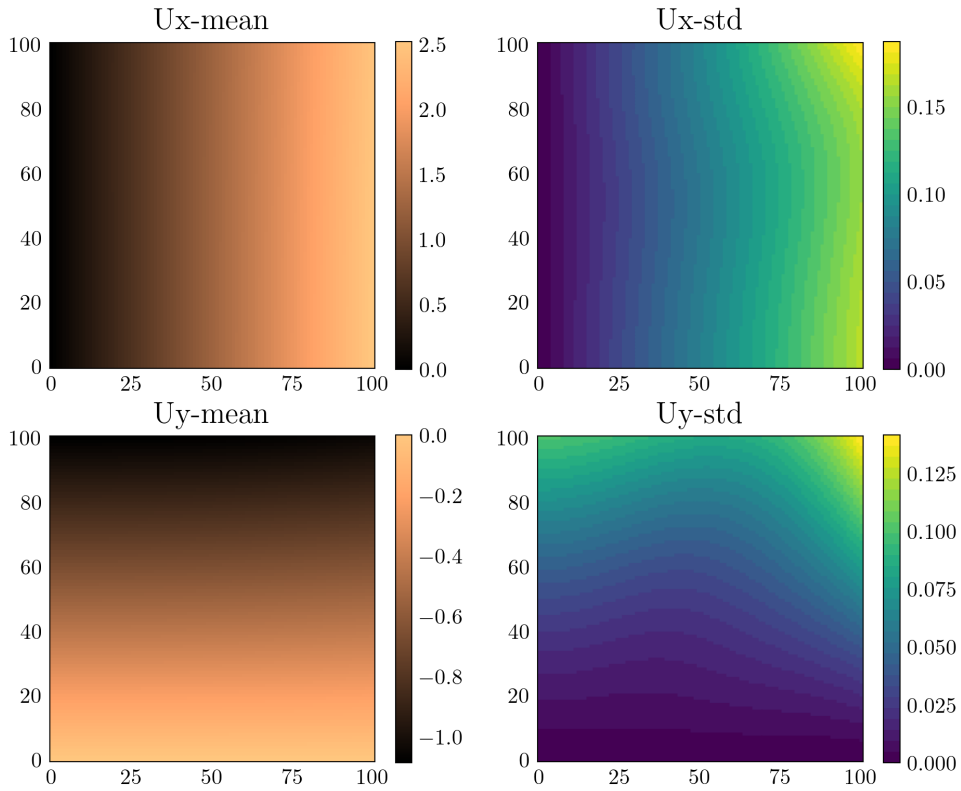


Figure 10: Mean and standard deviation of the displacement field for the composite plate problem using the SPINN-PC method.

The next step consists of using realistic correlation lengths for the random fields. This will increase the number of random variables to approximately 50, making the problem intractable with direct polynomial expansion. To address this, we intend to leverage the Neural Chaos framework for dimensionality reduction. While the initial SPINN-PC results are promising, the full implementation of SPINN-NC for this more complex problem remains under development.

4 Conclusion

In this work, we proposed a novel framework combining Separable Physics-Informed Neural Networks (SPINN) with Neural Chaos (NC) to propagate the random properties of composite materials. The SPINN-NC architecture allows for efficient training over both spatial and random dimensions, leveraging the separable structure of SPINN while learning the random basis functions directly through NC. The method was validated on a simple 1D Poisson equation problem, demonstrating its accuracy and efficiency. Subsequently, we applied a SPINN model with a polynomial basis (SPINN-PC) to a composite plate problem with larger correlation lengths, successfully validating the use of Physics-Informed Neural Networks for this class of problems. Our next step involves applying the full SPINN-NC framework to a more realistic scenario with smaller correlation lengths, leveraging the Neural Chaos framework for dimensionality reduction

REFERENCES

[1] G. Stefanou, D. Savvas, P. Gavallas, and I. Papaioannou, "The effect of random field parameter uncertainty on the response variability of composite structures," *Composites Part C: Open Access*, vol. 9, p. 100324, Oct. 2022.

[2] B. Sudret, "Polynomial chaos expansions and stochastic finite element methods," in *Risk and Reliability in Geotechnical Engineering* (J. C. Kok-Kwang Phoon, ed.), pp. 265–300, CRC Press, 2015.

[3] D. Zhang, L. Lu, L. Guo, and G. E. Karniadakis, "Quantifying total uncertainty in physics-informed neural networks for solving forward and inverse stochastic problems," *Journal of Computational Physics*, vol. 397, p. 108850, Nov. 2019.

[4] M. Raissi, P. Perdikaris, and G. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, Feb. 2019.

[5] D. Bonnet Eymard, A. Persoons, P. Gavallas, D. Moens, M. G. Faes, and G. Stefanou, "Physics-Informed Neural Networks to propagate random field properties of composite materials," in *Proceedings of ISMA2024 International Conference on Noise and Vibration Engineering USD2024 International Conference on Uncertainty in Structural Dynamics*, KU Leuven, Departement Werktuigkunde, LMSD (Mecha(tro)nic System Dynamics), Oct. 2024.

[6] G. Blatman and B. Sudret, "Adaptive sparse polynomial chaos expansion based on least angle regression," *Journal of Computational Physics*, vol. 230, pp. 2345–2367, Mar. 2011.

[7] B. Bahmani, I. G. Kevrekidis, and M. D. Shields, "Neural Chaos: A Spectral Stochastic Neural Operator," Feb. 2025. arXiv:2502.11835 [cs].

[8] J. Cho, S. Nam, H. Yang, S.-B. Yun, Y. Hong, and E. Park, "Separable Physics-Informed Neural Networks," Oct. 2023. arXiv:2306.15969 [cs].

[9] I. Lagaris, A. Likas, and D. Fotiadis, "Artificial neural networks for solving ordinary and partial differential equations," *IEEE Transactions on Neural Networks*, vol. 9, pp. 987–1000, Sept. 1998. Conference Name: IEEE Transactions on Neural Networks.

[10] S. Wang, S. Sankaran, H. Wang, and P. Perdikaris, *An Expert's Guide to Training Physics-informed Neural Networks*. Aug. 2023.

[11] S. J. Anagnostopoulos, J. D. Toscano, N. Stergiopulos, and G. E. Karniadakis, "Learning in PINNs: Phase transition, total diffusion, and generalization," Mar. 2024. arXiv:2403.18494 [cs].

[12] S. Sakamoto and R. Ghanem, "Polynomial Chaos Decomposition for the Simulation of Non-Gaussian Nonstationary Stochastic Processes," *Journal of Engineering Mechanics*, vol. 128, pp. 190–201, Feb. 2002. Publisher: American Society of Civil Engineers.

[13] S. Marelli and B. Sudret, *UQLab user manual - Polynomial chaos expansions*. July 2015.

[14] L. Lu, X. Meng, Z. Mao, and G. E. Karniadakis, "DeepXDE: A deep learning library for solving differential equations," *SIAM Review*, vol. 63, pp. 208–228, Jan. 2021. arXiv: 1907.04502.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

BRIDGE MODEL UPDATING USING MONITORED SEISMIC RESPONSES: INSIGHTS FROM A NUMERICAL PARAMETRIC STUDY

Pooria Mesbahi[1], Enrique García-Macías[2], Marco Breccolotti[1], and Filippo Ubertini[1]

[1]Department of Civil and Environmental Engineering, University of Perugia (UNIPG), Perugia, Italy
Via G. Duranti, 93, Perugia, 06125, Italy
e-mail: {pooria.mesbahi,marco.breccolotti,filippo.ubertini}@unipg.it

[2] Department of Structural Mechanics and Hydraulic Engineering, University of Granada (UGR), Granada, Spain
Campus Universitario de Fuentenueva s/n, Granada, 18071, Spain
e-mail: {enriquegm}@ugr.es

Abstract. *Model updating is a critical process in various branches of engineering, particularly in civil engineering, and plays a key role in areas such as structural health monitoring (SHM), risk assessment, decision-making, and vulnerability analysis. By integrating observed data into numerical models, engineers can refine key damage-sensitive model parameters, thereby improving the assessment of the reliability and safety of existing structures. However, many model updating methodologies, whether probabilistic or deterministic, face significant challenges. One of the main difficulties is the lack of uniqueness in the inverse calibration problem (or system identification), especially when dealing with large-scale models with various fitting parameters and limited experimental observations. Factors such as the selection of appropriate objective functions, constraints, and penalty functions in deterministic methods, and the choice of likelihoods and priors in stochastic (Bayesian) approaches, as well as decisions between using different modeling approaches such as finite element, fiber, or discrete element models, can all affect the accuracy and efficiency of the model updating process. In this research, a specific case of model updating is addressed: bridge model updating using monitored seismic responses. The core idea is that earthquake-induced responses provide a natural avenue for refining estimates of structural parameters as more seismic events are recorded. By formulating the cost function as the error between monitored and simulated responses in the time domain, the updating process can capture not only linear but also nonlinear behavior, even at intensity levels below ultimate limit states. To explore this approach further, a comparative study is conducted to evaluate the strengths and limitations of various numerical techniques. A case study involving a reinforced concrete (RC) cantilever bridge pier is presented, in which an accelerometer is deployed to update specific model parameters based on seismic response records.*

Keywords: Model updating, Bayesian model updating, Stochastic methods, Metaheuristic algorithms, Markov Chain Monte Carlo, System identification.

1 INTRODUCTION

Finite Element Model Updating (FEMU) has become a foundational tool in civil engineering to enhance the reliability of numerical simulations of structural behavior by aligning model predictions with observed responses. Applications of FEMU range from structural health monitoring (SHM) and damage detection to seismic vulnerability assessment and lifecycle reliability analysis. By calibrating uncertain parameters—such as stiffness, mass density, damping, and material properties—FEMU helps reduce parameter uncertainty and modeling errors, improve decision-making, and increase confidence in structural evaluations under both service and extreme conditions [1].

A critical challenge in FEMU is the *non-uniqueness of the inverse problem*, which refers to the fact that multiple combinations of parameter values may produce similar structural responses. This ill-posed nature is exacerbated when updating large-scale or nonlinear models with limited or noisy measurement data [1]. Ill-conditioning and identifiability issues can lead to unrealistic parameter estimates, undermining the reliability of the updated model. Several studies have attempted to address this problem by introducing regularization methods [3], model reduction techniques [2], and sensitivity-based parameter subset selection [1, 5].

According to the classification presented by Ereiz et al. [1], FEMU approaches can be broadly divided into *deterministic* and *stochastic (probabilistic)* methods. Deterministic methods treat model updating as a single-objective or multi-objective optimization problem, where the cost function—typically defined in terms of discrepancies between observed and simulated data—is minimized using optimization techniques such as gradient-based methods or metaheuristics like Genetic Algorithms (GAs) [6]. These methods are often computationally efficient but provide no insights into the uncertainty of the estimated parameters. In contrast, *stochastic methods*, rooted in Bayesian inference, aim to estimate the full posterior probability distribution of the model parameters by incorporating prior knowledge and measured data. Techniques such as Markov Chain Monte Carlo (MCMC) sampling offer rigorous uncertainty quantification but are often computationally intensive [3, 4].

Previous works in the literature have investigated ways to *improve accuracy, reduce computational cost, and quantify uncertainty* of both deterministic and probabilistic approaches. For deterministic approaches, researchers have developed hybrid optimization methods, regularization strategies, and multi-objective formulations to enhance convergence and robustness [7, 6]. Probabilistic studies, on the other hand, have employed advanced Bayesian techniques such as Transitional MCMC and approximate Bayesian comutation methods to reduce computational demand while maintaining rigorous uncertainty estimation [8, 9]. These studies also highlight how factors like data quality, parameter sensitivity, and prior distribution assumptions can affect the outcome of the updating process.

In this paper, we build upon these foundations by conducting a *comparative study of deterministic and probabilistic model updating strategies* applied to a *reinforced concrete (RC) cantilever bridge pier* subjected to seismic excitation and considering the nonlinear structural response as model evidence. Using acceleration time-history data collected at the pier's free end, we update three key structural parameters influencing the nonlinear behavior. The deterministic FEMU is performed using a *GA*, while the probabilistic update is implemented via *MCMC*, both coupled with *OpenSees* for nonlinear time history analysis (NLTHA). The general framework for model updating is implemented within the MATLAB R2024 programming environment, enabling data handling, optimization, and numerical simulation workflows. Our goal is to assess and compare the *accuracy, efficiency* of both approaches in a practical, earthquake-

excited structural scenario.

The remainder of this paper is structured as follows: Section 2 presents the theoretical formulation and general framework of the utilized methods; Section 3 describes the RC bridge pier case study and the FEMU workflow and discusses the results and key findings from both deterministic and stochastic approaches; and Section 4 closes the study with concluding remarks and future research directions.

2 Structural Dynamic Model Updating using vibration measurements

This study aims to compare the accuracy and computational efficiency of multi-variable FEMU strategies for RC structures exhibiting nonlinear behavior under seismic excitation, employing both probabilistic (stochastic) and deterministic approaches. The general objective is to update certain selected model parameters to minimize the discrepancy between the measured vibration response and the numerically simulated response of the structure under dynamic loading.

The nonlinear finite element (FE) model of the structure under investigation is governed by a collection of uncertain parameters. These include, but are not limited to, mass and inertia properties, applied gravitational loads, geometric dimensions, boundary and constraint conditions, damping characteristics, and parameters defining the nonlinear material behavior. For brevity, these will collectively be referred to as model parameters. It is assumed that selected parameters for updating are time-invariant throughout the duration of the seismic event. Two distinct model updating strategies are explored: a deterministic approach using a GA and a stochastic approach using Bayesian inference via four different algorithms.

2.1 Nonlinear FEMU with GA (deterministic approach)

In the deterministic model updating framework, the problem is formulated as an optimization task in which the optimal set of model parameters $\boldsymbol{\theta}^*$ minimizes an objective function $J(\boldsymbol{\theta}, \mathbf{x})$, defined as the mean squared error between the measured and simulated structural responses:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}, \mathbf{x}) = \arg\min_{\boldsymbol{\theta}} \mathbb{E}[\|\mathbf{y}_{\text{meas}} - \mathbf{y}(\boldsymbol{\theta}, \mathbf{x})\|^2] \tag{1}$$

where $\mathbf{x}$ denotes the non-updated parameters influencing the nonlinear structural response, such as the ground acceleration time history, initial displacements, and velocities. $\boldsymbol{\theta} \in \mathbb{R}^n$ represents the vector of model parameters to be identified (e.g., material properties such as concrete elasticity modulus, steel yield strength, or damping ratios), $\mathbf{y}_{\text{meas}} \in \mathbb{R}^m$ denotes the vector of experimentally measured structural responses (e.g., accelerations), and $\mathbf{y}(\boldsymbol{\theta}, \mathbf{x}) \in \mathbb{R}^m$ is the corresponding vector of simulated responses obtained from the FE model given the parameter set $\boldsymbol{\theta}$. The objective function $J(\boldsymbol{\theta}, \mathbf{x})$ corresponds to the expected mismatch between the observed and simulated data, with $\mathbb{E}$ in Eq. (1) representing the expectation operator.

To solve this optimization problem, a GA is employed. GA is a population-based, evolutionary metaheuristic inspired by the principles of natural selection. It does not rely on gradient information, which makes it particularly suitable for problems characterized by nonlinearity, nonconvexity, or noisy measurements, as commonly encountered in structural dynamic analyses under seismic excitation. It iteratively evolves a population of candidate solutions across successive generations to minimize a predefined fitness (objective) function. The workflow based on the GA, is illustrated schematically in the flowchart presented in Fig. 1. The deterministic FEMU procedure using GA proceeds as follows:

1. Initialize a population of N_p individuals (parameter vectors) uniformly distributed within predefined lower and upper bounds for each component of $\boldsymbol{\theta}$.

2. For each individual:

   - Conduct a nonlinear time-history dynamic analysis using the OpenSees platform.
   - Extract the acceleration response at the monitored degree of freedom(s).
   - Evaluate the objective function value $J(\boldsymbol{\theta}, \mathbf{x})$ in Eq. (1) based on the discrepancy with the corresponding measured data.

3. Apply selection, crossover, and mutation operators to generate a new population, and guide the evolutionary process toward convergence. Selection promotes the survival of fitter individuals, while crossover and mutation introduce diversity and exploration within the parameter space.

4. Repeat the evaluation and evolution steps until a convergence criterion is met (e.g., maximum number of generations or fitness stagnation threshold).

Due to the high computational demand of nonlinear dynamic simulations, particularly when evaluating large populations over many generations, parallel computing techniques are employed to accelerate the optimization process. Specifically, MATLAB's `parfor` loop structure is utilized to evaluate the fitness of individuals concurrently. Each GA population member is simulated independently via OpenSees on separate CPU cores, enabling substantial reductions in total computational time.

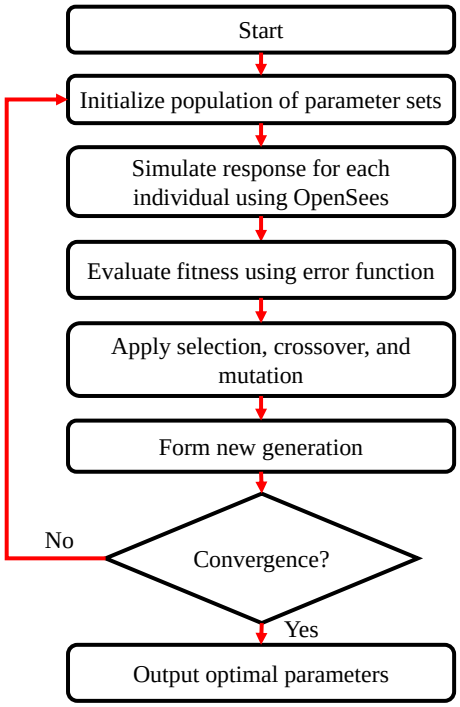


Figure 1: Deteministic FEMU with GA and OpenSees

2.2 Nonlinear FEMU with MCMC (probabilistic approach)

MCMC methods are widely used in Bayesian inference to sample from complex posterior distributions where analytical solutions are intractable. In the context of *FEMU* based on vibration measurements, MCMC provides a powerful tool for estimating the posterior distribution of uncertain model parameters $\boldsymbol{\theta}$ given observed dynamic response data $\mathbf{y}_{\text{meas}}$.

The Bayesian formulation treats the model parameters as random variables and updates their distribution based on measured data. Bayes' theorem defines the posterior distribution as:

$$p(\boldsymbol{\theta}|\mathbf{y}_{\text{meas}}) = \frac{p(\mathbf{y}_{\text{meas}}|\boldsymbol{\theta}) \cdot p(\boldsymbol{\theta})}{p(\mathbf{y}_{\text{meas}})} \tag{2}$$

where:

- $p(\boldsymbol{\theta})$ is the prior distribution reflecting prior knowledge or assumptions about the model parameters,
- $p(\mathbf{y}_{\text{meas}}|\boldsymbol{\theta})$ is the likelihood function, which measures how well the parameters $\boldsymbol{\theta}$ explain the observed vibration data,
- $p(\mathbf{y}_{\text{meas}})$ is the marginal likelihood or evidence, typically unknown and not needed in MCMC.

The posterior distribution $p(\boldsymbol{\theta}|\mathbf{y}_{\text{meas}})$ encapsulates the updated knowledge about the parameters after incorporating the measurements. MCMC methods allow for sampling from this posterior without explicitly computing the evidence. Assuming normally distributed errors in the model prediction, the likelihood is expressed as:

$$p(\mathbf{y}_{\text{meas}}|\boldsymbol{\theta}) \propto \exp\left(-\frac{1}{2}(\mathbf{y}_{\text{meas}} - \mathbf{y}(\boldsymbol{\theta}, \mathbf{x}))^{\top} \boldsymbol{\Sigma}^{-1} (\mathbf{y}_{\text{meas}} - \mathbf{y}(\boldsymbol{\theta}, \mathbf{x}))\right), \tag{3}$$

where $\mathbf{y}(\boldsymbol{\theta}, \mathbf{x})$ denotes the FE model response, and $\boldsymbol{\Sigma}$ is the covariance matrix representing the measurement noise and modeling uncertainty. The value of Σ plays a critical role in shaping the likelihood function, effectively controlling how strongly the data influence the posterior distribution. The sampling of the posterior distribution using a MCMC method enables not only parameter estimation but also the quantification of uncertainty associated with the identified parameters.

2.3 MCMC Techniques for Bayesian FEMU

This section presents the theoretical formulation and salient features of four different MCMC techniques which will be comparatively evaluated in the subsequent sections. These techniques include the classical Metropolis-Hastings (MH) algorithm [13, 14], the Adaptive Metropolis (AM) algorithm [11], the Delayed Rejection (DR) method [12], and the Delayed Rejection Adaptive Metropolis (DRAM) algorithm [15], which combines the features of DR and AM.

2.3.1 Metropolis-Hastings (MH) Algorithm

The Metropolis-Hastings (MH) algorithm (based on the work of Metropolis et al. [13] and later Hastings [14]) is the cornerstone of MCMC sampling. It generates a Markov chain whose stationary distribution converges to the target posterior distribution $p(\boldsymbol{\theta}|\mathbf{y}_{\text{meas}})$, where $\boldsymbol{\theta}$ denotes

the model parameters and $\mathbf{y}_{\text{meas}}$ the observed data. Given a current state $\boldsymbol{\theta}_k$, a new candidate $\boldsymbol{\theta}^*$ is drawn from a proposal distribution $q(\boldsymbol{\theta}^*|\boldsymbol{\theta}_k)$. The target distribution, commonly denoted as $\pi(\boldsymbol{\theta})$ in the MCMC literature, typically represents the posterior distribution of the model parameters given the observed data, i.e., $\pi(\boldsymbol{\theta}) = p(\boldsymbol{\theta} \mid \mathbf{y}_{\text{meas}})$ [18]. The acceptance probability α is computed as:

$$\alpha(\boldsymbol{\theta}_k, \boldsymbol{\theta}^*) = \min\left(1, \frac{\pi(\boldsymbol{\theta}^*)q(\boldsymbol{\theta}_k|\boldsymbol{\theta}^*)}{\pi(\boldsymbol{\theta}_k)q(\boldsymbol{\theta}^*|\boldsymbol{\theta}_k)}\right) \tag{4}$$

The candidate $\boldsymbol{\theta}^*$ is accepted with probability α, otherwise the chain remains at $\boldsymbol{\theta}_k$. The algorithm proceeds as follows:

Initialization $(i = 0)$

- Choose an initial guess $\boldsymbol{\theta}^{(0)}$
- Set Σ
- Select a proposal distribution q
- Set Priors

Iterative Sampling

- **Proposal:** Generate a candidate sample $\boldsymbol{\theta}^*$ from a Gaussian proposal distribution $q(\boldsymbol{\theta}^*|\boldsymbol{\theta}_k)$
- **Acceptance Ratio:** Compute the Metropolis-Hastings acceptance ratio in Eq. (4)
- **Acceptance Step:** Accept the new sample with probability α, i.e.,

$$\boldsymbol{\theta}^{(i+1)} = \begin{cases} \boldsymbol{\theta}^* & \text{with probability } \alpha \\ \boldsymbol{\theta}_k & \text{otherwise} \end{cases}$$

- Increment $k \leftarrow k + 1$ and repeat.

The likelihood function $p(\mathbf{y}_{\text{meas}}|\boldsymbol{\theta})$ is constructed based on the discrepancy between simulated and observed structural response (e.g., acceleration), which can be modeled using the Gaussian error model in Eq. (3).

After sufficient iterations, the Markov chain generated by the MH algorithm converges to the posterior distribution. The samples can then be used to estimate posterior statistics such as the mean, variance, credible intervals, and to perform sensitivity or uncertainty analysis. Diagnostic tools such as trace plots, acceptance rates, and autocorrelation functions are commonly used to assess convergence and mixing quality of the chain. A flowchart of MH is represented in Fig. 2

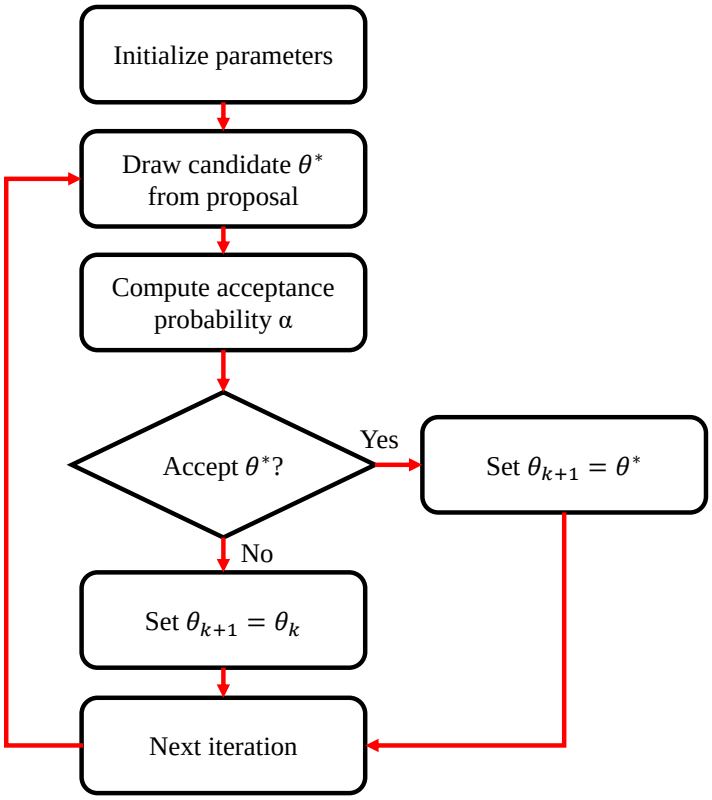


Figure 2: Metropolis-Hastings (MH) Algorithm

2.3.2 Adaptive Metropolis (AM)

Although MH is simple and robust, its performance can degrade in high-dimensional or highly correlated parameter spaces due to inefficient exploration. To alleviate some of these limitations, an Adaptive Metropolis (AM) algorithm [11] enhances the classical MH by allowing the proposal distribution to adapt during the sampling process. The proposal covariance matrix is updated online using the empirical covariance of the samples in the Markov chain. This adaptation aims to improve the efficiency and convergence of the chain by tuning the proposal to the shape of the target distribution. At iteration k, the candidate θ^* is drawn from a multivariate normal proposal distribution:

$$\boldsymbol{\theta}^* \sim \mathcal{N}(\boldsymbol{\theta}_k, s_d \Sigma_k), \tag{5}$$

where Σ_k is the empirical covariance matrix computed from $\{\theta_0, \ldots, \theta_{k-1}\}$, and s_d is a scaling parameter typically set as $s_d = 2.4^2/d$, with d being the dimension of the parameter space [17]. The adaptation is performed only after a burn-in period to ensure convergence properties.

2.3.3 Delayed Rejection (DR)

The Delayed Rejection (DR) technique [12] is an extension of MH that aims to improve acceptance rates by allowing a second (or more) proposal if the first one is rejected. Instead of remaining at $\boldsymbol{\theta}_k$ when $\boldsymbol{\theta}^*$ is rejected, a new candidate $\boldsymbol{\theta}^{**}$ is proposed using a different, usually narrower, proposal distribution. The first-stage acceptance probability is the presented in Eq. 4.

If $\boldsymbol{\theta}^*$ is rejected, the second-stage candidate $\boldsymbol{\theta}^{**}$ is sampled from $q_2(\boldsymbol{\theta}^{**}|\boldsymbol{\theta}_k, \boldsymbol{\theta}^*)$ with acceptance probability:

$$\alpha_2(\boldsymbol{\theta}_k, \boldsymbol{\theta}^*, \boldsymbol{\theta}^{**}) = \min\left(1, \frac{\pi(\boldsymbol{\theta}^{**})q_1(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{**})q_2(\boldsymbol{\theta}_k|\boldsymbol{\theta}^{**}, \boldsymbol{\theta}^*)\left[1-\alpha_1(\boldsymbol{\theta}^{**}, \boldsymbol{\theta}^*)\right]}{\pi(\boldsymbol{\theta}_k)q_1(\boldsymbol{\theta}^*|\boldsymbol{\theta}_k)q_2(\boldsymbol{\theta}^{**}|\boldsymbol{\theta}_k, \boldsymbol{\theta}^*)\left[1-\alpha_1(\boldsymbol{\theta}_k, \boldsymbol{\theta}^*)\right]}\right) \tag{6}$$

This approach can significantly increase the efficiency of the sampling process, especially in complex target distributions with narrow high-probability regions.

2.3.4 Delayed Rejection Adaptive Metropolis (DRAM)

The DRAM algorithm [15] combines the adaptivity of AM with the multiple proposal strategy of DR, aiming to achieve both fast convergence and high acceptance rates. The adaptation of the proposal distribution is performed in conjunction with the delayed rejection steps, allowing the algorithm to efficiently explore the parameter space. Key features of DRAM include:

- Online adaptation of the covariance matrix to fit the posterior distribution.
- Multiple proposal stages to avoid stagnation and improve acceptance.
- Capability to efficiently handle correlated or multimodal posteriors.

The DRAM algorithm has been shown to perform better than MH, AM, or DR individually, particularly in Bayesian FEMU problems involving high-dimensional and poorly scaled parameter spaces. A flowchart illustrating the workflow of DRAM is illustrated in Fig. 3

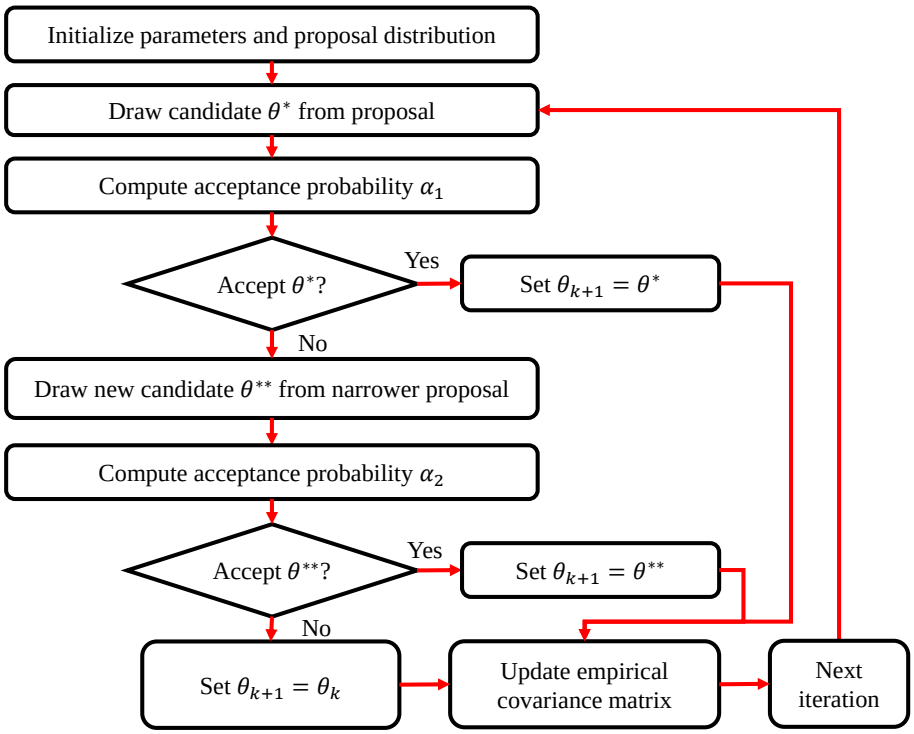


Figure 3: Delayed rejection adaptive metropolis (DRAM) algorithm.

3 Numerical example and discussion

To compare the effectiveness of the model updating approaches presented in the previous sections, a circular RC bridge pier is selected as a case study. This choice is motivated by the widespread use of such piers in bridge structures located in seismically active regions worldwide [16]. The geometric layout and material properties of the pier are illustrated in Fig. 4.

The mass contributing to the computation of inertial forces is defined by accounting for both the dead load of the superstructure (including the deck and cap beam), modeled as a lumped mass of 113,456 kg, and the self-weight of the pier, modeled as a distributed mass of 1,672 kg/m. The pier is fixed at the base, representing a cantilever configuration, and the P–Δ effect is included in the analysis to account for geometric nonlinearity.

For the dynamic excitation, an ground acceleration from the 1994 Northridge earthquake (station: Castaic - Old Ridge Route/360°, 6.4 Ml, 4:30:55 AM PST, 17 Jan 1994) [10] is applied. This ground motion is selected to ensure that the pier, when modeled with its initial (true) parameters, exhibits nonlinear plastic behavior. A NLTHA is then performed, and the resulting acceleration response at the top of the pier is considered the reference output in this analysis. To simulate realistic measurement conditions, artificial noise is added to the acceleration response, with a standard deviation corresponding to 5% of gravity. The generated time series data is considered henceforth as synthetic experimental data.

In the model updating process, three parameters are selected for calibration: the elastic modulus of concrete E_c, the yield strength of reinforcing steel f_y, and the damping ratio ζ for the first elastic mode (T1: 0.5733 sec). The assumed true values for E_c, f_y, and ζ are 2.5×10^{10} Pa, 4.6×10^8 Pa, and 5%, respectively. For both deterministic FEMU, where the bounds act as constraints, and probabilistic FEMU, where they define the limits of uniform prior distributions, the following parameter ranges are adopted: E_c varies between 2.13×10^{10} and 2.88×10^{10} Pa; f_y ranges from 4.23×10^8 to 4.97×10^8 Pa; and ζ is considered within the interval 3.60%, and 6.40%.

The hysteretic behavior of the pier, accelerogram, acceleration and displacement response (at free end) are shown in Fig. 5, and as it can be noted in this figure that the pier experienced nonlinearity during this ground motion.

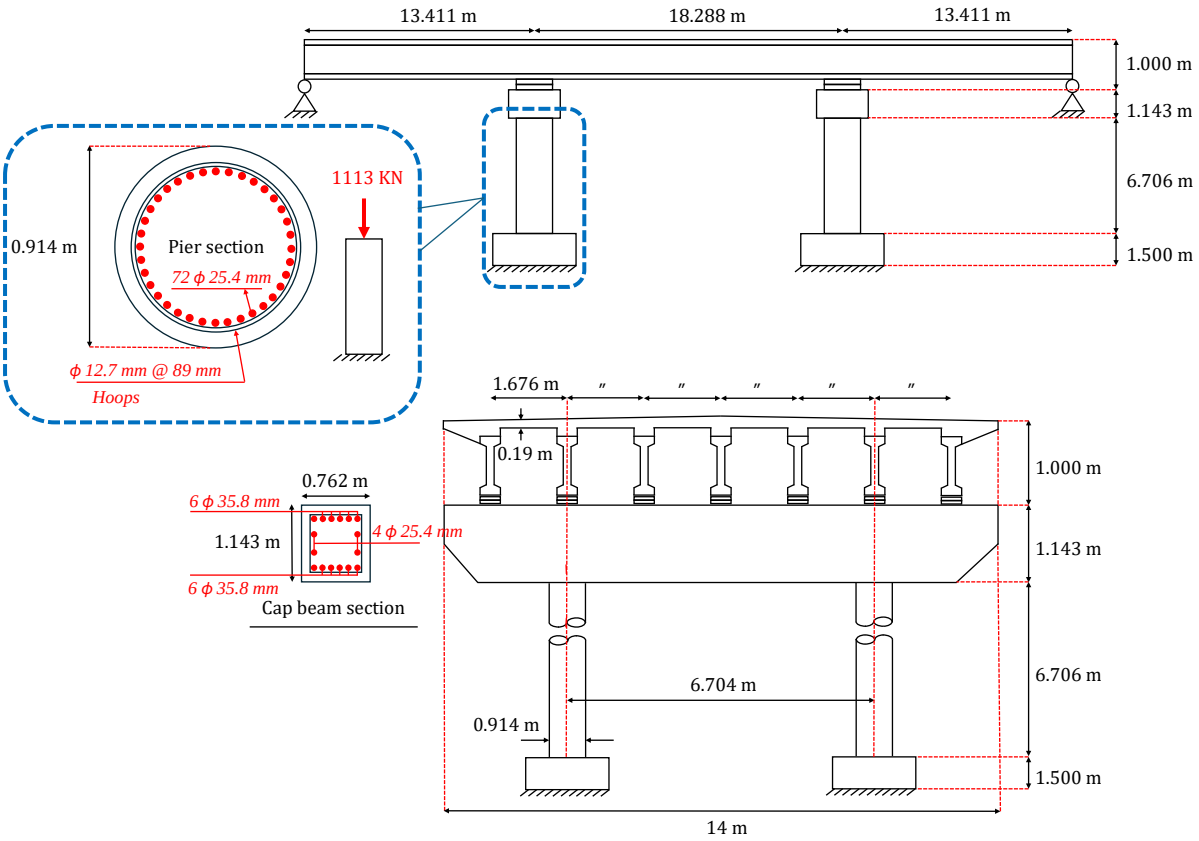


Figure 4: Considered layout of multispan continuous concrete I-girder bridge

(a) Bridge pier response in terms of base shear and displacement

(b) Accelerogram and time-history response

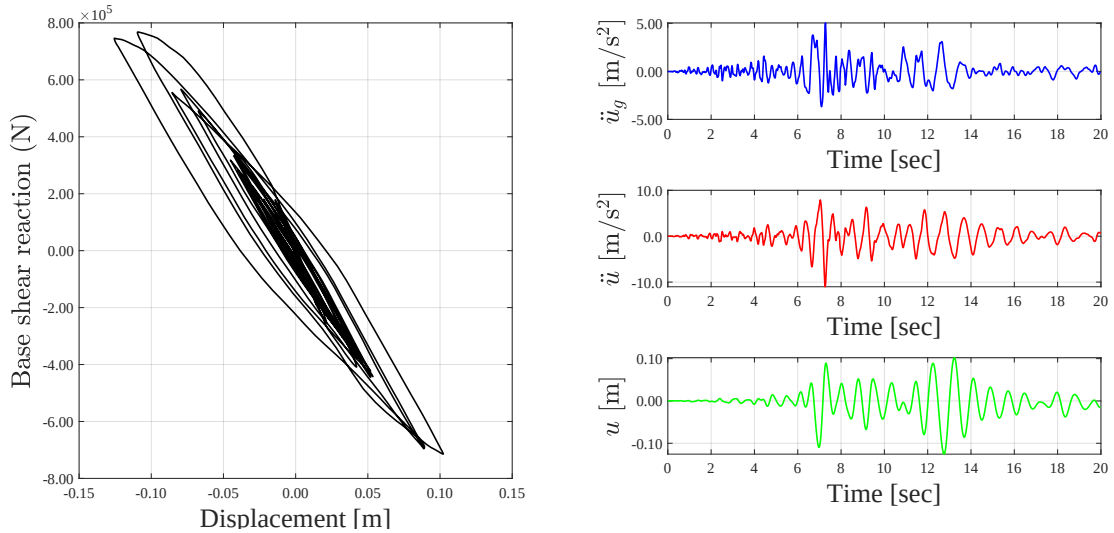


Figure 5: Bridge pier response to 1994 Northridge earthquake

3.1 Deterministic FEMU results

The results in Table 1 summarize the performance of a deterministic FEMU approach using a GA for two different population sizes, namely a population of 20 (P20) and 50 (P50). The optimal solutions obtained with the larger population size (P50) show significantly better accuracy compared to those with P20, as evidenced by the lower estimation errors across all parameters. In particular, the estimation error for ζ drops from 6.4% to just 0.4%, highlighting the sensitivity of this parameter to the population size in the optimization process. The estimation error for each model parameter is computed as the absolute difference between the estimated value and the true value, normalized by the true value and expressed as a percentage.

The objective function values in the final generation further confirm the improvement in solution quality with a larger population. Both the mean and best values of the objective function are lower for P50, indicating better convergence behavior. This suggests that increasing the population size enhances the global search ability of the GA, reducing the risk of premature convergence and allowing more accurate identification of the true model parameters.

However, this improved accuracy comes at the cost of higher computational time. The stopping criteria were set to 100 generations for P20 and 40 generations for P50. The total runtime increased from 6,974 seconds for P20 to 8,504 seconds for P50. This increase is attributed to the added computational complexity and overhead associated with evaluating a larger population in each generation of the GA. The implementation of parallel computing significantly reduces the computational time, achieving a speedup of more than 14 times for both the P20 and P50 cases as shown in Table 1. Fig. 6 presents the convergence behavior of both the mean and best values of the objective function for the deterministic approach. The results clearly indicate that the P50 case yields lower objective function values in both metrics compared to the P20 case. Furthermore, Fig. 7 illustrates the convergence of the estimated model parameters. A comparison with the true values demonstrates that, across all three parameters, the P50 case provides more accurate estimations than P20.

Table 1: Results of nonlinear FEMU with GA

	Population	
	P20	P50
Optimal solution		
E_c [Pa]	2.4451×10^{10}	2.5272×10^{10}
f_y [Pa]	4.6474×10^{8}	4.5905×10^{8}
ζ [%]	5.32	5.02
Estimation error		
E_c [%]	2.2000	1.0900
f_y [%]	1.0300	0.2070
ζ [%]	6.4000	0.4000
Objective function in last generation		
Mean value $(\mathrm{m}^2/\mathrm{s}^4)$	3.5864×10^{3}	2.8278×10^{3}
Best value $(\mathrm{m}^2/\mathrm{s}^4)$	3.3472×10^{3}	2.7668×10^{3}
Computational cost		
Time (sec)	6974	8504
Time (with parallel computation, 7 cores) (sec)	484	590

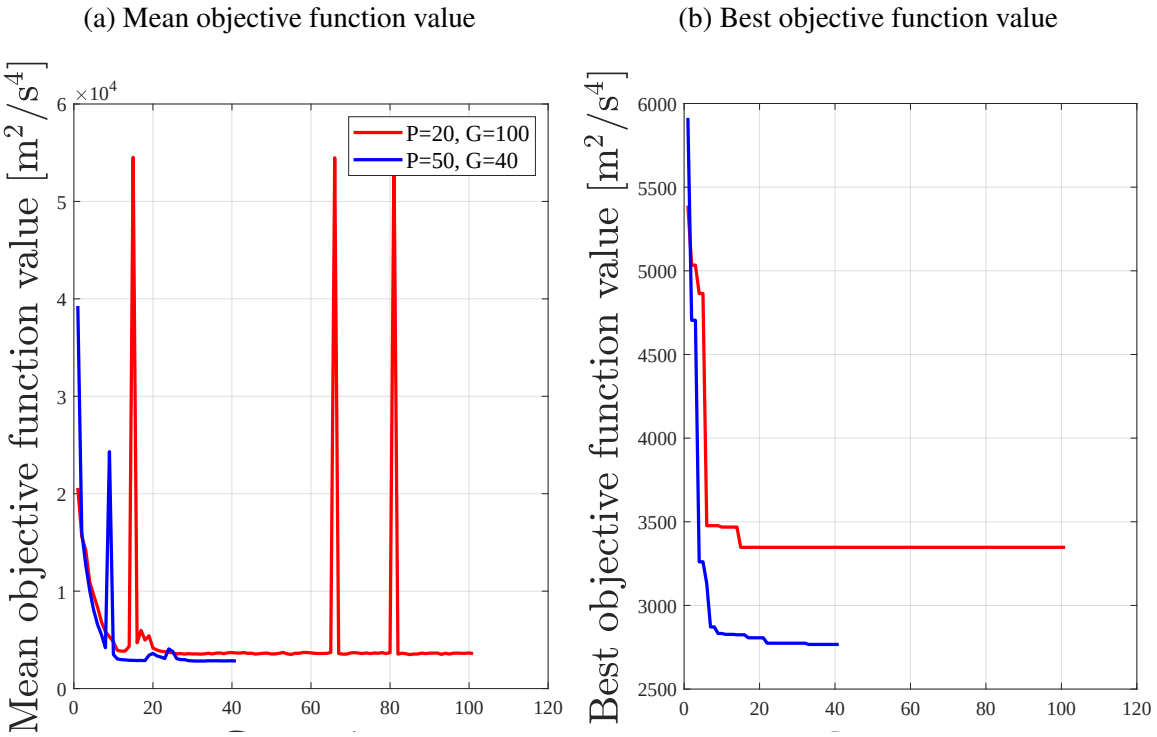


Figure 6: Objective function convergence in deterministic approach

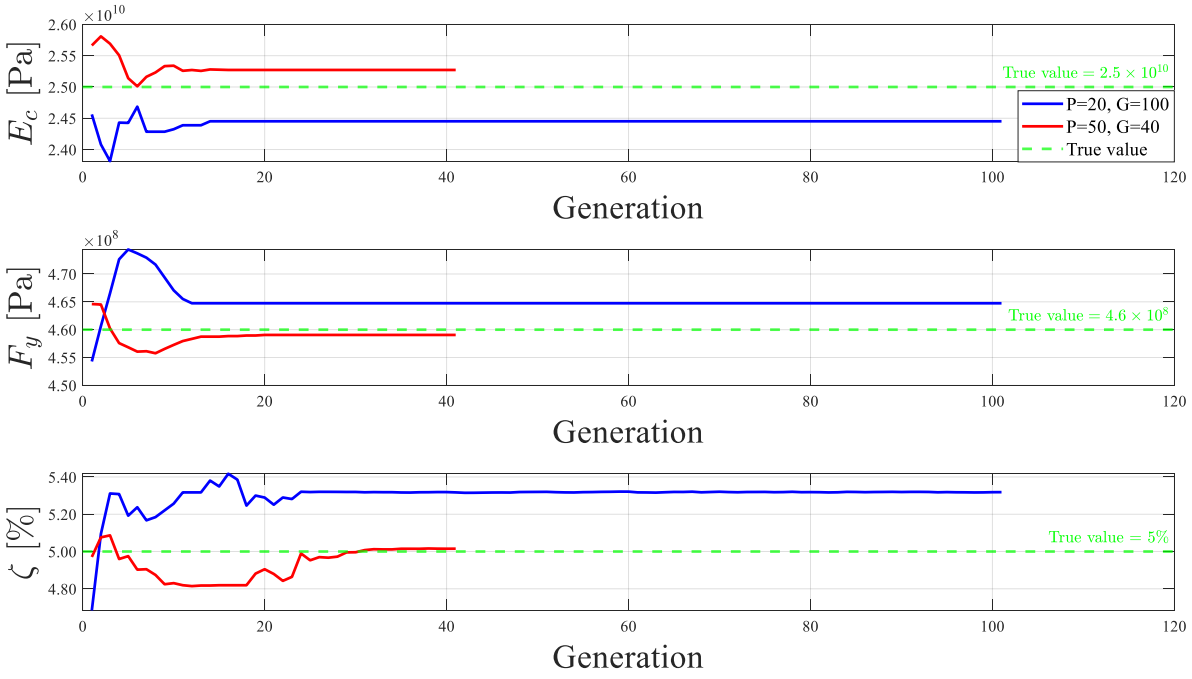


Figure 7: Convergence of model parameters in deteministic approach

3.2 Probabilistic FEMU results

The probabilistic FEMU results using MCMC algorithms (Table 2) demonstrate that all four methods – MH, DR, AM, and DRAM – produce nearly identical mean estimates for the key parameters: Young's modulus ($E_c \approx 2.49$–2.50×10^{10} Pa), steel yield stress ($f_y \approx 4.58$–4.60×10^8 Pa), and damping ratio ($\zeta \approx 0.0502$–0.0506). This consistency across algorithms indicates robust convergence in the solution space for this class of inverse problems.

However, significant differences emerge in computational efficiency, with AM being the fastest (23,460 s), MH moderately efficient (27,247 s), and both DR and DRAM requiring approximately twice the computational time of AM (46,000 s) despite employing the same number of samples (8000). These observations suggest that the adaptive mechanism in AM provides substantial computational advantages without compromising solution accuracy, while the delayed rejection strategy in DR and DRAM incurs significant additional cost.

Although in this case study the number of model parameters is limited to three—resulting in comparable accuracy among the algorithms—their performance may diverge with an increased number of parameters. In particular, algorithms specifically designed for high-dimensional or highly correlated parameter spaces are expected to yield more accurate and efficient estimations under such conditions.

Analysis of the coefficient of variation (COV) reveals consistent patterns in parameter uncertainty across all methods. Among model parameters (ζ) shows substantially greater COV (9.63–9.83%) compare to the others and yield stress exhibits the lowest COV (3.53–3.72%). The consistently higher variability in ζ estimates suggests either greater sensitivity to algorithmic differences or inherent challenges in precisely constraining this parameter through the available measurements. This pattern holds across all methods, indicating it may reflect fundamental properties of the inverse problem rather than algorithm-specific behaviors.

The results demonstrate that while different MCMC variants achieve equivalent mean solutions for deterministic FEMU problems, their computational performance and stability characteristics differ significantly. The AM algorithm emerges as particularly advantageous, combining computational efficiency with reliable convergence. The MH method offers slightly better stability for f_y estimation at moderate additional computational cost, while DR and DRAM provide comparable accuracy but require substantially greater resources. The consistently higher uncertainty in ζ estimates warrants particular attention in practical applications, suggesting that either additional measurement information or alternative parameterization approaches may be needed to better constrain this parameter. These findings provide valuable guidance for algorithm selection in FEMU applications, where the choice between methods may depend on the relative importance of computational efficiency versus parameter stability for specific engineering problems.

Table 2: Results of nonlinear FEMU with MCMC

	MCMC algorithm			
	MH	DR	AM	DRAM
Optimal solution (mean value)				
E_c [Pa]	2.49×10^{10}	2.50×10^{10}	2.50×10^{10}	2.49×10^{10}
f_y [Pa]	4.58×10^{8}	4.59×10^{8}	4.60×10^{8}	4.59×10^{8}
ζ [%]	5.02	5.06	5.06	5.04
Optimal solution (COV)				
E_c [%]	4.88	4.84	4.87	4.9
f_y [%]	3.53	3.630	3.72	3.64
ζ [%]	9.68	9.63	9.83	9.82
Computational cost				
Number of samples	8000	8000	8000	8000
Time (sec)	27247	46044	23460	45899

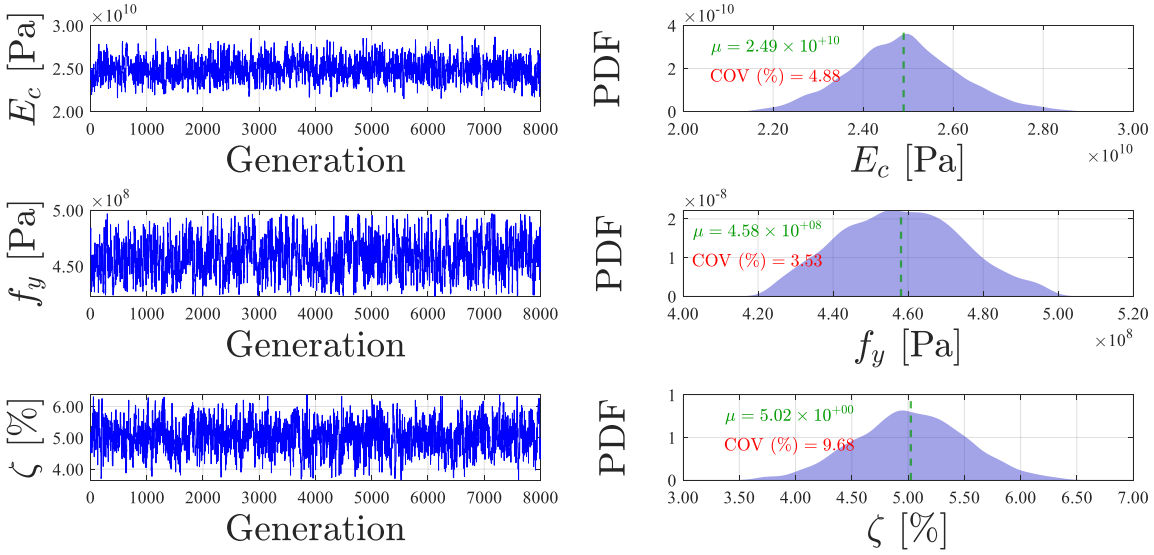


Figure 8: MCMC trace and posterioir PDF using the MH algoritm

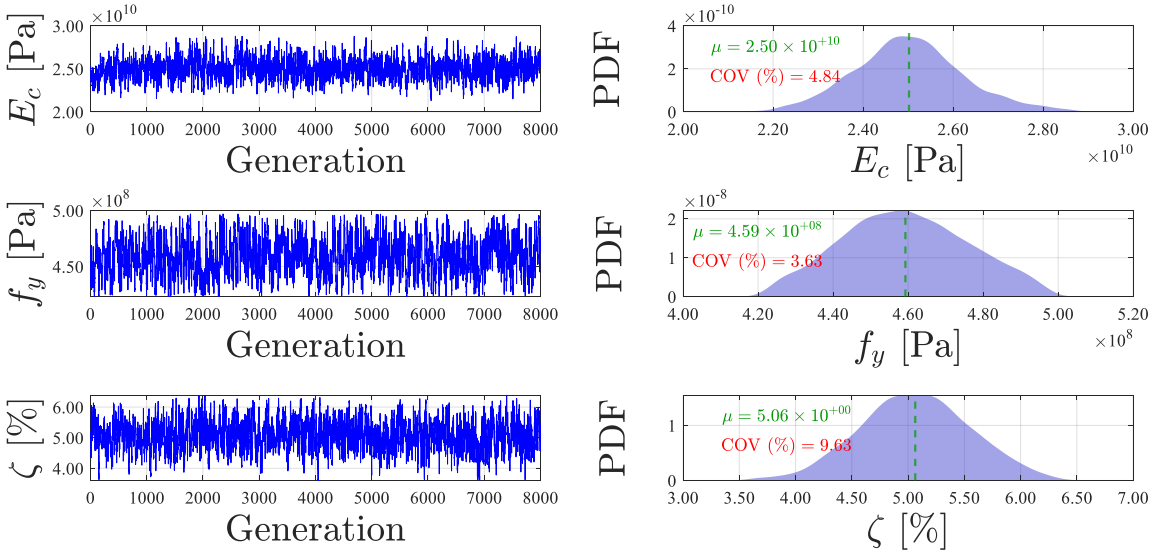


Figure 9: MCMC trace and posterioir PDF using the DR algoritm

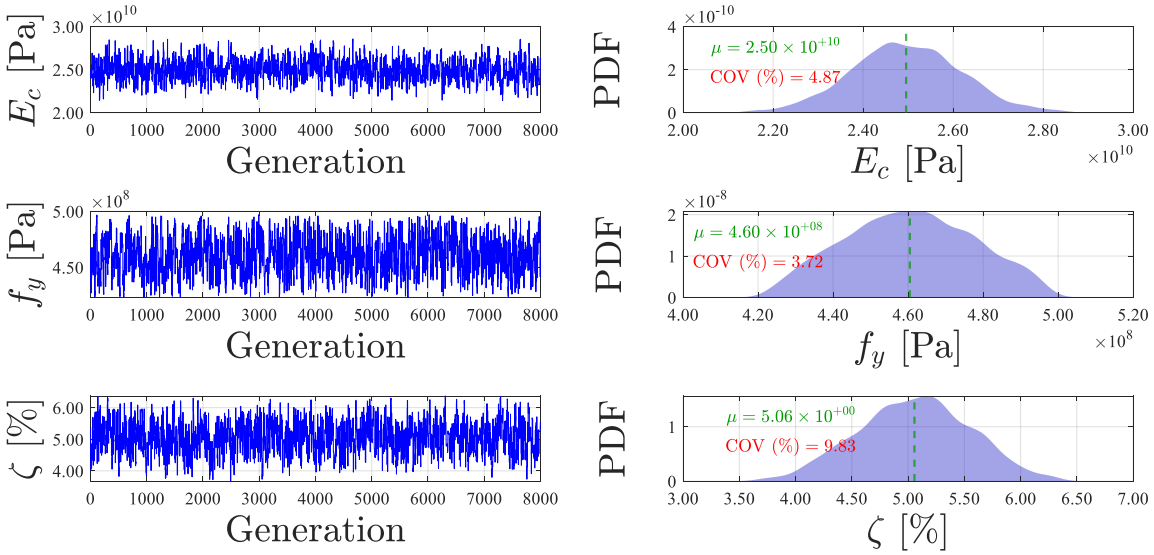


Figure 10: MCMC trace and posterioir PDF using the AM algoritm

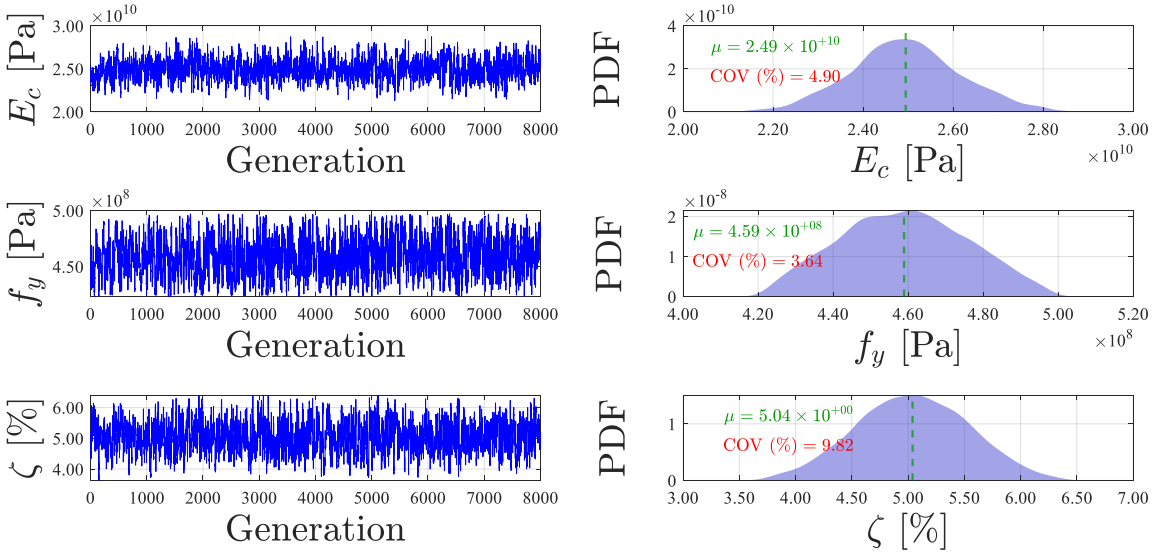


Figure 11: MCMC trace and posterioir PDF using the DRAM algoritm

4 Conclusions

In this study, two distinct approaches—deterministic and probabilistic—were investigated for nonlinear finite element model updating (FEMU) of a reinforced concrete bridge pier subjected to seismic loading. As bridge piers are critical components in the seismic response of highway bridges, their accurate modeling is essential for post-earthquake assessment.

The results have demonstrated that both deterministic and probabilistic approaches are capable of accurately estimating the true values of the selected model parameters in this case study. While the probabilistic approach offers the additional advantage of uncertainty quantification by updating the full probability distributions of the parameters, it comes with significantly higher computational costs and increased complexity—particularly in tuning hyperparameters to ensure chain stability. The comparison between different sampling algorithms within the probabilistic framework revealed similar accuracy levels due to the limited number of parameters (three in this case). However, this similarity may not hold in higher-dimensional problems, where algorithmic efficiency and accuracy could diverge.

The use of OpenSees proved highly effective due to its computational efficiency in nonlinear dynamic analyses. Parallel computing was successfully employed within the genetic algorithm (GA) used for the deterministic approach, leading to substantial reductions in computational time. In contrast, traditional MCMC algorithms do not inherently support parallelization, making them less efficient for large-scale applications. To address this limitation, more advanced sampling methods such as Transitional Markov Chain Monte Carlo (TMCMC) are recommended for future probabilistic FEMU studies. In conclusion, the integration of dynamic measurements, numerical modeling, and optimization algorithms offers a powerful toolset for enhancing the predictive accuracy of structural models. This approach holds promise for application in structural health monitoring, seismic vulnerability assessment, and performance-based engineering of reinforced concrete structures.

Several directions are proposed for further research to enhance FEMU methodologies:

- Inclusion of a larger set of model parameters, particularly those influencing nonlinear behavior.

- Accounting for correlations between parameters and updating them jointly.
- Development of a general framework to automatically select or optimize hyperparameters based on the structural model and parameter set.

Acknowledgements

Support by FABRE – "Research consortium for the evaluation and monitoring of bridges, viaducts and other structures" (www.consorziofabre.it/en) within the activities of the FABRE-ANAS 2021-2025 research program is gratefully acknowledged by the authors. Any opinions expressed in the paper do not necessarily reflect the views of the funder.

REFERENCES

[1] S. Ereiz, I. Duvnjak, J.F. Jiménez-Alonso, Review of finite element model updating methods for structural applications. *Structures*, **41**, 684–723, 2022.

[2] J.E. Mottershead, M.I. Friswell, Model updating in structural dynamics: A survey. *Journal of Sound and Vibration*, **167**(2), 347–375, 1993.

[3] J.L. Beck, L.S. Katafygiotis, Updating models and their uncertainties. I: Bayesian statistical framework. *Journal of Engineering Mechanics*, **124**(4), 455–461, 1998.

[4] K.V. Yuen, *Bayesian Methods for Structural Dynamics and Civil Engineering*. John Wiley & Sons, 2010.

[5] B. Jaishi, W.X. Ren, Structural finite element model updating using response surface method. *Engineering Structures*, **27**(3), 261–271, 2005.

[6] J.F. Jiménez-Alonso, R. Poyraz, A. Sáez, Multi-objective model updating of footbridge models using GA and dynamic measurements. *Engineering Structures*, **244**, 112792, 2021.

[7] H. Sun, Y.L. Xu, Multi-objective optimization-based FEMU for large-scale aqueduct structures. *Journal of Bridge Engineering*, **24**(10), 04019094, 2019.

[8] J.L. Beck, L.S. Katafygiotis, et al., Bayesian updating of structural models and reliability using Markov Chain Monte Carlo simulation. *Journal of Engineering Mechanics*, **127**(4), 438–455, 2001.

[9] L.S. Katafygiotis, C. Papadimitriou, Bayesian updating of structural systems using MCMC simulation with applications to model and damage identification. *Probabilistic Engineering Mechanics*, **26**(3), 357–370, 2011.

[10] Center for Engineering Strong Motion Data (CESMD), CESMD – A Cooperative Effort, [Online]. Available: `http://strongmotioncenter.org/`. [Accessed: September 2014].

[11] H. Haario, E. Saksman, J. Tamminen, An adaptive Metropolis algorithm. *Bernoulli*, **7**(2), 223–242, 2001.

[12] A. Mira *et al.*, On Metropolis-Hastings algorithms with delayed rejection. *Metron*, **59**(3–4), 231–241, 2001.

[13] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, **21**(6), 1087–1092, 1953.

[14] W.K. Hastings, Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, **57**(1), 97–109, 1970.

[15] H. Haario, M. Laine, A. Mira, E. Saksman, DRAM: efficient adaptive MCMC. *Statistics and Computing*, **16**, 339–354, 2006.

[16] P. Gardoni, A. Der Kiureghian, K.M. Mosalam, Probabilistic capacity models and fragility estimates for reinforced concrete columns based on experimental observations. *Journal of Engineering Mechanics*, **128**(10), 1024–1038, 2002.

[17] A. Gelman, G.O. Roberts, W.R. Gilks, Efficient Metropolis jumping rules. *Bayesian Statistics 5*, **5**, 599–608, 1996.

[18] M. Laine, et al., Adaptive MCMC methods with applications in environmental and geophysical models. *Finnish Meteorological Institute*, 2008.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

A GENERALIZED CONTACT POINT RESPONSE ESTIMATION VIA ON-BOARD SENSING

Ho Man Siu[1], Elias G. Dimitrakopoulos[1], Costas Papadimitriou[2]

[1]The Hong Kong University of Science and Technology
Department of Civil and Environmental Engineering
Clear Water Bay, Kowloon, Hong Kong, China
e-mail: hmsiu@connect.ust.hk, ilias@ust.hk

[2]University of Thessaly
Department of Mechanical Engineering
Pedion Areos, Volos 38334
e-mail: costasp@uth.gr

Abstract. *This work aims to present a generalized approach for estimating contact point acceleration response using only vehicle measurement, assuming the parameter of the vehicle model is known. The estimated contact point response is useful in identifying the frequencies of the passing bridge since the contact point response is free of vehicle frequency. To solve the input-state estimation problem involved, we employ an Augmented Kalman Filter (AKF). This work presents proof of concept utilizing a simplified single-degree-of-freedom (SDOF) and a more realistic multi-degree-of-freedom (MDOF) vehicle model, both representing typical road vehicles. Meanwhile, we model the bridge as a simply supported beam with the finite element method. This analysis provides valuable insight into the effect of different parameters on the accuracy of contact point response. Our demonstration relies on simulated data from a highway bridge-vehicle interaction model, incorporating the impact of road roughness.*

Keywords: On-board sensing, bridges, input-state estimation, frequency estimation, contact point response.

1 INTRODUCTION

The modal properties of a bridge often reflect the state of structural health. Therefore, monitoring the changes in modal properties can facilitate proper maintenance action. Conventionally, engineers rely on sensors mounted at different locations on the target bridge [1] to estimate modal frequencies. Although numerous studies have shown the robustness and efficiency of these direct methods [2, 3, 4, 5, 6], such techniques often come with significant installation, labor, and maintenance expenses. Importantly, this type of approach suffers from the restriction of "one-system-per bridge", hence it is challenging to meet the growing demand for bridge structural health monitoring.

Alternatively, indirect approaches, a.k.a. vehicle scanning methods, have proliferated in the bridge structural health monitoring field in the past several decades, since such methods enable the monitoring of multiple bridges with one instrumented test vehicle. Yang et al. [7] first proposed the concept of extracting bridge frequencies from the response of a single-degree-of-freedom (SDOF) vehicle traversing the bridge and demonstrated its feasibility experimentally [8]. Subsequent studies have explored estimating frequencies and modal shapes from vehicle acceleration measurements using various signal processing techniques, including time-frequency domain decomposition [9], continuous wavelet transform [10], Hilbert-Huang transform [11], and two-peak spectrum idealized filter [12].

Although extracting bridge modal information from vehicle responses is feasible, vehicle frequencies often overlap with bridge frequencies in the vehicle response spectrum, complicating the identification process. To overcome this challenge, using the contact point response of the scanning vehicle helps mitigate vehicle frequency interference [13, 14, 15]. Utilizing this property of the contact point signal, recent studies proposed various signal processing techniques for the extraction of bridge frequencies [16], damping ratios [16], and mode shapes [13, 15]. Nevertheless, road roughness can still obscure the bridge signal if the bridge response is weak. To tackle this issue, recent studies have utilized the concept that vehicle wheels traverse the same road roughness at different times, focusing on residual vehicle contact point signals. Jin et al. [17, 18, 19] suggested the application of subspace identification techniques on the residual vehicle contact point signal to estimate bridge frequencies. Meanwhile, Yang et al. [20, 21, 22] proposed the use of residual contact point acceleration alongside signal processing to obtain bridge modal properties.

Directly measuring contact point response presents challenges due to the lack of appropriate sensors. Thus, inferring contact point response from vehicle measurements becomes necessary. Most studies have employed analytical approaches to back-calculate the contact point acceleration of vehicles. However, the analytical expressions for back-calculating contact point acceleration vary with the types of vehicle models [13, 14, 15, 16, 20, 21, 22, 23, 24]. Instead of relying on the analytical expression to back-calculate the contact point response, the field has also adopted Bayesian inference techniques to infer the contact point signal. In particular, the application of the Bayesian expectation-maximization algorithm has been suggested to estimate the contact point displacement [25]. Building on the idea of using the Bayesian filtering technique, this study introduces a new state-space formulation that relates contact point acceleration to the flexible modal response of the vehicle. Based on the proposed state-space model, we estimate contact point acceleration using an Augmented Kalman Filter (AKF) [26, 27].

2 VEHICLE BRIDGE SYSTEM MODELING

In this study, we examine a vehicle transversing a bridge at a constant speed v. The primary focus is on the dynamics of the vehicle-bridge system in the vertical plane. We model the vehicle as a multibody with m degrees of freedom (DOF) and use the Euler-Bernoulli beam element to construct the bridge model (Figure 1). The time-varying contact force(s) $\boldsymbol{\lambda}(t)$ between the wheels and the bridge couples the vehicle bridge systems, which is also known as vehicle-bridge interaction (VBI).

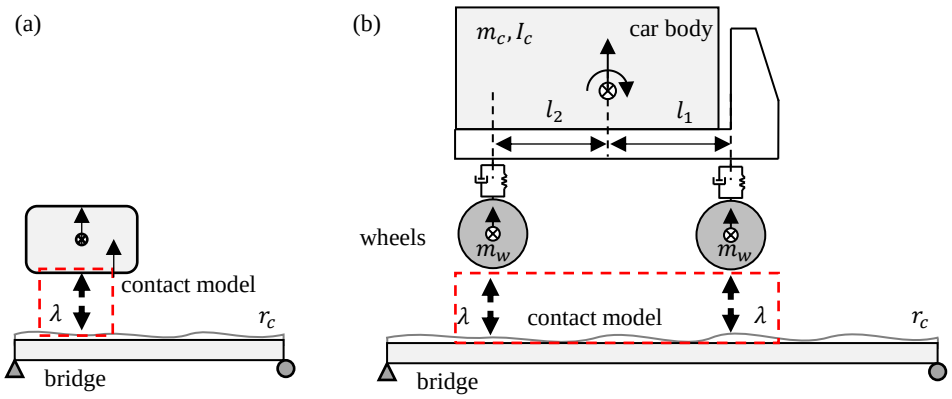


Figure 1: (a) SDOF (b) MDOF vehicle model on a bridge.

The equation of motion (EOM) of the vehicle sub-system can take the form:

$$\mathbf{m}_v\ddot{\mathbf{u}}_v(t) + \mathbf{c}_v\dot{\mathbf{u}}_v(t) + \mathbf{k}_v\mathbf{u}_v(t) - \mathbf{W}_v\boldsymbol{\lambda}(t) = \mathbf{f}_v \tag{1}$$

where, $\mathbf{u}_v(t)$ is the displacement vector of the vehicle and $\mathbf{m}_v, \mathbf{c}_v, \mathbf{k}_v$ are, respectively, the mass, damping, and stiffness matrices of the vehicle (where subscript $()_v$ denotes the vehicle subsystem). $\mathbf{W}_v$ is a contact direction matrix of the vehicle, which maps the contact forces to the wheels' DOFs [28, 29, 30, 31].

Similar to the vehicle subsystem, the EOM of the bridge about its (statically) deformed configuration under its self-weight is:

$$\mathbf{m}_b\ddot{\mathbf{u}}_b(t) + \mathbf{c}_b\dot{\mathbf{u}}_b(t) + \mathbf{k}_b\mathbf{u}_b(t) + \mathbf{W}_b\boldsymbol{\lambda}(t) = \mathbf{W}_b\mathbf{f}_{wv} \tag{2}$$

with $\mathbf{u}_b(t)$ being the displacement vector of the bridge and $\mathbf{m}_b$, $\mathbf{c}_b$, and $\mathbf{k}_b$ being the mass, damping, and stiffness matrices of the bridge (subscript $()_b$ denotes the bridge subsystem). $\mathbf{W}_b$ is the contact direction matrix of the bridge that maps the contact force $\boldsymbol{\lambda}$ and the vehicle weight $\mathbf{f}_{wv}$ to the appropriate DOFs of the bridge [28, 29, 30, 31].

Combining Eqs. (1) and (2), the EOM of the coupled vehicle-bridge system becomes:

$$\mathbf{M}\ddot{\mathbf{u}}(t) + \mathbf{C}\dot{\mathbf{u}}(t) + \mathbf{K}\mathbf{u}(t) + \mathbf{W}_c\boldsymbol{\lambda}(t) = \mathbf{f} \tag{3}$$

where, $\mathbf{u}$ is the displacement vector of the whole system and $\mathbf{M}$, $\mathbf{C}$, and $\mathbf{K}$ are, respectively, the mass, damping, and stiffness matrices of the whole system:

$$\mathbf{M} = \begin{bmatrix} \mathbf{m}_v & \mathbf{0} \\ \mathbf{0} & \mathbf{m}_b \end{bmatrix}, \mathbf{C} = \begin{bmatrix} \mathbf{c}_v & \mathbf{0} \\ \mathbf{0} & \mathbf{c}_b \end{bmatrix}, \mathbf{K} = \begin{bmatrix} \mathbf{k}_v & \mathbf{0} \\ \mathbf{0} & \mathbf{k}_b \end{bmatrix} \tag{4}$$

and $\mathbf{W}_c = \begin{bmatrix} -\mathbf{W}_v{}^{\mathrm{T}} & \mathbf{W}_b{}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ is the contact direction matrix of the coupled system.

In this study, we assume a compliance contact model consisting of a spring and a damper connected in parallel between the wheel and the bridge. Accordingly, the contact force acting on the i^{th} wheel λ_i has the following expression: [29, 32].

$$\lambda_i(t) = k_{\lambda,i}\left(-u_{w,i} + u_{cp,i}\right) + c_{\lambda,i}\left(-\dot{u}_{w,i} + \dot{u}_{cp,i}\right) \tag{5}$$

where $k_{\lambda,i}$ and $c_{\lambda,i}$ are the contact stiffness and damping coefficient respectively, and $u_{w,i}$ is the displacement of the i^{th} wheel. The contact point response (displacement $u_{cp,i}$, velocity $\dot{u}_{cp,i}$ and acceleration $\ddot{u}_{cp,i}$) has the following expression:

$$u_{cp,i} = \mathbf{W}_{b,i}^{\mathrm{T}}(x_i)\mathbf{u}_b(t) + r_c(x_i) \tag{6}$$

$$\dot{u}_{cp,i} = \mathbf{W}_{b,i}^{\mathrm{T}}(x_i)\dot{\mathbf{u}}_b(t) + v\mathbf{W}'^{\mathrm{T}}_{b,i}(x_i)\mathbf{u}_b(t) + v{r_c}'(x_i) \tag{7}$$

$$\ddot{u}_{cp,i} = \mathbf{W}_{b,i}^{\mathrm{T}}(x_i)\ddot{\mathbf{u}}_b(t) + 2v\mathbf{W}'^{\mathrm{T}}_{b,i}(x_i)\dot{\mathbf{u}}_b(t) + v^2\mathbf{W}''^{\mathrm{T}}_{b,i}(x_i)\mathbf{u}_b(t) + v^2{r_c}''(x_i) \tag{8}$$

where r_c is the road roughness and $\mathbf{W}_{b,i}$ is the i^{th} column of $\mathbf{W}_b$. The prime ($\square'$) denotes the derivative with respect to the space variable x.

We model the road roughness (Figure 2) as stationary stochastic processes using the spectral representation method [33]. and adopt the road roughness parameters from ISO 8606:2016 [34].

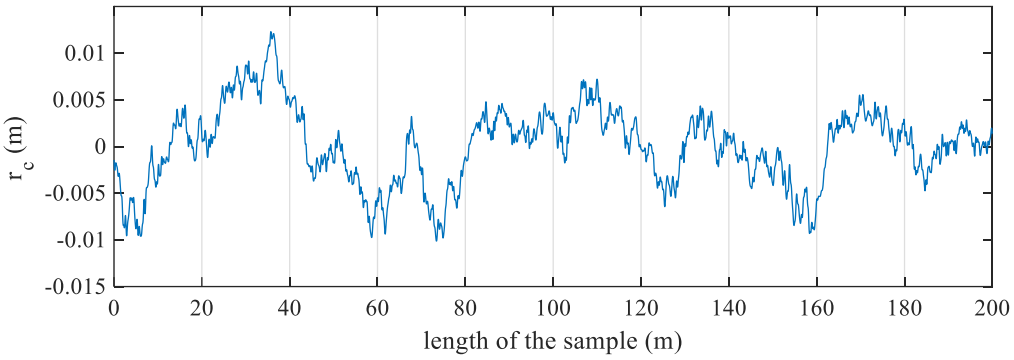


Figure 2: Sample of class A road roughness profiles according to ISO8606 [34].

3 Contact point response estimation

The contact point response and the vehicle response are connected. To study this, we consider a modified vehicle system (see Figure 3), where the vehicle model includes additional DOFs with small mass that remain in continuous contact with the bridge/ road surface. This approach works because the compliance contact model represents a special case of a rigid contact model with a small (close to zero) contact mass.

3.1 Modal analysis of the modified vehicle

The previous study [27] demonstrates the feasibility of estimating contact force using the state-space model of flexible mode dynamics. By introducing a partitioned mass-normalized mode shape matrix of the modified vehicle system:

$$\mathbf{\Phi}_v = \begin{bmatrix} \mathbf{\Phi}_{rigid} & \mathbf{\Phi}_{flex} \end{bmatrix} \tag{9}$$

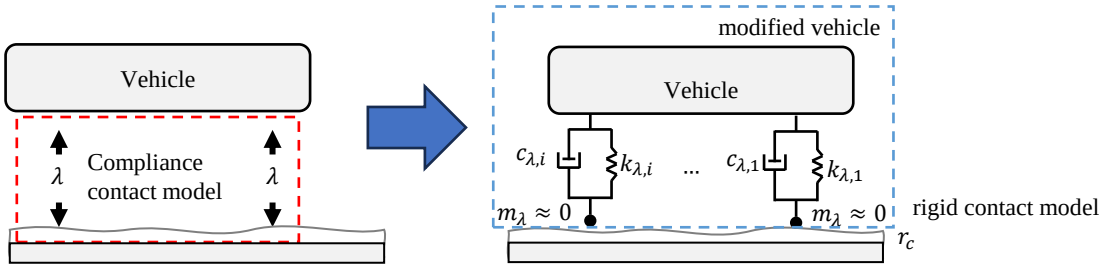


Figure 3: Modification of vehicle model.

where matrix $\mathbf{\Phi}$ comprises all m_0 rigid body mode shape(s) (mode(s) with zero eigenfrequency) $\mathbf{\Phi}_{rigid}$ and all m_f flexible mode shape(s) $\mathbf{\Phi}_{flex}$ (where $m_f = m - m_0$), the relationship between the physical and modal coordinates of the modified vehicle is:

$$\mathbf{u}_v(t) = \mathbf{\Phi}_v \boldsymbol{\xi}_v \tag{10}$$

where $\boldsymbol{\xi}_v(t)$ represents the modal displacement vector. One can partition the modal displacement vector into two parts:

$$\boldsymbol{\xi}_v = \begin{bmatrix} \boldsymbol{\xi}_{rigid}^{\mathrm{T}} & \boldsymbol{\xi}_{flex}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \tag{11}$$

where $\boldsymbol{\xi}_{rigid}$ is the modal displacement of the rigid body mode(s) and $\boldsymbol{\xi}_{flex}$ is the modal displacement of the flexible modes. Substituting Eq. (10) into Eq. (1) and pre-multiplying both sides by $\mathbf{\Phi}^{\mathrm{T}}$ yields the EOM of the modified vehicle in modal coordinates:

$$\ddot{\boldsymbol{\xi}}_{rigid} = \mathbf{\Phi}_{rigid}^{\mathrm{T}} \mathbf{W}_v \boldsymbol{\lambda}(t) \tag{12}$$

$$\ddot{\boldsymbol{\xi}}_{flex} + \mathbf{Z}_{flex} \dot{\boldsymbol{\xi}}_{flex} + \mathbf{\Lambda}_{flex} \boldsymbol{\xi}_{flex} = \mathbf{\Phi}_{flex}^{\mathrm{T}} \mathbf{W}_v \boldsymbol{\lambda}(t) \tag{13}$$

with $\mathbf{Z}_{flex} = \mathbf{\Phi}_{flex}^{\mathrm{T}} \mathbf{c}_v \mathbf{\Phi}_{flex}$ and $\mathbf{\Lambda}_{flex} = \mathbf{\Phi}_{flex}^{\mathrm{T}} \mathbf{k}_v \mathbf{\Phi}_{flex}$.

From the equation above, we can express the vehicle displacement and acceleration as follows:

$$\mathbf{u}_v = \mathbf{u}_{v,rigid} + \mathbf{u}_{v,flex} = \mathbf{\Phi}_{rigid} \boldsymbol{\xi}_{rigid} + \mathbf{\Phi}_{flex} \boldsymbol{\xi}_{flex} \tag{14}$$

$$\ddot{\mathbf{u}}_v = \left[\mathbf{\Phi}_{rigid} \mathbf{\Phi}_{rigid}^{T} + \mathbf{\Phi}_{flex} \mathbf{\Phi}_{flex}^{\mathrm{T}} \right] \mathbf{W}_v \boldsymbol{\lambda}(t) - \mathbf{\Phi}_{flex} \mathbf{\Lambda}_{flex} \boldsymbol{\xi}_{flex} - \mathbf{\Phi}_{flex} \mathbf{Z}_{flex} \dot{\boldsymbol{\xi}}_{flex} \tag{15}$$

where $\mathbf{u}_{v,rigid}$ and $\mathbf{u}_{v,flex}$ are the contribution of the rigid body and flexible modes in vehicle displacement, respectively.

Then, we can derive the following relationship between the contact point acceleration and contact force:

$$\ddot{\mathbf{u}}_{cp} = \mathbf{W}_v^{\mathrm{T}} \ddot{\mathbf{u}}_v = \mathbf{W}_v^{\mathrm{T}} \mathbf{\Phi} \mathbf{\Phi}^{\mathrm{T}} \mathbf{W}_v \boldsymbol{\lambda}(t) - \mathbf{W}_v^{\mathrm{T}} \mathbf{\Phi}_{flex} \mathbf{Z}_{flex} \dot{\boldsymbol{\xi}}_{flex} - \mathbf{W}_v^{\mathrm{T}} \mathbf{\Phi}_{flex} \mathbf{\Lambda}_{flex} \boldsymbol{\xi}_{flex} \tag{16}$$

$$\boldsymbol{\lambda}(t) = \left(\mathbf{W}_v^{\mathrm{T}} \mathbf{\Phi} \mathbf{\Phi}^{\mathrm{T}} \mathbf{W}_v \right)^{-1} \left(\ddot{\mathbf{u}}_{cp} + \mathbf{W}_v^{\mathrm{T}} \mathbf{\Phi}_{flex} \mathbf{Z}_{flex} \dot{\boldsymbol{\xi}}_{flex} + \mathbf{W}_v^{\mathrm{T}} \mathbf{\Phi}_{flex} \mathbf{\Lambda}_{flex} \boldsymbol{\xi}_{flex} \right) \tag{17}$$

Substituting Eq. (17) into Eq. (12) and (13) yields the modal EOM of the vehicle with contact point acceleration $\ddot{\mathbf{u}}_{cp}$ as input:

$$\ddot{\boldsymbol{\xi}}_{rigid} = \mathbf{\Phi}_{rigid}^{\mathrm{T}} \mathbf{W}_v \left(\mathbf{W}_v^{\mathrm{T}} \mathbf{\Phi} \mathbf{\Phi}^{\mathrm{T}} \mathbf{W}_v \right)^{-1} \left(\ddot{\mathbf{u}}_{cp} + \mathbf{W}_v^{\mathrm{T}} \mathbf{\Phi}_{flex} \mathbf{Z}_{flex} \dot{\boldsymbol{\xi}}_{flex} + \mathbf{W}_v^{\mathrm{T}} \mathbf{\Phi}_{flex} \mathbf{\Lambda}_{flex} \boldsymbol{\xi}_{flex} \right) \tag{18}$$

$$\begin{aligned}\ddot{\boldsymbol{\xi}}_{flex} = {} & \boldsymbol{\Phi}_{flex}^{\mathrm{T}}\mathbf{W}_v\left(\mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\right)^{-1}\left(\ddot{\mathbf{u}}_{cp} + \mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}_{flex}\mathbf{Z}_{flex}\dot{\boldsymbol{\xi}}_{flex} + \mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}_{flex}\boldsymbol{\Lambda}_{flex}\boldsymbol{\xi}_{flex}\right) \\ & - \mathbf{Z}_{flex}\dot{\boldsymbol{\xi}}_{flex} - \boldsymbol{\Lambda}_{flex}\boldsymbol{\xi}_{flex}\end{aligned} \tag{19}$$

From Eqs. (18) and (19), we can express the vehicle acceleration in term of $\ddot{\mathbf{u}}_{cp}$, $\dot{\boldsymbol{\xi}}_{flex}$ and $\boldsymbol{\xi}_{flex}$ as:

$$\begin{aligned}\ddot{\mathbf{u}}_v = {} & \boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\left(\mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\right)^{-1}\ddot{\mathbf{u}}_{cp} + \left(\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\left(\mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\right)^{-1}\mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}_{flex} - \boldsymbol{\Phi}_{flex}\right)\mathbf{Z}_{flex}\dot{\boldsymbol{\xi}}_{flex} \\ & + \left(\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\left(\mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\right)^{-1}\mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}_{flex} - \boldsymbol{\Phi}_{flex}\right)\boldsymbol{\Lambda}_{flex}\boldsymbol{\xi}_{flex}\end{aligned} \tag{20}$$

and the strain at the suspension springs as:

$$\boldsymbol{\varepsilon}(t) = \boldsymbol{\Psi}\mathbf{u}_v(t) = \boldsymbol{\Psi}\boldsymbol{\Phi}_{flex}\boldsymbol{\xi}_{flex}(t) \tag{21}$$

where $\boldsymbol{\Psi}$ is the matrix that associates the displacements at all DOFs to the strains at the springs of the train vehicle model.

3.2 State-space model of the vehicle with contact point acceleration as input

The state-space form of the modal EOM for the modified vehicle model (Eq. (13)) is: [35]:

$$\dot{\mathbf{x}}(t) = \mathbf{A}_c\mathbf{x}(t) + \mathbf{B}_c\ddot{\mathbf{u}}_{cp}(t) \tag{22}$$

where the state vector $\mathbf{x}(t)$, state matrix $\mathbf{A}_c$ and the input matrix $\mathbf{B}_c$ are:

$$\mathbf{x}(t) = \begin{bmatrix}\boldsymbol{\xi}_{flex} \\ \dot{\boldsymbol{\xi}}_{flex}\end{bmatrix}, \quad \mathbf{A}_c = \begin{bmatrix}\mathbf{0} & \mathbf{I} \\ \mathbf{I}_{flex}\boldsymbol{\Lambda}_{flex} & \mathbf{I}_{flex}\mathbf{Z}_{flex}\end{bmatrix}, \quad \mathbf{B}_c = \begin{bmatrix}\mathbf{0} \\ \boldsymbol{\Phi}_{flex}^{\mathrm{T}}\mathbf{W}_v\left(\mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\right)^{-1}\end{bmatrix} \tag{23}$$

$$\mathbf{I}_{flex} = \left(\boldsymbol{\Phi}_{flex}^{\mathrm{T}}\mathbf{W}_v\left(\mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\right)^{-1}\mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}_{flex} - \mathbf{I}\right) \tag{24}$$

The observation equation for accelerations and strain measurements is:

$$\mathbf{y}(t) = \mathbf{G}\mathbf{x}(t) + \mathbf{J}\ddot{\mathbf{u}}_{cp}(t) + \mathbf{v}(t) \tag{25}$$

where $\mathbf{y}(t)$ is the measurement vector in continuous time, the measurement noise is white, i.e., $\mathbf{v} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_y)$, with measurement error covariance $\mathbf{Q}_y \in \mathbb{R}^{N_O \times N_O}$ and N_O is the number of measured quantities. From Eqs. (20) and (21), $\mathbf{G}$ and $\mathbf{J}$ matrices for acceleration and strain measurements it follows:

$$\mathbf{G} = \begin{bmatrix}\mathbf{H}\mathbf{I}_{mod}\boldsymbol{\Lambda}_{flex} & \mathbf{H}\mathbf{I}_{mod}\mathbf{Z}_{flex} \\ \mathbf{S}\boldsymbol{\Psi}\boldsymbol{\Phi}_{flex} & \mathbf{0}\end{bmatrix}, \quad \mathbf{J} = \begin{bmatrix}\mathbf{H}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\left(\mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\right)^{-1} \\ \mathbf{0}\end{bmatrix}, \tag{26}$$

where

$$\mathbf{I}_{mod} = \left(\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\left(\mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{W}_v\right)^{-1}\mathbf{W}_v^{\mathrm{T}}\boldsymbol{\Phi}_{flex} - \boldsymbol{\Phi}_{flex}\right) \tag{27}$$

In practice, we take vehicle measurements at discrete time instant $t_k = k\Delta t$, where k is the time step and Δt is the sampling time interval. Hence, it is necessary to discretize the vehicle

state-space model. Using the Tustin method [36], we can express the discrete state-space form of the EOM for the vehicle as:

$$\mathbf{x}_{k+1} = \mathbf{A}_d \mathbf{x}_k + \mathbf{B}_d \ddot{\mathbf{u}}_{cp,k} + \mathbf{w}_k \tag{28}$$

$$\mathbf{y}_k = \mathbf{G}_d \mathbf{x}_k + \mathbf{J}_d \ddot{\mathbf{u}}_{cp,k} + \mathbf{v}_k \tag{29}$$

where the value of the state-space vector and the contact point acceleration at time instant t_k are $\mathbf{x}_k$ and $\ddot{\mathbf{u}}_{cp,k}$, respectively ($\mathbf{x}_k \equiv \mathbf{x}(k\Delta t)$ and $\ddot{\mathbf{u}}_{cp,k} \equiv \ddot{\mathbf{u}}_{cp}(k\Delta t)$). We assume the process noise $\mathbf{w}_k$ is zero-mean Gaussian, i.e., $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_x)$ with process noise covariance $\mathbf{Q}_x \in \mathbb{R}^{2m_f \times 2m_f}$.

3.3 AKF for unknown contact point acceleration estimation

This study employs a random walk model [26, 27] to describe the changes in the unknown contact point acceleration $\ddot{\mathbf{u}}_{cp,k}$ at different time steps:

$$\ddot{\mathbf{u}}_{cp,k+1} = \ddot{\mathbf{u}}_{cp,k} + \boldsymbol{\eta}_k \tag{30}$$

where $\boldsymbol{\eta}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_\lambda)$ is a zero-mean Gaussian with covariance matrix $\mathbf{Q}_\lambda \in \mathbb{R}^{n_\lambda \times n_\lambda}$. By introducing the augmented state vector $\mathbf{x}_k^a = [\mathbf{x}_k^{\mathrm{T}}, \ddot{\mathbf{u}}_{cp,k}^{\mathrm{T}}]^{\mathrm{T}}$, the augmented state-space model and observation equations are formulated as follows:

$$\begin{aligned} \mathbf{x}_{k+1}^a &= \mathbf{A}^a \mathbf{x}_k^a + \boldsymbol{\zeta}_k \\ \mathbf{y}_k &= \mathbf{G}^a \mathbf{x}_k^a + \mathbf{v}_k \end{aligned} \tag{31}$$

where the augmented system matrix $\mathbf{A}^a \in \mathbb{R}^{(2m_f+n_\lambda)\times(2m_f+n_\lambda)}$ and observation matrix $\mathbf{G}^a \in \mathbb{R}^{N_0\times(2m_f+n_\lambda)}$ are:

$$\mathbf{A}^a = \begin{bmatrix} \mathbf{A}_d & \mathbf{B}_d \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad \mathbf{G}^a = \begin{bmatrix} \mathbf{G}_d & \mathbf{J}_d \end{bmatrix} \tag{32}$$

The augmented process noise $\boldsymbol{\zeta}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_a)$ has a covariance matrix $\mathbf{Q}_a \in \mathbb{R}^{(m_f+n_\lambda)\times(m_f+n_\lambda)}$:

$$\mathbf{Q}_a = \begin{bmatrix} \mathbf{Q}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_\lambda \end{bmatrix} \tag{33}$$

Using a Kalman Filtering algorithm [26, 27, 35], we can obtain the estimated augmented state $\hat{\mathbf{x}}_{k|k}^a$, which contains the contact point acceleration of interest. The algorithm involves a time update:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k-1}^a &= \mathbf{A}^a \hat{\mathbf{x}}_{k-1|k-1}^a \\ \mathbf{P}_{k|k-1} &= \mathbf{A}^a \mathbf{P}_{k-1|k-1} (\mathbf{A}^a)^{\mathrm{T}} + \mathbf{Q}_a \end{aligned} \tag{34}$$

and a measurement update:

$$\begin{aligned} \mathbf{K}_k &= \mathbf{P}_{k|k-1} (\mathbf{G}^a)^{\mathrm{T}} (\mathbf{G}^a \mathbf{P}_{k|k-1} (\mathbf{G}^a)^{\mathrm{T}} + \mathbf{Q}_y)^{-1} \\ \hat{\mathbf{x}}_{k|k}^a &= \hat{\mathbf{x}}_{k|k-1}^a + \mathbf{K}_k (\mathbf{y}_k - \mathbf{G}^a \hat{\mathbf{x}}_{k|k-1}^a) \\ \mathbf{P}_{k|k} &= \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{G}^a \mathbf{P}_{k|k-1} \end{aligned} \tag{35}$$

for every time step.

4 NUMERICAL EXAMPLES

To evaluate the performance of the proposed Augmented Kalman Filter (AKF), we employ simulated data from bridge-vehicle interaction model. In the following examples, the vehicle enters the bridge at $t = 5$s and moves along a simply supported bridge modeled with Euler-Bernoulli beams element. The bridge is 30 m long with mass per unit length 36t/m, flexural rigidity $EI = 205\text{GNm}^2$, and the damping ratio of the bridge is $\zeta_b = 2\%$ [17].

4.1 SDOF vehicle

To verify the proposed method, this section studies a simplified SDOF vehicle (as illustrated in Figure 1(a)) moving across the bridge with constant speed $v = 72\text{km/h}$. The SDOF vehicle has a mass of $m_c = 5.75$t, with a suspension featuring a stiffness coefficient $k_p = 1.595$MN/m and a damping coefficient $c_p = 3.20$kN/m. We assume a road roughness profile of class A [34].

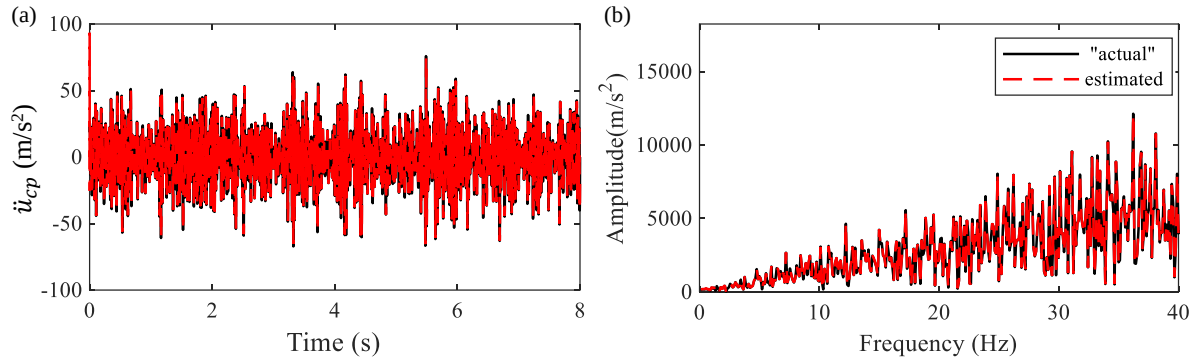


Figure 4: Comparison between the "actual" and estimated contact point acceleration using acceleration measurement of the car body: (a) time history; (b) Fourier spectrum ($v = 72\text{km/h}$).

Figure 4 compares the "actual" and estimated contact point acceleration based on acceleration measurements of the vehicle body. Throughout this study, the term "actual" refers to the response obtained from vehicle-bridge simulation. The estimated contact point acceleration and the corresponding spectrum closely match the "actual" contact point acceleration when there is no measurement noise. This demonstrates the effectiveness of the proposed state-space model and AKF in accurately estimating the unknown contact point acceleration.

4.2 2 axles MDOF vehicle

The section examines a more realistic 2-axles MDOFs truck vehicle (Figure 1(b)) moving at a constant speed $v = 72\text{km/h}$. The road roughness profile remains the same. Table 1 shows the mechanical properties of the vehicle [37].

Table 1: Mechanical properties of the 2 axles MDOF vehicle

m_c (t)	I_c (tm^2)	m_{w1} (t)	m_{w2} (t)	k (kN/m)	c (kNs/m)	k_λ (kN/m)	c_λ (kNs/m)	l_1 (m)	l_2 (m)
4.48	5.52	0.8	0.71	798	46.42	702	1.6	2.6	3

Figure 5 compares the "actual" and estimated contact point acceleration using acceleration measurement of all DOF and the strain measurements of all the suspension springs. The AKF method effectively captures all high-frequency details of the contact point acceleration. The accuracy of the estimated contact point acceleration for both axles, in both the time and frequency

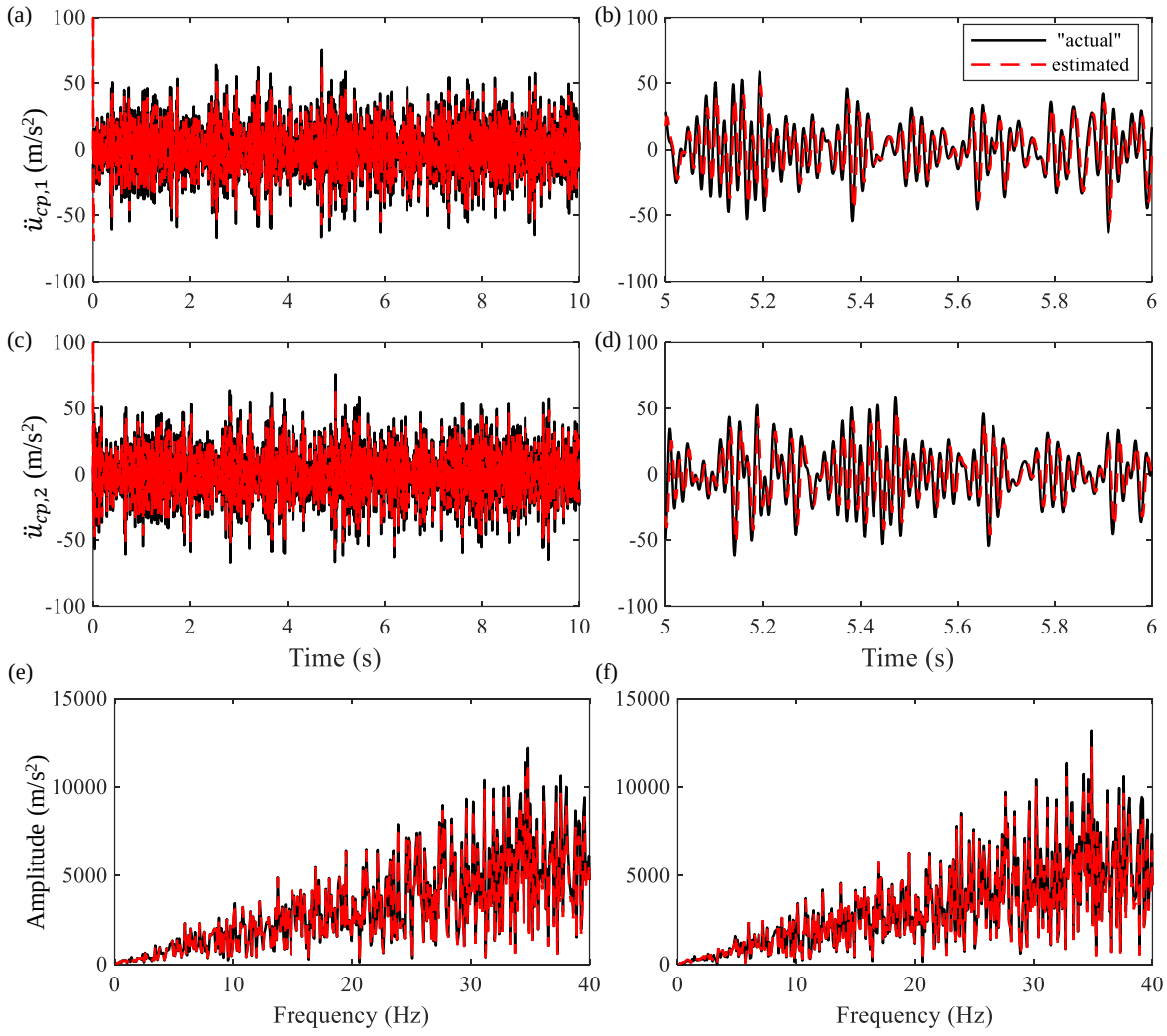


Figure 5: Comparison between the "actual" and estimated contact point acceleration of (a) wheel 1, and (b) wheel 2 using acceleration measurements at all DOFs and strain measurements of all suspensions ($v = 72$km/h).

domains, demonstrates the versatility of the proposed state-space formulation. This formulation applies to any vehicle type, making it valuable for real-world applications that require estimating contact point acceleration for various vehicles.

To extract useful information about the passing bridge, such as modal frequencies, from vehicle measurements, numerous studies have shown that the residual contact point response (i.e., acceleration) serves as a better signal for this purpose, as it remains free from the effects of road roughness and vehicle frequency.

Figure 6 illustrates the comparison between the "actual" and estimated residual contact point acceleration. Although a significant error appears in the estimated residual contact point when the vehicle's first wheel enters ($t = 5$s) and exits ($t = 6.5$s) the bridge, the estimated residual contact point acceleration aligns closely with the "actual" values in the time domain. Furthermore, we can observe the first three modes of the bridge within the spectrum of the estimated residual contact point acceleration. This observation highlights the potential of the proposed formulation to estimate bridge properties using only vehicle measurements.

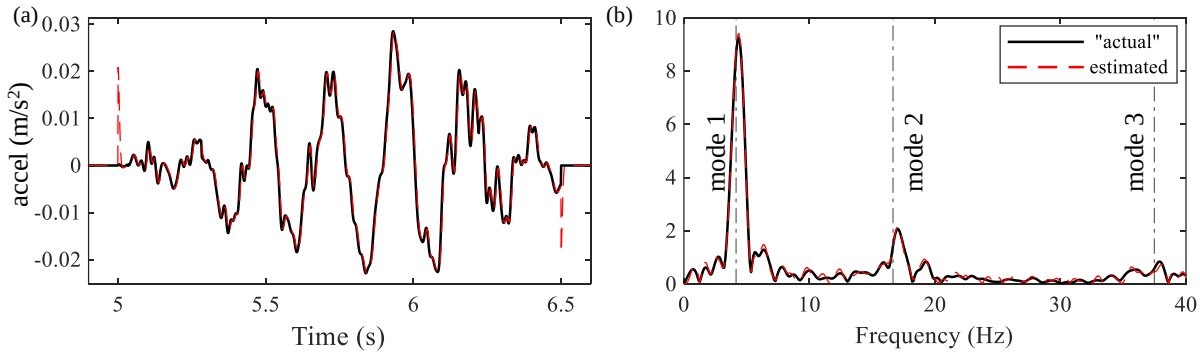


Figure 6: Comparison between the "actual" and estimated residual contact point acceleration: (a) time history; (b) Fourier spectrum

5 CONCLUSIONS

This study explores the feasibility of identifying the vehicle-bridge contact point acceleration using only measurements from on-board sensors installed on vehicles. Specifically, it proposes employing the Augmented Kalman Filter (AKF) to identify the unknown contact point acceleration experienced by the passing vehicle. The study assumes a known vehicle model for the vehicle dynamics. Using a mode superposition description of the vehicle dynamics, we remove the influence of the rigid body mode(s) and formulate a state-space model with contact point acceleration as the input. To demonstrate the effectiveness of the AKF under various conditions, the analysis relies on numerical simulation experiments. The vehicle-bridge simulation examines SDOF and MDOFs vehicle models while considering road roughness. Comparisons between the estimated and "actual" contact point acceleration show that the AKF can achieve high accuracy when properly tuned with suitable covariance settings and sufficient measurement data. The accuracy of the estimated contact point acceleration is adequate for bridge health monitoring purposes.

6 ACKNOWLEDGEMENTS

Financial support to the first author was provided by the University Grants Committee Research Grants Council of Hong Kong, under Grant Reference Number GRF 16212522.

REFERENCES

[1] Y. An, E. Chatzi, S.-H. Sim, S. Laflamme, B. Blachowski, J. Ou, Recent progress and future trends on damage identification methods for bridge structures, Structural Control and Health Monitoring 26 (10) (2019) e2416. doi:https://doi.org/10.1002/stc.2416.

[2] C. McGeown, F. Huseynov, D. Hester, P. McGetrick, E. J. O'Brien, V. Pakrashi, Using measured rotation on a beam to detect changes in its structural condition, Journal of Structural Integrity and Maintenance (2021). doi:10.1080/24705314.2021.1906092.

[3] D. Martinez, A. Malekjafarian, E. J. O'Brien, Bridge health monitoring using deflection measurements under random traffic, Structural Control Health Monitoring (2020). doi:10.1002/stc.2593.

[4] Y. Y. Li, Hypersensitivity of strain-based indicators for structural damage identification: A review, Mechanical Systems and Signal Processing (2010). doi:10.1016/j.ymssp.2009.11.002.

[5] J.-X. Mao, H. Wang, D.-M. Feng, T.-Y. Tao, W. Zheng, Investigation of dynamic properties of long-span cable-stayed bridges based on one-year monitoring data under normal operating condition, Structural Control Health Monitoring (2018). doi:10.1002/stc.2146.

[6] L. Zhang, E. J. OBrien, D. Hajializadeh, L. Deng, S. Yin, S. Yin, Bridge damage identification using rotation measurement, Journal of Bridge Engineering (2023). doi:10.1061/jbenf2.beeng-5891.

[7] Y. Yang, C. W. Lin, J. D. Yau, Extracting bridge frequencies from the dynamic response of a passing vehicle, Journal of Sound and Vibration 272 (3) (2004) 471–493. doi:https://doi.org/10.1016/S0022-460X(03)00378-X.

[8] C. Lin, Y. Yang, Use of a passing vehicle to scan the fundamental bridge frequencies: An experimental verification, Engineering Structures 27 (13) (2005) 1865–1878. doi:https://doi.org/10.1016/j.engstruct.2005.06.016.

[9] A. Malekjafarian, E. J. OBrien, Identification of bridge mode shapes using short time frequency domain decomposition of the responses measured in a passing vehicle, Engineering Structures 81 (2014) 386–397. doi:https://doi.org/10.1016/j.engstruct.2014.10.007.

[10] E. Erduran, F. M. Pettersen, S. Gonen, A. Lau, Identification of vibration frequencies of railway bridges from train-mounted sensors using wavelet transformation, Sensors 23 (3) (2023). doi:10.3390/s23031191.

[11] A. Malekjafarian, E. J. OBrien, On the use of a passing vehicle for the estimation of bridge mode shapes, Journal of Sound and Vibration 397 (2017) 77–91. doi:https://doi.org/10.1016/j.jsv.2017.02.051.

[12] J. Zhou, Z. Lu, Z. Zhou, C. Pan, S. Cao, J. Cheng, J. Zhang, Extraction of bridge mode shapes from the response of a two-axle passing vehicle using a two-peak spectrum idealized filter approach, Mechanical Systems and Signal Processing 190 (2023) 110122. doi:https://doi.org/10.1016/j.ymssp.2023.110122.

[13] Y. B. Yang, Y. H. Liu, H. Xu, Recovering mode shapes of curved bridges by a scanning vehicle, International Journal of Mechanical Sciences 253 (2023) 108404. doi:https://doi.org/10.1016/j.ijmecsci.2023.108404.

[14] H. Xu, Y. H. Liu, Z. L. Wang, K. Shi, B. Zhang, Y. B. Yang, General contact response of single-axle two-mass test vehicles for scanning bridge frequencies considering suspension effect, Engineering Structures 270 (2022) 114880. doi:https://doi.org/10.1016/j.engstruct.2022.114880.

[15] H. Xu, Y. H. Liu, M. Yang, D. S. Yang, Y. B. Yang, Mode shape construction for bridges from contact responses of a two-axle test vehicle by wavelet transform, Mechanical Systems and Signal Processing 195 (2023) 110304. doi:https://doi.org/10.1016/j.ymssp.2023.110304.

[16] Y. B. Yang, K. Shi, Z. L. Wang, H. Xu, B. Zhang, Y. T. Wu, Using a Single-DOF Test Vehicle to Simultaneously Retrieve the First Few Frequencies and Damping Ratios of the Bridge, International Journal of Structural Stability and Dynamics 21 (08) (2021) 2150108. doi:10.1142/S021945542150108X.

[17] N. Jin, Y. B. Yang, E. G. Dimitrakopoulos, T. S. Paraskeva, L. S. Katafygiotis, Application of short-time stochastic subspace identification to estimate bridge frequencies from a traversing vehicle, Engineering Structures 230 (2021) 111688. doi:https://doi.org/10.1016/j.engstruct.2020.111688.

[18] N. Jin, V. K. Dertimanis, E. N. Chatzi, E. G. Dimitrakopoulos, L. S. Katafygiotis, Subspace identification of bridge dynamics via traversing vehicle measurements, Journal of Sound and Vibration 523 (2022) 116690. doi:https://doi.org/10.1016/j.jsv.2021.116690.

[19] Y. Quan, Q. Zeng, N. Jin, Y. Zhu, C. Liu, Subspace identification of bridge frequencies based on the dimensionless response of a two-axle vehicle, Sensors 24 (6) (2024). doi:10.3390/s24061946.

[20] Y. B. Yang, X. Q. Mo, K. Shi, Z. L. Wang, H. Xu, Y. T. Wu, Contact residue for simultaneous removal of vehicle's frequency and surface roughness in scanning bridge frequencies using two connected vehicles, International Journal of Structural Stability and Dynamics 21 (13) (2021) 2171006. doi:10.1142/S0219455421710061.

[21] Y. B. Yang, X. Q. Mo, K. Shi, Z.-L. Wang, H. Xu, Y. T. Wu, Scanning torsional-flexural frequencies of thin-walled box girders with rough surface from vehicles' residual contact response: Theoretical study, Thin-Walled Structures 169 (2021) 108332. doi:https://doi.org/10.1016/j.tws.2021.108332.

[22] Y. B. Yang, H. Xu, Z.-L. Wang, K. Shi, Using vehicle–bridge contact spectra and residue to scan bridge's modal properties with vehicle frequencies and road roughness eliminated, Structural Control and Health Monitoring 29 (8) (2022) e2968. doi:https://doi.org/10.1002/stc.2968.

[23] Y. Yang, X. Mo, K. Shi, S. Gao, N. Liu, Z. Han, Effect of damping on torsional-flexural frequencies of monosymmetric thin-walled beams scanned by moving vehicles, Thin-Walled Structures 198 (2024) 111633. doi:https://doi.org/10.1016/j.tws.2024.111633.

[24] B. Wang, Y. Zeng, D. Feng, Theoretical study of extracting modal frequencies and hinge joint stiffness for thin-walled assembled multi-girder bridges from 3d vehicle, Thin-Walled Structures 205 (2024) 112429. doi:https://doi.org/10.1016/j.tws.2024.112429.

[25] J. Li, X. Zhu, S. Chen, W. Ruan, Contact-point response reconstruction for indirect bridge monitoring via bayesian expectation-maximization based augmented kalman filter, Engineering Structures 309 (2024) 118066. doi:https://doi.org/10.1016/j.engstruct.2024.118066.

[26] E. Lourens, E. Reynders, G. De Roeck, G. Degrande, G. Lombaert, An augmented Kalman filter for force identification in structural dynamics, Mechanical Systems and Signal Processing 27 (1) (2012) 446–460. doi:10.1016/j.ymssp.2011.09.025.

[27] H. M. Siu, F. Filippitzis, C. D. Stoura, C. Papadimitriou, E. G. Dimitrakopoulos, Utilizing on-board sensing of passing train vehicles for virtual sensing of bridges, Engineering Structures 319 (2024) 118808. doi:https://doi.org/10.1016/j.engstruct.2024.118808.

[28] C. D. Stoura, E. G. Dimitrakopoulos, Additional damping effect on bridges because of vehicle-bridge interaction, Journal of Sound and Vibration 476 (2020) 115294. doi:https://doi.org/10.1016/j.jsv.2020.115294.

[29] C. D. Stoura, E. G. Dimitrakopoulos, MDOF extension of the modified bridge system method for vehicle–bridge interaction, Nonlinear Dynamics 102 (2020) 2103–2123. doi:10.1007/s11071-020-06022-6.

[30] H. Homaei, E. Dimitrakopoulos, A. Bakhshi, Vehicle-bridge interaction and the tuned-mass damper effect on bridges during vertical earthquake excitation, Acta Mechanica 235 (2024) 1379–1394. doi:10.1007/s00707-023-03533-2.

[31] H. Homaei, C. D. Stoura, E. G. Dimitrakopoulos, Extended modified bridge system (embs) method for decoupling seismic vehicle-bridge interaction, Earthquake Engineering & Structural Dynamics 53 (13) (2024) 4054–4075. doi:https://doi.org/10.1002/eqe.4209.

[32] E. G. Dimitrakopoulos, Q. Zeng, A three-dimensional dynamic analysis scheme for the interaction between trains and curved railway bridges, Computers & Structures 149 (2015) 43–60. doi:https://doi.org/10.1016/j.compstruc.2014.12.002.

[33] Y.-B. Yang, Z. Yao, Y. S. Wu, Vehicle-bridge interaction dynamics: with applications to high-speed railways, World Scientific, River Edge, NJ, 2004.

[34] ISO 8608:2016 Mechanical vibration — Road surface profiles — Reporting of measured data, Standard, International Organization for Standardization, Geneva, CH (2016).

[35] D. Simon, Optimal state estimation: Kalman, H_∞, and Nonlinear Approaches, John Wiley & Sons, 2006. doi:10.1002/0470045345.

[36] C. D. Stoura, V. K. Dertimanis, C. Hoelzl, C. Kossmann, A. Cigada, E. N. Chatzi, A Model-Based Bayesian Inference Approach for On-Board Monitoring of Rail Roughness Profiles: Application on Field Measurement Data of the Swiss Federal Railways Network, Structural Control and Health Monitoring 2023 (2023) 8855542. doi:10.1155/2023/8855542.

[37] Y. Li, N. Chen, K. Zhao, H. Liao, Seismic response analysis of road vehicle-bridge system for continuous rigid frame bridges with high piers, Earthquake Engineering and Engineering Vibration 11 (2012) 593–602. doi:10.1007/s11803-012-0144-y.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

OPTIMAL SENSOR PLACEMENT FOR VIBRATION MONITORING OF BRIDGE STRUCTURES

Daniele Giunta[1], Eugenio Chioccarelli[1], Eleni Chatzi[2]

[1] Department of Civil Engineering, Energy, Environment and Materials, Università Mediterranea
Via Rodolfo Zehender - Feo di Vito, Reggio Calabria, 89124, Italy
{daniele.giunta, eugenio.chioccarelli}@unirc.it

[2] Department of Civil, Environmental, and Geomatic Engineering, ETH Zürich
Stefano-Franscini Platz 5, 8093, Zürich, Switzerland
chatzi@ibk.baug.ethz.ch

Abstract

A structural vibration-based monitoring system typically relies on deployment of a network of accelerometers on the monitored structure. Operational modal analysis tools are then commonly used for processing of the acquired data in order to extract the dynamic properties of interest. A crucial issue is to determine a sensor configuration that maximizes the accuracy in identification of the structural dynamic properties, while minimizing the required number of sensors and optimizing their locations. This task is commonly referred to as optimal sensor placement.

This study evaluates the performance of three methods for optimally selecting potential accelerometer locations based on their effectiveness in capturing the dynamic properties of the structure: the modal kinetic energy method, the effective independence method, and the information entropy method. These methods use results from modal analysis performed on a numerical model. To minimize the number of sensors, for each of the analyzed methods, an additional optimization step based on a modal frequency metric is adopted. This includes a numerical simulation of an ambient vibration test performed by exciting the finite element model of the structure with white Gaussian noise. The simulated acceleration responses are processed using the stochastic subspace identification with unweighted principal components technique.

The resulting procedures are applied to the case study of a prestressed concrete bridge, demonstrating that the optimized sensor network slightly depends on the adopted ranking method.

Keywords: Structural health monitoring, Operational modal analysis, Modal properties, Accelerometers, Optimization

1 INTRODUCTION

Structural health monitoring (SHM) has increasingly become essential for managing existing infrastructure, ensuring safety, reliability, and serviceability throughout their lifespan. An essential aspect of SHM is accurately assessing the dynamic behavior of structures through operational modal analysis (OMA), a methodology that extracts structural modal properties from measured vibration responses under operational conditions. A critical challenge in OMA relates to optimal sensor placement (OSP), which aims to strategically position sensors on a structure to maximize the accuracy of dynamic property identification. Different methodologies have been proposed in the literature for addressing the OSP problem. The modal kinetic energy (MKE) method [1], the effective independence (EFI) method [2] and the information entropy (IE) method [3] are among the most widely used methods [4,5]. All these OSP methods are based on the modal displacements derived from a numerical model of the studied structure, such as a finite element (FE) model. The MKE method prioritizes sensor locations where the structure exhibits maximum kinetic energy, thus enhancing signal quality for mode shape identification. The EFI method aims at maximizing the linear independence between the target mode shapes of the dynamic identification by ranking sensor positions based on their contributions to the Fisher information matrix. The IE method selects sensor positions to minimizing the uncertainty of structural parameters by means of entropy estimation.

Since all the OSP methods rank the candidate sensor locations but do not identify, by themselves, an optimal number (i.e., minimal number for the required accuracy) of sensors, a procedure to design a sensors network characterized by an optimized number and location of the sensors was discussed in previous works [6,7]. The procedure, that is identified hereafter as optimization of sensors network (OSN), was formulated coupling the ranking of sensor locations from the EFI method with a frequency metric, the cumulative frequency error (CFE), computed via a numerical simulation of the ambient vibration test of the monitored structure. Such a simulation was carried out through a linear dynamic analysis of the FE model. The simulated accelerations were then used to retrieve the modal frequency of the structure via a frequency domain decomposition (FDD) technique [8].

This work aims to extend the study of [6] by: (i) comparatively analyzing the performances of three different OSP methods, i.e., EFI, MKE, and IE, to be used for OSN; (ii) replacing the FDD technique with the stochastic subspace identification with unweighted principal components (SSI-UPC) technique [9] for the CFE computation.

In the following sections, the paper first provides a brief overview of the theoretical background of the considered OSP methods. Subsequently, the developed sensor network optimization algorithm, which is designed to efficiently determine the number of sensors based on frequency metrics, is detailed. Then, the case study structure, a prestressed concrete bridge, and its FE model are described. The results obtained from applying the optimization algorithm to the case study structure are presented, including a comparative analysis of the different OSP methods. Finally, the main findings of the research are summarized.

2 OPTIMAL SENSOR PLACEMENT METHODS

Among the different OSP techniques, the MKE, EFI, and IE methods were selected to evaluate the performance of the proposed optimization algorithm, as each uses distinct criteria to determine optimal sensor locations. The first step for implementing OSP methods is to create a FE model of the structure. Three input parameters are then defined: (i) a set of N target modes, obtained by performing a modal analysis of the model and representing the modes of interest for identification; (ii) a set of S candidate sensor locations on the structure; (iii) the number of sensors T to be deployed in the final configuration. The number of candidate locations S must

exceed T, which in turn must be at least equal to N, as identifying a set of N modes requires at least $T = N$ sensors. All the sensors are considered as uniaxial accelerometers, meaning that each i-th sensor defines both its position and its measurement direction. This section outlines the analytical formulations of the considered OSP methods, while their typical workflow and how they are integrated into the proposed algorithm are presented in Section 3.

2.1 The modal kinetic energy method

The MKE method aims to enhance the modal information captured through sensor placement by positioning sensors where the kinetic energy contribution to the target modes is maximized. This sensor placement is expected to provide a higher signal-to-noise ratio and a more accurate mode shape identification [1,10]. The kinetic energy matrix $[MKE]$ is defined in Eq. (1):

$$[MKE] = [\Phi] \odot ([M] \cdot [\Phi]), \tag{1}$$

where $[\Phi]$ is the mode shape matrix of dimensions $[S \times N]$, containing the displacement of the i-th sensor ($i = 1, \dots S$) when the j-th mode ($j = 1, \dots N$) is activated, $[M]$ (dimensions $[S \times S]$) is the mass matrix that collects the values of the masses associated to each monitored node, $\odot$ denotes the element-wise product, and $\cdot$ the matrix product. Each element MKE_{ij} of the kinetic energy matrix is a scalar value that quantifies the contribution of the i-th sensor to the kinetic energy of the j-th mode. To assess the effectiveness of each sensor at maximizing the kinetic energy of all the target modes, MKE_i can be computed as per Eq. (2):

$$MKE_i = \sum_{j=1}^{N} MKE_{ij}. \tag{2}$$

The MKE_i is a ranking index for each sensor placement; higher values indicating more informative sensor positions for mode shapes identification.

2.2 The effective independence method

The EFI method is based on the computation of the Fisher Information Matrix $[FIM]$, which has dimensions $[N \times N]$ and is obtained as defined in Eq. (3):

$$[FIM] = [\Phi]^T [\Phi]. \tag{3}$$

The effective independence value, EFI_i, for the i-th sensor is defined as per Eq. (4):

$$EFI_i = \{\varphi\}_i \cdot [FIM]^{-1} \cdot \{\varphi\}_i^T, \tag{4}$$

where $\{\varphi\}_i$ is a row vector of dimensions $[1 \times N]$ containing the modal displacements of the i-th sensor for each of the N modes (i.e., the i-th row of the $[\Phi]$ matrix) and $[FIM]^{-1}$ is the inverse of the Fisher information matrix. The EFI_i index measures the contribution of the i-th sensor to the linear independence of the target mode shapes [2]. It ranges between 0 and 1, with higher values representing a greater contribution to modal identification.

2.3 The information entropy method

The IE approach measures the uncertainty associated with the system parameters $\{\theta\} \in \mathbb{R}^{N_\theta}$ in a Bayesian framework. In the OSP problem, the parameter vector $\{\theta\}$ may comprise the modal displacements (or other model parameters). N_θ is a scalar indicating the number of parameters to be estimated. The IE method aims to minimize the Shannon entropy of $\{\theta\}$, as lower entropy corresponds to reduced uncertainty in the system parameter estimation [3]. The information entropy IE for a given sensor configuration is defined in Eq. (5):

$$IE(\{\theta_0\}) = \frac{1}{2} N_\theta \ln(2\pi) - \frac{1}{2} \ln\left(\det([FIM(\{\theta_0\})])\right), \quad (5)$$

in which $\{\theta_0\}$ represents the optimal value of the parameter vector $\{\theta\}$ and the matrix $[FIM(\{\theta_0\})]$ is the Fisher information matrix with dimensions $[N_\theta \times N_\theta]$. For dynamic identification, which is the focus of this work, when the parameters $\{\theta\}$ correspond to the mode shapes, N_θ is set to N (i.e., the number of target modes), and the $[FIM]$ is computed as described in Eq. (3). Both the EFI method and the IEI method rely on the $[FIM]$ matrix since it is an indicator of the information contained in the data.

3 SENSOR NETWORK OPTIMIZATION ALGORITHM

This section outlines the proposed sensor network optimization algorithm. First, the traditional backward sequential sensor placement (BSSP) algorithm, which serves as the basis for the proposed approach, is described. Then, the OSP algorithm, defined to optimize sensor numbers using a frequency-based metric and including a numerical simulation of ambient vibration test, is presented.

3.1 The traditional OSP algorithm with BSSP

The OSP methods described in Section 2 can be implemented using a BSSP algorithm. In BSSP, the procedure begins by considering sensors deployed at all S candidate locations and removes one sensor per iteration. The sensor providing the least contribution to identifying the target modal parameters, based on the considered OSP method, is removed at each step. For the MKE and the EFI methods, the sensor to be removed is the sensor with the lowest value of the related index, i.e., the MKE_i and the EFI_i (see Eq. (2) and Eq. (4), respectively). For the IE method, the information entropy IE is computed as per Eq. (5) for the full set of S sensors. Rather than assigning a single-sensor index, the IE of all possible configuration subsets obtained by removing one sensor are evaluated, and the configuration with the lowest IE value is then retained and used to identify the sensor to be removed at each iteration. The iteration count is denoted as k. The procedure continues until the remaining sensors reach the desired number T, which occurs when $k = S - T + 1$. The general flowchart for this procedure is illustrated in Figure 1a.

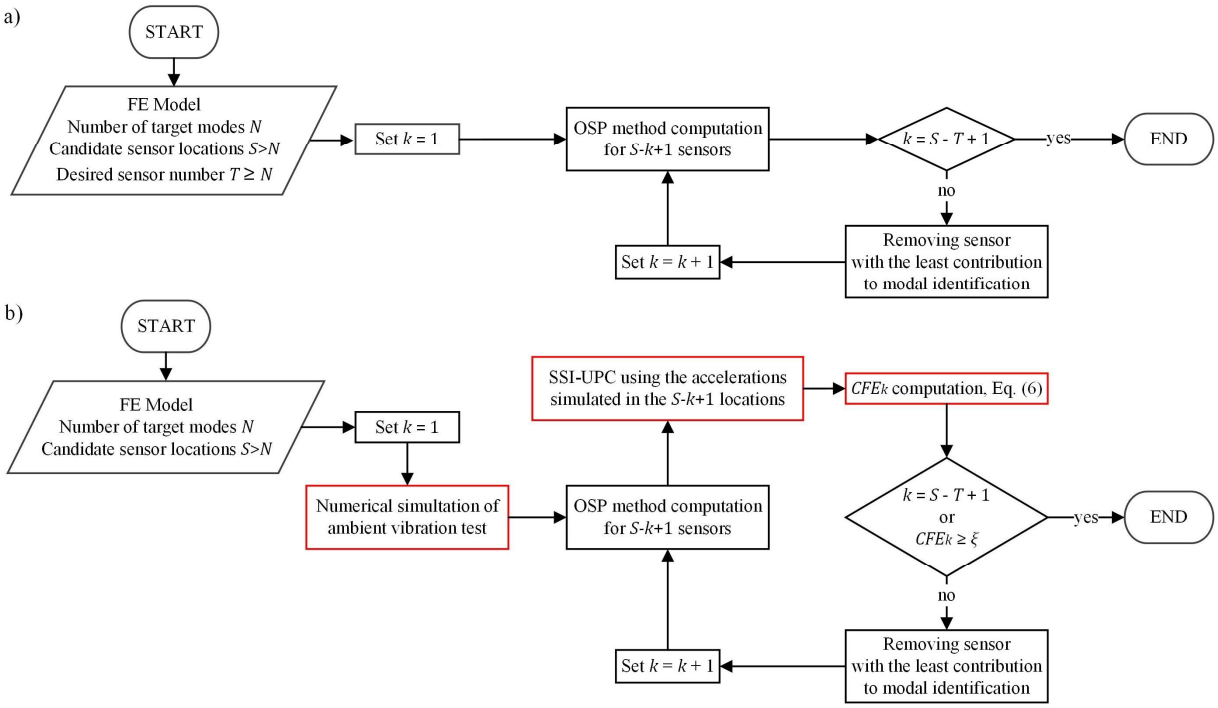


Figure 1. Flowcharts of OSP algorithms: (a) OSP methods using a BSSP algorithm (a); (b) OSN algorithm implementing numerical simulation of ambient vibration test.

3.2 The frequency-based OSN algorithm

The proposed algorithm integrates the outcomes from numerical modal analysis and operational modal analysis simulation based on the FE model. In Figure 1b, the flowchart of the algorithm is shown; red boxes highlight the steps introduced by this procedure compared to the BSSP procedure implemented for the traditional OSP methods. First, a FE model of the structure is realized and N target modes and S candidate sensor location are selected. Afterwards, a numerical simulation of ambient vibration test, employing white Gaussian noise as an excitation, is performed to replicate the vibration conditions expected during actual structural monitoring [11]. Such an excitation is applied along the three main directions at the base of the structure (exciting the whole model) and on the deck (emulating the random traffic excitation). From these simulations, acceleration time-histories at candidate sensor locations are generated. Then, the iterative procedure begins, setting the iteration count $k = 1$, and for each iteration of the algorithm, the effectiveness of each sensor location is assessed using an OSP method. Locations with lower rankings are iteratively excluded based on their limited contribution to the target modes identification.

The selection of an optimal sensor number is performed by introducing the CFE metric, quantifying differences between the numerically identified modal frequencies and the target frequencies from modal analysis [6,7]. The cumulative frequency error CFE_k, defined in Eq. (6), is computed as the sum of the relative difference between the frequency of the j-th mode identified by the numerical simulation, say $f_{DI,j}$, and the target frequency of the same mode, say $f_{FE,j}$ (obtained by the modal analysis of the FE model):

$$CFE_k = 100 \cdot \sum_{j=1}^{N} \frac{\left| f_{DI,j} - f_{FE,j} \right|}{f_{FE,j}}. \tag{6}$$

The iterative removal of sensor locations continues until CFE_k exceeds an acceptance threshold ξ or until the number of sensors equals the number of target modes N. Since factors

such as monitoring system efficiency, identification objectives, number of target frequencies influence ξ, a case-specific approach should be considered for the definition of the threshold. Analyzing the variation of CFE_k with the sensor number can help to determine an appropriate threshold ξ (see Section 5).

In the previous application of this OSN algorithm discussed in [6], the FDD technique was used to retrieve the simulated modal frequencies. Herein, the simulated time series are analyzed using a subspace identification technique, the SSI-UPC technique. The SSI-UPC technique identifies the dynamic properties of structures by estimating a state-space model directly from the measured output data [12,13]. In contrast to frequency domain approaches, it operates entirely in the time domain. A key aspect of SSI-UPC is selecting an appropriate model order, representing the dimensions of the state-space model, which significantly influences the accuracy and reliability of the identified modal parameters [14]. The SSI-UPC was selected as identification technique due to its capability to estimate higher-frequency modes that were not identified using the peak-picking approach within the FDD. Capturing these higher-frequency modes is relevant, as they may manifest in dynamic identification of real structures, including the case-study structure discussed here. However, adopting SSI-UPC involves a trade-off in terms of computational time and effort. This issue is further discussed in Section 5.

As experimentally demonstrated in [6], the optimal sensor configuration identified by the presented OSN algorithm provides a sensor network capable of accurate natural frequency identification with the minimum number of required sensors.

4 CASE STUDY STRUCTURE AND FINITE ELEMENT MODEL

The selected structure is the Volto Santo Viaduct, a prestressed concrete bridge whose detailed geometry and actual monitoring data are reported in [6]. The side view of the bridge is presented in Figure 2 via a satellite picture. The bridge, located in Naples, Italy, consists of four 18.00 m simply supported decks supported by two abutments and three sets of circular-section reinforced concrete columns. Each span is composed of nine precast prestressed girders connected by crossbeams and topped with a cast-in-place reinforced concrete slab.

Figure 2. Satellite side view of the Volto Santo viaduct (satellite views © 2022 Google).

A three-dimensional FE model was set up using the FE software SAP2000 v24 [15] to represent one of the four typical spans of the viaduct, replicating the actual simply supported boundary conditions of the deck. The model discretizes the deck by combining frame elements for the girders and crossbeams, and shell elements for the reinforced concrete slab. The columns and bent caps are modeled via frame elements, with fixed supports at the base nodes. The model is shown in Figure 3.

The modal analysis of the FE model was performed to investigate the dynamic behavior of the structure and identify the target modes of interest for the OSN application. In Table 1 the modal frequencies and the description of the mode shapes are reported. The first five modes dominate the global response (e.g., the rigid translations of the deck, the vertical bending and torsion of the deck, and the global torsional mode). The remaining modes describe higher order bending and torsion of the deck.

The overall FE model captures both global and local modes of vibration that can be excited under ambient conditions. Thus, the model is used to: (i) compute the modal displacements that feed into the EFI, MKE, and IE methods; (ii) simulate the ambient vibration test with a linear dynamic analysis in order to generate acceleration time-histories for the CFE_k computation within the OSP procedure.

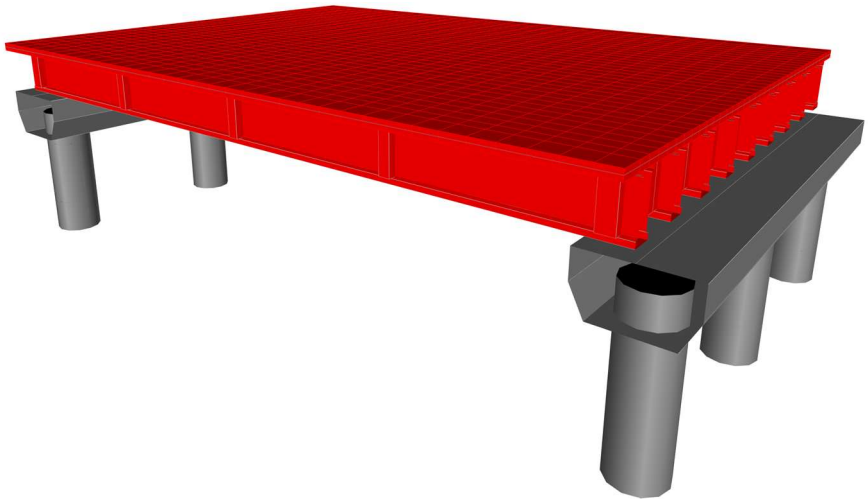


Figure 3. Tridimensional view of the FE model of the structure.

Table 1. Modal frequencies, f_{FE}, and description of the mode shapes obtained from the FE model.

Mode N.	f_{FE} (Hz)	Description
1	4.18	1st longitudinal translational global mode
2	6.15	1st vertical bending mode of the deck
3	6.90	1st torsional mode of the deck
4	6.79	1st transversal translational global mode
5	11.22	1st torsional global mode
6	15.92	2nd torsional mode of the deck
7	21.57	2nd vertical bending mode of the deck
8	21.85	3rd torsional mode of the deck
9	26.74	1st transversal bending mode of the deck
10	27.98	4th torsional mode of the deck

5 OPTIMIZED MONITORING SYSTEMS

To apply the OSN algorithm, $N = 9$ target modes were selected: modes 1 to 4 and 6 to 10 of Table 1. These modes were identified with a dynamic identification carried out on the actual structure. Mode 5 is neglected as it was not experimentally identified. It has to be noted that only the first four modes were selected as target modes in the previous application [6], as these were the only identifiable modes using the FDD technique. However, the adoption of the SSI-UPC technique in this work allowed the additional identification of the higher frequency modes, i.e., modes 6 to 10, thus, expanding the target mode number.

The candidate sensor locations were defined on the FE model, with a total of $S = 333$ sensors, corresponding to 111 nodes. At each node, it is considered that three uniaxial sensors are placed to capture the spatial acceleration of the monitored points. These nodes are distributed over the whole structure. In Figure 4 the axis of the columns and the beams composing the deck are sketched. The red dots are the selected nodes for the computation of the OSN procedure.

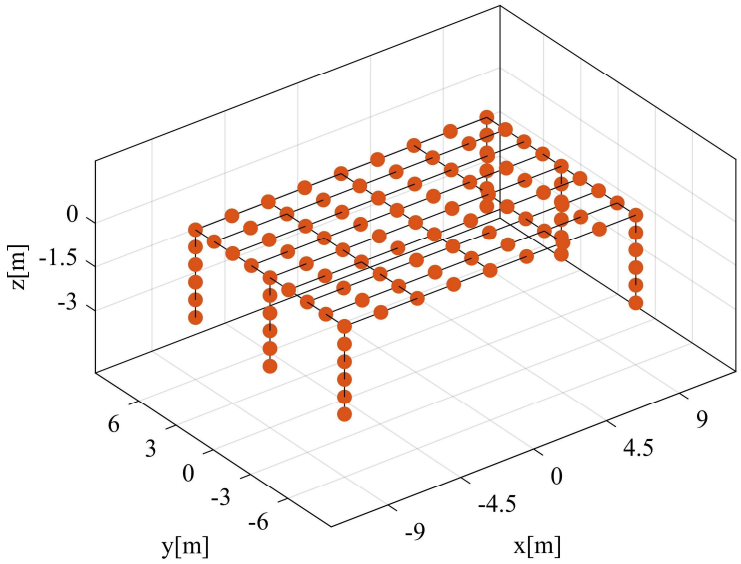


Figure 4. Candidate sensor locations defined on the FE model.

A linear dynamic analysis of the numerical model was performed by exciting the structure with white Gaussian noise. This analysis provided acceleration time series recorded at each candidate sensor with a sampling frequency of 100 Hz. The sampling frequency was chosen to ensure that the corresponding Nyquist frequency (50 Hz) exceeds the highest target frequency (Mode 10 – 27.98 Hz), allowing for accurate estimation of all modes up to the Nyquist limit. For each OSP technique (i.e., MKE, EFI, and IE), $(S - N + 1)$ sensor configurations were defined through the iterative procedure presented in Figure 1b, by sequentially removing sensor locations with the least contribution to modal identification based on the respective rankings (in the case of MKE, Eq. (1) was applied assuming the same mass value associated to each monitored node). As a result, a total of $3 \times (S - N + 1)$ sensor configurations were obtained. For each method and, thus, for each configuration, the cumulative frequency error CFE_k was computed. For this purpose, the time series were analyzed using the SSI-UPC technique implemented in the ARTeMIS Modal software [16]. The duration of the time series was 600 s, a sufficient duration for performing the considered identification technique. The SSI-UPC was performed for the first 32 sensor configurations obtained by the OSP rankings, i.e., with sensors ranging from 9 to 40 sensors. Thus, a total of 96 dynamic identifications were carried out. The lower bound (9 sensors) represents the minimum sensor number required to identify the 9 target modes, while the upper bound (40 sensors) was chosen as a reasonable limit to maintain feasibility in real applications of the OSN algorithm and to limit computational cost and processing time.

5.1 The SSI-UPC technique application within the OSP algorithm

This subsection discusses how model order selection influences the CFE estimation when applying the SSI-UPC technique for the numerical dynamic identification. The SSI-UPC dynamic identification was performed considering two different approaches: (i) for each

considered sensor configuration, the model order providing the best solution in terms of frequency identification (with respect to the target modal frequencies) was selected; (ii) the order of the model used for frequency identification was fixed (set at a model order equal to 60 for this application, which was the maximum model order set in the first approach). Approach (i) aligns with the strategy commonly adopted in practical dynamic identification using subspace-based techniques [14]. Since the model order cannot be a-priori specified and should be at least twice the number of modes to be identified, an overestimation is typically introduced. As a result, spurious (i.e., numerical) modes appear in the model alongside the real modes. This overestimation often yields a more accurate estimation of the modes of interest. This approach introduces some randomness, as the model order selection remains partly arbitrary. On the other hand, approach (ii) removes this randomness by fixing the model order and speeds up the identification process. However, it may not always adapt well to changes in the measured data, thus potentially reducing the accuracy of frequency identification. Figure 5a and Figure 5b plot the CFE value as a function of the sensor number for the two approaches, using different colors and markers for each OSP method. In approach (i) (see Figure 5a), for all the considered OSP methods, the CFE has a clear trend, as it starts at approximately 25% for 9-sensor configurations, decreases to about 15% up to 23 sensors, and remains constant as the sensor number further increases. Beyond 24 sensors, MKE configurations fail to identify mode 1 frequency (translational longitudinal mode) due to excessive presence of sensors in the vertical direction, reducing sensitivity to this horizontal mode. This limitation is indicated in the plot with grey markers for MKE configurations unable to estimate the frequency of mode 1. In Figure 5b, related to approach (ii), the obtained CFE does not show a clear trend increasing the sensor number. This behavior suggests that increasing the number of monitored degrees of freedom, without adjusting the identification parameters, does not always lead to a significant improvement in frequency identification accuracy. Instead, with approach (i), the adaptive nature of the method effectively uses additional data, gradually reducing uncertainty. However, once a stable information level is reached, adding more sensors no longer provides meaningful additional insights. These results conceptually align with findings in [17], discussing that classical Bayesian model updating cannot explicitly quantify modeling errors, since these are lumped together with other sources of uncertainty such as measurement noise and parameter variability. As a result, if modeling errors are significant, additional data might not provide meaningful new insights unless these errors are separately characterized and accounted for. Although the presented application is focused on OSN rather than Bayesian updating, a similar effect is observed: simply increasing the sensor number (monitored degrees of freedom) without adapting identification parameters (such as model order) does not always lead to improved frequency estimation accuracy. This aspect highlights how adjusting identification parameters (approach (i)) benefits frequency estimation in this OSN application, as shown by the CFE trends in Figure 5a and Figure 5b. Thus, the remaining part of this application will refer to Figure 5a.

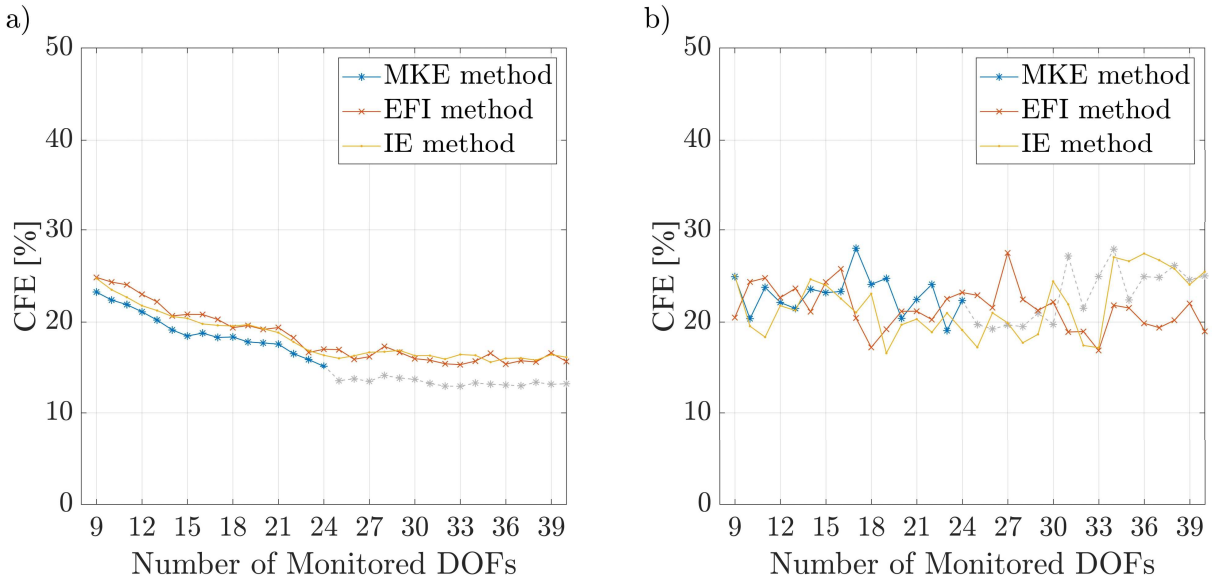


Figure 5. Cumulative frequency error varying the number of sensors: (a) adaptive model order selection – approach (i); (b) fixed model order – approach (ii).

5.2 OSP methods comparison

Figure 5a shows that all three OSP methods exhibit similar trends in CFE, as they distribute sensors in a comparable manner as the number of monitored degrees of freedom increases. In addition to the previously mentioned variations in slope at 9 and 23 sensor configurations, another significant feature of the plot in Figure 5a occurs at 15 sensors, where a reduction in the slope of the curves can be observed. In Figure 6a, Figure 6b, and Figure 6c, the 15-sensor configurations for the MKE, the EFI, and the IE methods are shown, respectively. In each figure, arrows indicate sensors measuring along the horizontal directions and dots denote sensors recording along the vertical direction. As depicted, although the sensor placement carried out with the three methods is similar, distributing sensors along the deck, the MKE method places only vertical sensors as it tends to maximize the kinetic energy of the considered modes. Since 6 out of 9 target modes are vertical modes of the deck, the method places sensors in the vertical direction at nodes where modal displacements are largest. Hence, sensor configurations with a low number of monitored degrees of freedom obtained using the MKE method are not able to properly capture the mode shapes of horizontal modes, such as Mode 1, Mode 4, and Mode 8 described in Table 1, as no sensors are placed along the horizontal directions. However, mode shape reconstruction was not taken into account in the OSN procedure, as the optimization was performed considering the target modal frequencies as objective through the CFE metric. As suggested in [4,5], trade-offs between different OSP metrics should be considered in an optimization procedure, e.g., including a metric based on the mode shape comparison. Nevertheless, frequency identification of these modes was still achieved using the sensor configurations derived from the MKE method, allowing to compare the results in terms of CFE between this method and the EFI and IE methods. On the other hand, the EFI and IE methods prioritize maximizing the independence and information content of all the target mode shapes. As a result, they tend to distribute sensors ensuring that all mode shapes can be captured. Such configurations allow the reconstruction of all the nine target mode shapes.

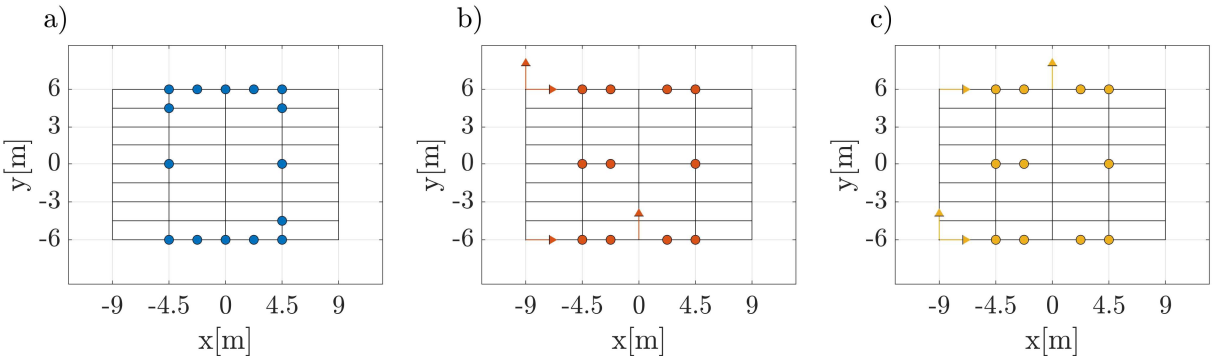


Figure 6. 15-sensor configuration obtained by using (a) the MKE, (b) the EFI and (c) the IE methods.

The optimal configurations (i.e., those minimizing the CFE value and allowing the target frequency identification) should be selected at 33 sensors (CFE = 15.38%) for the EFI method, 35 sensors (CFE = 15.65%) for the IE method, and 24 sensors (CFE = 15.19%) for the MKE method. However, all methods converge to comparable CFE values once the sensor number reaches about 23-24. Thus, selecting 23-sensor configuration as optimal can be considered a reasonable compromise for this application. Figure 7 illustrates the resulting optimal sensor configurations obtained using the three OSP methods, showing that while orientation preferences differ, the final configurations are similar and provide analogous frequency identification accuracy.

This comparative study extends the optimization approach introduced in [6], by verifying that the CFE metric remains robust across different OSP methods and by introducing the SSI-UPC technique as a valuable tool for extracting modal frequencies within the numerical dynamic identification process. The experimental validation of the applied OSN algorithm was already discussed in [6].

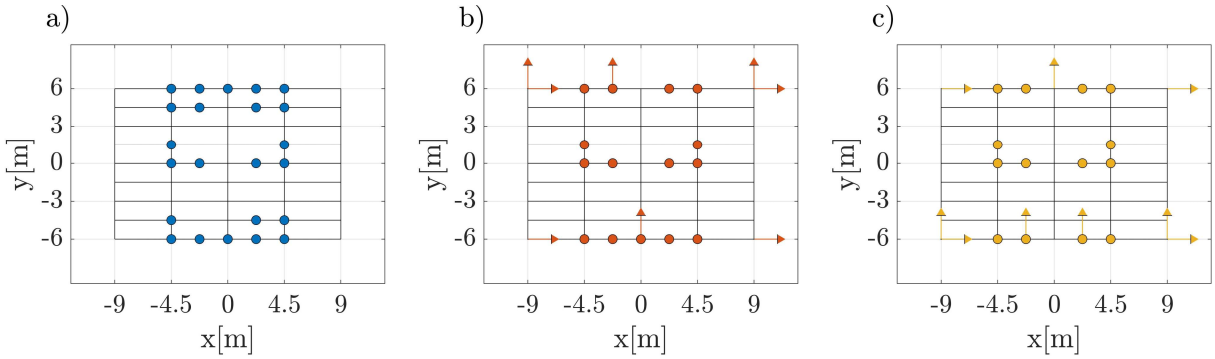


Figure 7. Optimal sensor configurations for different OSP methods: (a) MKE; (b) EFI; (c) IE.

6 CONCLUSIONS

In this study, three well-known OSP methods, i.e., the MKE, the EFI, and the IE methods, were compared using an optimization algorithm based on the CFE metric to an existing bridge. This work extends previous investigations by implementing a subspace identification technique, the SSI-UPC technique, within the OSN algorithm, providing a more robust and reliable assessment of the modal frequencies. The main findings of this work can be summarized as follows:

- All three OSP methods achieved similar sensor configurations and converged to comparable CFE values for the selected target modes. While the MKE method tends to place sensors in the vertical direction (reflecting the dominance of vertical modes),

the EFI and IE methods distribute sensors more evenly across both horizontal and vertical directions.

- The adoption of the SSI-UPC identification technique improved frequency identification accuracy compared to the frequency-domain identification method used in a previous study. This improvement allowed for an increased number of target modes as input of the OSN algorithm, matching the number of actual modes identified through an OMA conducted on the real case study structure. However, the reliance on numerical modeling and computational resources represents a limitation.
- Across all methods, the CFE stabilized around a sensor number of 23–24, indicating that beyond this number, additional sensors contributed minimal improvements to frequency identification. Hence, 23 sensors emerged as a reasonable compromise between accuracy and feasibility.

In conclusion, while numerical simulations and computational demands remain a practical limitation, the results highlight the robustness of the CFE-based OSN algorithm.

7 ACKNOWLEDGEMENTS

The presented work has been partially supported by the project "DS2: Digital Smart Structures", funded by INAIL (Italian Workers' Compensation Authority), BRIC 2021 and partially carried out as part of the activities envisaged by the Agreement between the High Council of Public Works (CSLLPP) and the ReLUIS Consortium implementing Ministerial Decree 578/2020 and Ministerial Decree 240/2022. Prof. Chatzi wishes to acknowledge the MSCA Staff Exchanges 2021 project ReCharged (Grant ID: 101086413). This project has received funding from the Horizon Europe Programme under the Marie Skłodowska-Curie Staff Exchanges Action (GA no. 101086413), co-funded by the UK Research & Innovation and the Swiss State Secretariat for Education, Research & Innovation. The contents of this paper represent the authors' ideas and do not necessarily correspond to the official opinion and policies of the funding entities.

REFERENCES

[1] G. Heo, M.L. Wang, D. Satpathi, Optimal transducer placement for health monitoring of long span bridge. *Soil Dynamics and Earthquake Engineering*, **16**, 495–502, 1997. https://doi.org/10.1016/S0267-7261(97)00010-9.

[2] D.C. Kammer, Sensor placement for on-orbit modal identification and correlation of large space structures. *Journal of Guidance, Control, and Dynamics,* **14**, 251–259, 1991. https://doi.org/10.2514/3.20635.

[3] C. Papadimitriou, Optimal sensor placement methodology for parametric identification of structural systems. *Journal of Sound and Vibration,* **278**, 923–947, 2004. https://doi.org/10.1016/j.jsv.2003.10.063.

[4] C. Leyder, V. Dertimanis, A. Frangi, E. Chatzi, G. Lombaert, Optimal sensor placement methods and metrics–comparison and implementation on a timber frame structure. *Structure and Infrastructure Engineering,* **14**, 997–1010, 2018. https://doi.org/10.1080/15732479.2018.1438483.

[5] M. Meo, G. Zumpano, On the optimal sensor placement techniques for a bridge structure. *Engineering Structures,* **27**, 1488–1497, 2005. https://doi.org/10.1016/j.engstruct.2005.03.015.

[6] E. Chioccarelli, D. Giunta, A. De Angelis, A. Bilotta, Discussing optimal sensor placement for ambient vibration test of an existing bridge. *Engineering Structures,* **310**, 2024. https://doi.org/10.1016/j.engstruct.2024.118000.

[7] P. Pachón, M. Infantes, M. Cámara, V. Compán, E. García-Macías, M.I. Friswell, R. Castro-Triguero, Evaluation of optimal sensor placement algorithms for the Structural Health Monitoring of architectural heritage. Application to the Monastery of San Jerónimo de Buenavista (Seville, Spain). *Engineering Structures*, **202**, 2020. https://doi.org/10.1016/j.engstruct.2019.109843.

[8] R. Brinker, L. Zhang, P. Anderson, Modal identification of output-only systems using frequency domain decomposition. *Smart Materials and Structures*, **10**, 2001. https://doi.org/10.1088/0964-1726/10/3/303.

[9] P. Van Overschee, B. De Moor, *Subspace identification for linear systems*, Springer US, 1996. https://doi.org/10.1007/978-1-4613-0465-4.

[10] M. Papadopoulos, E. Garcia. Sensor placement methodologies for dynamic testing. *AIAA Journal*, **36**, 256–263, 1998. https://doi.org/10.2514/2.7509.

[11] V. Pepe, A. De Angelis, M.R. Pecce, Numerical simulation of ambient vibration tests: A case study. *7th International Conference on Advances in Experimental Structural Engineering,* EUCENTRE, Pavia, Italy, 2017. https://doi.org/10.7414/7aese.T5.200.

[12] R. Brinker, C. Ventura, *Introduction to Operational Modal Analysis*. John Wiley & Sons, Ltd, 2015. https://doi.org/10.1002/9781118535141.

[13] K.S. Arun, S.Y. Kung, Balanced approximation of stochastic systems. *SIAM Journal on Matrix Analysis and Applications*, **11**, 807–819, 1990. https://doi.org/10.1137/0611003.

[14] C. Rainieri, G. Fabbrocino, *Operational modal analysis of civil engineering structures: An Introduction and Guide for Applications*. Springer, 2014. https://doi.org/10.1007/978-1-4939-0767-0.

[15] Computers & Structures inc., Structural and earthquake engineering software. SAP2000 Version 24, (2016).

[16] Structural Vibration Solutions A/S. ARTeMIS Modal Pro Version 6.0, (2019).

[17] M. Song, I. Behmanesh, B. Moaveni, C. Papadimitriou, Modeling error estimation and response prediction of a 10-story building model through a hierarchical bayesian model updating framework. *Frontiers in Built Environment*, **5**, 2019. https://doi.org/10.3389/fbuil.2019.00007.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

COMBINING EXPERIMENTAL DATA AND NUMERICAL MODELING TO ANALYZE TRAIN-STRUCTURE INTERACTION ON THE HISTORIC SAN MICHELE BRIDGE (ITALY, 1889)

Lorenzo Ermolli[1], Michele Guerini[1], Charikleia D. Stoura[2,3], Rosalba Ferrari[1], Vasilis K. Dertimanis[2], Eleni N. Chatzi[2], Egidio Rizzi[1]

[1] Dipartimento di Ingegneria e Scienze Applicate, Università degli studi di Bergamo
viale G. Marconi 5, I-24044 Dalmine (BG), Italy
l.ermolli@studenti.unibg.it, michele.guerini@unibg.it, rosalba.ferrari@unibg.it, egidio.rizzi@unibg.it

[2] Department of Civil, Environmental and Geomatic Engineering, ETH Zürich
Stefano-Franscini-Platz 5, 8093 Zürich, Switzerland
charikleia.stoura@ibk.baug.ethz.ch, v.derti@ibk.baug.ethz.ch, chatzi@ibk.baug.ethz.ch

[3] Department of Mechanical Engineering, Politecnico di Milano
via La Masa 1, 20156 Milan (MI), Italy
charikleia.stoura@polimi.it

Abstract. *As part of ongoing efforts to study and preserve the San Michele Bridge, a remarkable example of 19th-century iron architecture and a true symbol of Italian industrial archaeology, an experimental campaign was conducted on it in May 2024. Completed in 1889 by the Società Nazionale delle Officine di Savigliano (SNOS), the San Michele Bridge spans 266 meters across the Adda River, connecting the towns of Paderno d'Adda (Lecco bank) and Calusco d'Adda (Bergamo bank) in Lombardy, northern Italy. Its innovative design features a doubly built-in parabolic arch and an upper continuous beam, supporting both railway and road traffic. During the experimental campaign, accelerometers were deployed both on the bridge structure and on a crossing railway vehicle, allowing for the simultaneous collection of acceleration data from both systems, aiming at investigating the dynamic behavior of the bridge under vehicle-bridge interaction effects, which constitutes a significant excitation mechanism for the bridge structure. This study presents a preliminary dynamic analysis on the bridge that first models the railway vehicle crossings as moving loads through a finite element model of the bridge, enabling numerical simulations of the system's dynamic response. Then, it foresees a comparison of numerical predictions with experimental data, in order to refine the FEM model and gain insight into the dynamic interaction of the coupled vehicle-bridge system under operating conditions. This research is part of a broader initiative to establish a comprehensive Structural Health Monitoring (SHM) platform for the San Michele Bridge. Such a system would enhance long-term monitoring, facilitate early damage detection, and support the preservation and safety of this historic viaduct. The findings highlight the importance of integrating advanced experimental techniques with numerical modeling to safeguard valuable, historic infrastructure.*

Keywords: Railway bridge, FEM model, Moving load, Dynamic interaction, Modal updating.

1 INTRODUCTION

The San Michele Bridge was built during the 19th century to connect the provinces of Lecco and Bergamo, otherwise separated by the Adda River, between Paderno d'Adda (LC) and Calusco d'Adda (BG). This structure is a remarkable example of an arch bridge, consisting of a continuous box girder supported by nine bearings, four of which directly resting on the 150-meter-span arch. Given its historical and still present territorial significance, the preservation of such iconic infrastructure is of paramount importance. To this end, the creation of a long-term Structural Health Monitoring (SHM) platform for the Paderno d'Adda Bridge was proposed in [5]. The project also aims to extend the knowledge acquired during this phase of the study on the San Michele Bridge to other similar infrastructures. The SHM platform to be developed will be based on a digital twin of the structure that may facilitate early damage detection, improve predictive maintenance, and most importantly contribute to the preservation of this significant historical structure.

As a starting point for the development of this project, an experimental campaign was conducted in May 2024 as a joint effort between research groups from the University of Bergamo and ETH Zurich, aiming at investigating the dynamic behaviour of the bridge under vehicle-structure interaction effects. The experimental campaign was carried out without closing the bridge to regular transit and taking advantage of the nighttime when train traffic is usually stopped and road traffic very limited. This permitted to have time and space to set up all the instrumentation. Acceleration data were recorded under both ambient and loaded conditions, the latter involving multiple crossings of a special railway convoy at various speeds. To simultaneously capture the dynamic response of the bridge and the convoy, a data acquisition system equipped with accelerometers was installed on both the bridge and the train.

The primary goal of the study here presented is to investigate the effect of a moving load on the structure by foreseeing a comparison among simulations carried out numerically with a dedicated FEM model of the structure with real data acquired during the experimental campaign. The adopted San Michele Bridge FEM model is based on that previously developed by the UniBG Research Group [5, 6] and adapted for the dynamic simulations of multiple moving loads [10, 18, 20], representing the convoy used during the experimental campaign. Due to the location of the sensors during the campaign, a part of the bridge structure rather than the entire bridge model is used to perform the aforementioned simulations. The subpart of interest includes the upper continuous beam of the bridge where the trains travel, within it. To simulate the effects of the supporting substructures (arch and piers), which do not form part of the substructure of interest, elastic boundary conditions are introduced, in the form of springs, to be calibrated.

This study is presented as follows: in Section 2 the experimental campaign is described, providing some information about the sensor positioning and the rail-vehicles used to perform dynamic tests on the structure. Following Section 3 presents the FEM model of the Paderno d'Adda Bridge, including the substructuring process, the initial guess procedure of the spring stiffnesses introduced at the level of the upper continuous beam, and lastly the moving load modelling of the train convoy. Then, Section 4 presents preliminary numerical results of dynamic simulations under moving loads. Lastly, Conclusions present the main findings of this study.

2 EXPERIMENTAL CAMPAIGN

In May 2024, an extensive experimental campaign was carried out during two overnight sessions, taking advantage of the usual suspension of train traffic. Acceleration response signals to ambient excitation, road traffic, and dedicated rail-vehicle passage were collected, with the aim of gathering information on the structural behaviour under traffic conditions of road and train traffic and investigating the vehicle-bridge interaction in further analyses.

To acquire acceleration signals, 12 uniaxial seismic accelerometers *Model 731A*, *Wilcoxon* were installed on the bridge at the railway level inside the upper continuous beam, where rail-vehicles transited during controlled tests. Accelerations were measured in both transverse horizontal and vertical directions. Details about sensor positioning are illustrated in Figure 1. Some sensors were positioned at the level of the piers, while others were installed slightly offset from them, except for a couple of sensors placed at the key of the arch.

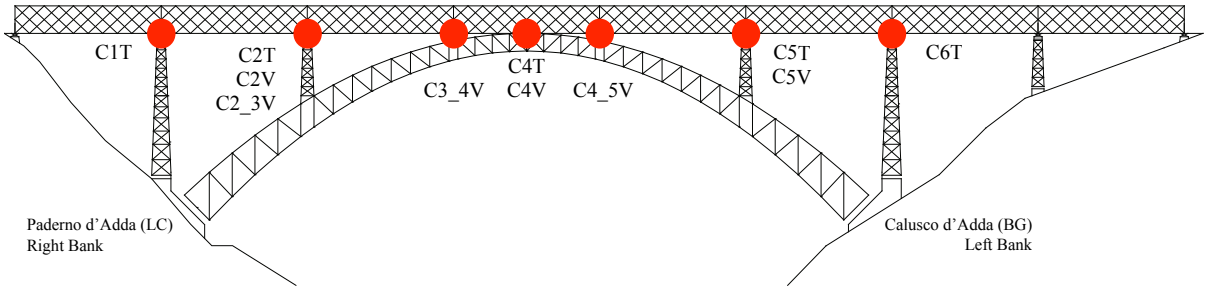


Figure 1: San Michele Bridge (1889), structure and sensor configuration adopted during the experimental campaign (May 2024). Solid red dots indicate sensor location, and labels report number of span and direction, namely "T" for Transverse horizontal direction and "V" for Vertical direction.

During the campaign, dynamic tests were performed with a dedicated rail-vehicle instrumented on the axles with monoaxial and triaxial accelerometers. The tests consisted in the controlled passage of the vehicles represented in Figure 2, at different speeds, moving directions, and configurations. In particular, two configurations were adopted: a locomotive only (a), and locomotive with a short convoy of two wagons (b).

(a) Locomotive (b) Two-wagon convoy

Figure 2: Representation of the locomotive (a) and the two-wagon convoy (b) employed during the experimental campaign (May 2024).

These vehicles ran over the bridge from Paderno d'Adda to Calusco d'Adda and viceversa at different controlled speeds, ranging from 3 km/h to 15 km/h, and the structural response was registered with accelerometers positioned on both structure and vehicle.

For the purposes of this study, only the acceleration signals from structure-mounted sensors are considered, specifically from tests conducted with the locomotive and two-wagon convoy travelling from Paderno d'Adda to Calusco d'Adda.

3 BRIDGE FINITE ELEMENT MODEL

The initial idea was to numerically simulate the dynamic behavior of the structure under a moving load, designed to be as representative as possible of the rail-vehicle employed for the aforementioned dynamic tests.

To achieve this, the FEM model of the bridge, originally developed in MATLAB for static and limit analyses [5, 6] was employed. The model was created using Euler-Bernoulli beam elements, whose mechanical properties were derived directly from the original design drawings. The model was formulated in three dimensions and consisted in 5337 beam elements, 2216 nodes, and 13296 degrees of freedom (dofs). The global FEM model represented the entire structure, including the five piers, the doubly built-in parabolic arch and the upper continuous beam.

For the moving load simulations, only the 3D box-formed upper continuous beam was retained. To capture the influence of the substructure, translational springs were added in the transverse, horizontal, and vertical directions, with stiffness coefficients to be calibrated.

Finally, the damping matrix was defined using a Rayleigh damping model [3]. The two Rayleigh coefficients were determined by solving the system of equations corresponding to a damping ratio of 1% for the first two vibration modes of the structure.

3.1 Substructuring of the bridge

A substructuring approach was adopted to isolate the continuous box girder from the rest of the structure. This process required the implementation of translational springs in the numerical model, to reproduce the supporting conditions otherwise provided by the piers and the bridge arch.

The new boundary conditions were defined under the assumption that the main load-bearing element, i.e. the parabolic arch, primarily undergoes flexural deformations, in addition to axial effects. Instead, the piers mainly experience (limited) axial deformations in the vertical direction, with a contained flexural contribution in the transverse horizontal and longitudinal directions. A schematic 2D drawing of the substructuring process is shown in Figure 3 where boundary conditions were imposed on the 3D box-formed upper continuous beam substructure as a result of the removal of supporting structures.

Concerning the vertical direction for the four supports directly connected to the arch, the induced deformations are governed by the flexural response of the arch rather than by axial effects. For this reason, translational elastic springs are introduced only at the corresponding positions to account for the higher deformation. Specifically, starting from the Paderno d'Adda side (left side according to Figure 3), a couple of elastic springs is located at the second, third, fourth, and fifth bearings both downstream and upstream, as the substructured model is still 3D. All remaining supports, namely at the bridge extremities and at the bearings on the ground, are described by absolute rollers.

To avoid defining an excessive number of springs, the symmetry of the bridge is assumed

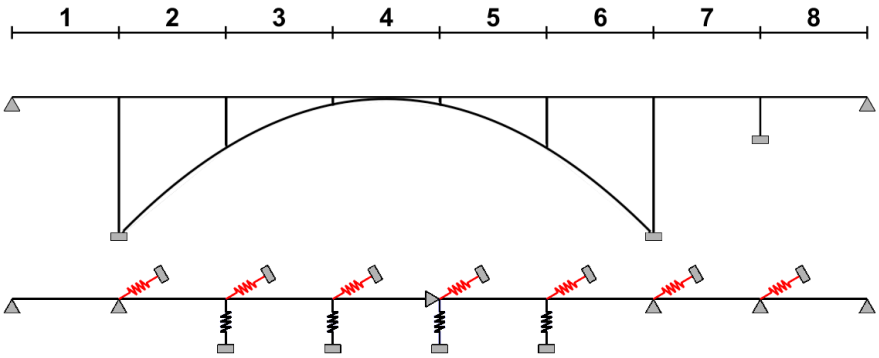


Figure 3: Planar schematic representation of the substructuring process applied to the bridge FEM model. While the beam substructure remains 3D and springs are applied on both sides of the bridge, only one side is illustrated here for clarity. Vertical springs are shown in black and transverse horizontal springs in red.

along the longitudinal direction and with respect to the keystone of the arch. Consequently, each couple has the same stiffness value, and the stiffness of the second couple of springs (between the second and third spans) is assumed equal to that of the fifth spring (between the fifth and sixth spans), while the stiffness of the third spring (between the third and fourth spans) is equal to that of the fourth spring (between the fourth and fifth spans), resulting in two stiffness values to be calibrated in the vertical direction.

In the transverse horizontal direction, the behaviour of the continuous box beam differs significantly. In this case, no symmetry axis was found other than the longitudinal one, resulting in seven distinct stiffness values.

Therefore, boundary conditions are applied exclusively to the edge beams on the lower deck of the box section, as the central beams do not rest on the piers.

The substructured FEM model of the box beam comprises 3,648 beam elements and 4,308 unconstrained dynamic DOFs, including 22 elastic springs defined at the boundaries with the eliminated substructures, of which 9 parameters must be calibrated. Although these values shall be refined through a modal updating procedure, it is essential to provide initial values that may best represent the actual boundary conditions.

To initialize the stiffness values of these springs, two distinct static scenarios are considered, each aimed at calibrating the stiffness of the springs in a specific direction. In both cases, initialization is performed by ensuring the equivalence of displacement values, assessed through static analyses, recorded under different loading scenarios.

For the vertical springs, the selected loading scenario corresponds to the self-weight of the entire structure. A static analysis of the entire structure is conducted, recording the vertical displacement values at the spring application points. The stiffness values are then adjusted until the displacements between the two models converge. The maximum percentage difference between the substructure's deformed shape and the entire bridge's deformed shape is approximately 1.63%. This static analysis additionally confirms that these points exhibited symmetrical behaviour, validating the initial modelling assumption.

To define initial values in the transverse horizontal direction, transverse forces of equal magnitude are applied at the spring locations. Similar to the calibration process in the vertical direction, the spring stiffness values are determined so that they ensure that the transverse displacements closely match those evaluated in the full-structure model. For this case, the maximum percentage difference between the substructure's deformed shape and the bridge's deformed shape is approximately 3.46%.

Stiffness values of the springs introduced in the upper continuous beam, extracted from the complete model, are reported in Table 1, where they are distinguished in transverse horizontal and vertical directions.

Bearing n. / Direction	1	2	3	4	5	6	7
Horizontal	23000	12000	2900	3100	4800	7500	34000
Vertical	–	74000	47000	47000	74000	–	–

Table 1: Stiffness values of the springs introduced in the substructure of the Paderno d'Adda Bridge (1889), to represent the effect of the structure under the box-formed upper continuous beam. Values are reported in kN/m.

Figures 4 and 5 illustrate the static deformed shapes of the system for the vertical and transverse horizontal direction spring calibrations, respectively. In each scenario, the deformation of the entire structure is shown first, followed by that of the upper substructure only.

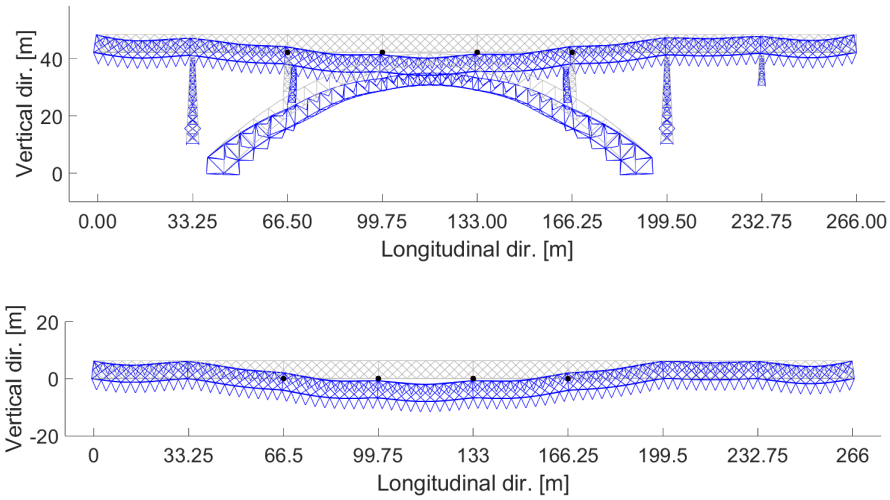


Figure 4: Deformations of full (up) and substructured (down) bridge models in the vertical-longitudinal two-dimensional plane. The deformed shape is shown in blue, the undeformed shape in grey, and the black circles indicate the positions of the vertical springs.

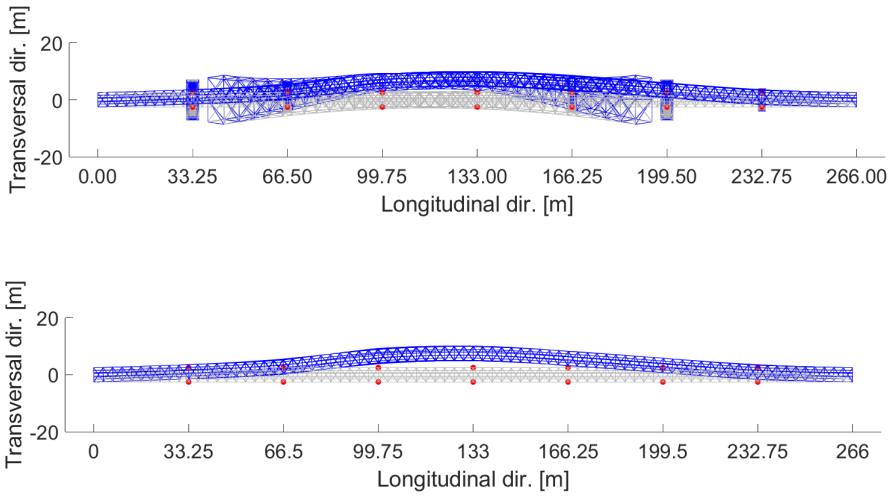


Figure 5: Deformations of full (up) and substructured (down) bridge models in the transverse-longitudinal two-dimensional plane. The deformed shape is shown in blue, the undeformed shape in grey, and the red circles indicate the positions of the transverse horizontal springs.

3.2 Modelling of the rail vehicle as moving loads on the FEM model of the beam

The test rail-vehicle represented by Figure 2 is simplified for the simulation as a sequence of moving loads acting on the rails. These loads were calibrated to reproduce as closely as possible the loading conditions (i.e. the weight of the wagons and locomotive axles) imposed during the experimental campaign. The simplified representation of the convoy is illustrated in Figure 6, where the load is represented by the red arrows and also the relative distance (in meters) between two consecutive axles. The weight of each axle is detailed in Table 2.

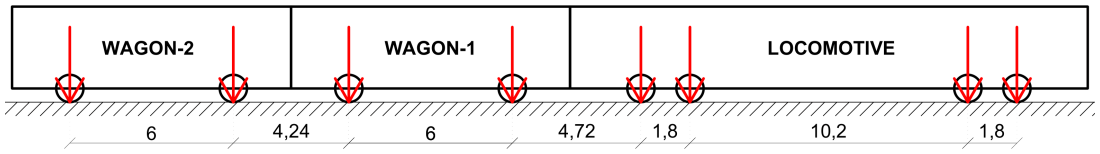


Figure 6: Convoy setup with a graphical representation of the axle spacings (in meters) for the railroad vehicle used during the experimental campaign.

Unit Type	Locomotive	Wagon-1	Wagon-2
1st axle	16.60	14.50	14.50
2nd axle	16.60	14.50	14.50
3rd axle	16.60		
4th axle	16.60		

Table 2: Axle loads [t] of the test convoy.

The sequence of moving loads is applied at the train wheel locations and its effect on the nodes of each beam FE is modelled with equivalent nodal forces varying at each time instant [10]. Consequently, each vertical moving load is distributed at the nodes of each beam via an equivalent vertical forces $\mathbf{V}_i$ and a bending moments $\mathbf{W}_i$, as shown in Figure 7.

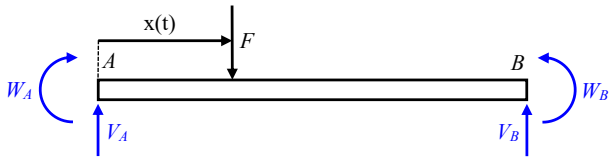


Figure 7: Equivalent nodal forces for a single beam element under a moving load of magnitude F at time instant t

The equivalent nodal forces on nodes A and B, for a single moving load of magnitude F, are determined as:

$$\begin{Bmatrix} V_A(t) \\ W_A(t) \\ V_B(t) \\ W_B(t) \end{Bmatrix} = F \begin{Bmatrix} 1 - \frac{3x^2}{l^2} + \frac{2x^3}{l^3} \\ x - \frac{2x^2}{l} + \frac{x^3}{l^2} \\ \frac{3x^2}{l^2} - \frac{2x^3}{l^3} \\ -\frac{x^2}{l} + \frac{x^3}{l^2} \end{Bmatrix} \tag{1}$$

According to Figure 6, a total of 16 moving loads must be set up to represent the forces transmitted to the bridge by the railroad vehicles: 8 on the left track and 8 on the right track.

Furthermore, it is assumed that the load moves from left to right (from Paderno d'Adda to Calusco d'Adda) at constant speed, as shown in Figure 8.

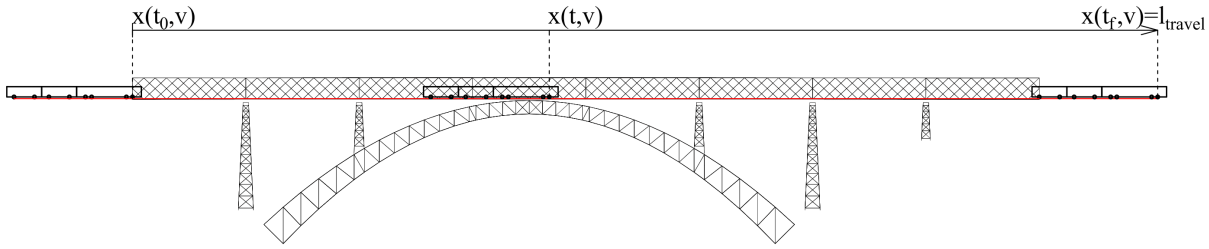


Figure 8: Modeling of the moving load on the San Michele Bridge. The Paderno d'Adda (LC) riverbank is shown on the left side, while the Calusco d'Adda (BG) riverbank, with the additional intermediate pier, is on the right side.

Eventually, the (linear) dynamic equations of motion for the discrete structural model are defined as follows:

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{K}\mathbf{x} = \mathbf{F}(t) \tag{2}$$

The numerical integration of the second-order differential equations, Eq. (2), is computed by a classical Newmark method implementation with the average acceleration scheme. The numerical simulation starts when the first convoy axle enters the upper continuous beam and ends when the last axle exits the structure.

4 DYNAMIC SIMULATIONS OF THE TRAIN CONVOY

Several simulations were carried out with the aforementioned FEM model of the box-formed upper continuous beam of the Paderno d'Adda Bridge subjected to moving loads, testing different traversing speeds and configurations. In particular, in this section, the dynamic response in terms of displacement, velocity, and acceleration is presented for the load configuration in Figure 6, transiting at $v = 6\,\text{km/h} = 1.667\,\text{m/s}$ and at $v = 15\,\text{km/h} = 4.167\,\text{m/s}$. The time step size has been selected by the following relation:

$$\Delta t = \frac{h/v}{n} \tag{3}$$

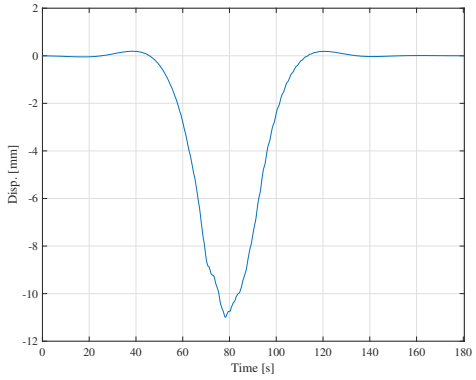
where $h = 3.325$ m represents the length of the rail FEs where the load is applied, v is the speed of the convoy in m/s, and n is the number of steps taken by the moving load to travel along the entire length of the finite element. Adopting $n = 1000$ steps, the two time steps for the two speeds have corresponding size equal to 0.002 s and 0.000798 s.

The numerical results are represented in Figure 9 as time histories of displacement, velocity, and acceleration of the node located at the midpoint of the 4-th span, in correspondence of the key of the arch, at the level of the railway.

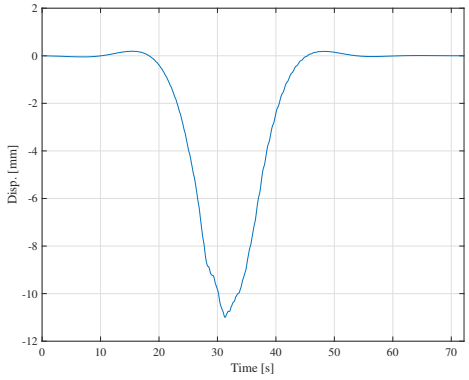
The preliminary results show that, although the train speed increases from 6 km/h to 15 km/h, which still represents a contained speeding condition, the displacement exhibits only a minor variation, whereas the velocity at the observed node rises by a factor of about three (notice the different scales in Figs. 9(c) and 9(d)). However, these initial results in terms of displacement and velocity cannot be experimentally validated, since no corresponding measurements were collected during the experimental campaign. Instead, accelerations from structure-mounted sensors shall be used in future studies to assess the quality of the present FEM simulations.

The effects of speed variation are instead more evident in the acceleration responses (notice again the different scales in Figs. 9(e) and 9(f)), as the node registers a maximum acceleration almost one order of magnitude larger, from about 1.5 mm/s^2 to 10 mm/s^2 when the speed increases from 6 km/h to 15 km/h.

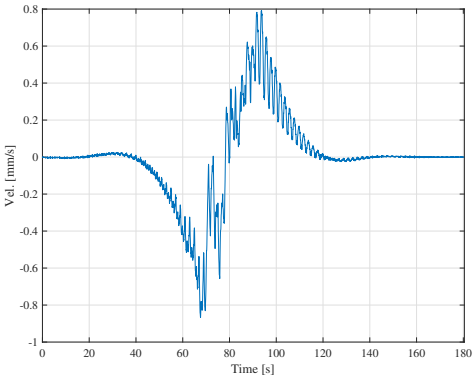
However, these preliminary results seem to show discrepancies with higher acceleration responses recorded during the experimental campaign, suggesting that further modelling refinements shall be needed. As part of ongoing efforts, the first steps will involve also a dynamic calibration of the springs and updating the FEM model based on experimental responses and extracted modal properties [9]. Subsequent studies will focus on a deeper investigation of wheel-rail interaction to better capture its effect on the bridge's dynamic behaviour, also taking advantage of train-recorded signals.



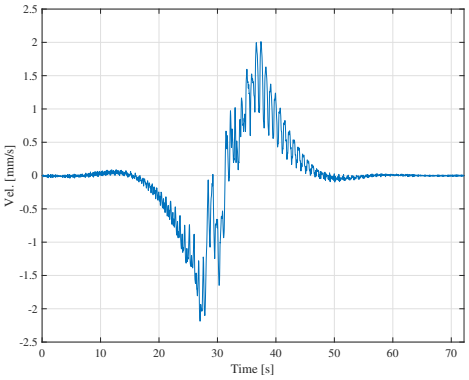
(a) Displacement, $v = 6$ km/h



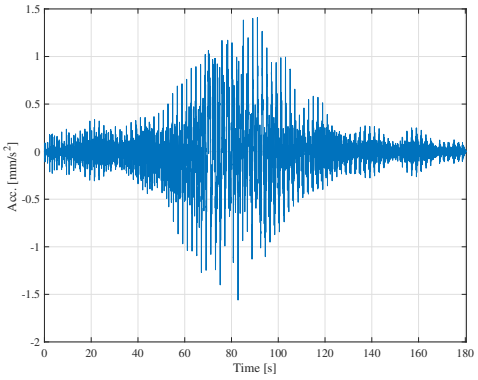
(b) Displacement, $v = 15$ km/h



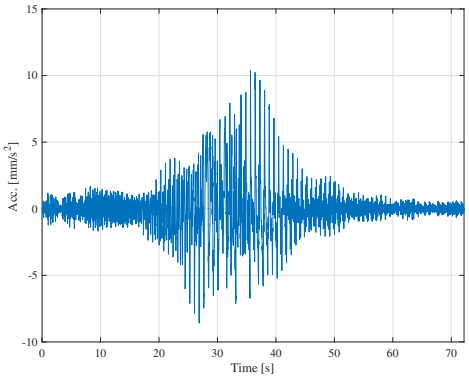
(c) Velocity, $v = 6$ km/h



(d) Velocity, $v = 15$ km/h



(e) Acceleration, $v = 6$ km/h



(f) Acceleration, $v = 15$ km/h

Figure 9: Displacement, velocity, and acceleration responses of the node located at the midpoint of the 4-th span at the key of the arch, when the moving loads transit at $v = 6$ km/h and $v = 15$ km/h.

5 CONCLUSIONS

The initial phase of a broader project devoted to the development of a SHM platform for San Michele Bridge (1889) provided insights into the dynamic behaviour of the infrastructure under ambient, road, and rail excitations. The experimental campaign conducted in May 2024, with sensors installed on both the structure and the test vehicles, provided an exhaustive dataset of structural responses for further analyses and model updating, to be post-processed.

This work reported first numerical simulations carried out within a modelling numerical framework. A FEM model of the structure was developed, based on a previous static model. Other elements, including a lumped mass matrix and Rayleigh damping, were introduced to enable dynamic analyses.

The FEM model was further substructured to retain only the 3D box-formed upper continuous beam, where the rail vehicles transit, within it. To account for the effects of the arch and piers, four vertical springs and seven transverse horizontal springs were added, on each side of the upper continuous beam. The linear dynamic equations of motion for a discrete structural model were solved numerically with the Newmark method.

The simulations highlighted interesting results, about displacement, velocity, and acceleration responses, accelerations likely looking underestimated, by the present modelling assumptions, with experimentally recorded ones. This may suggest possible improvements of the model, to better predict the dynamic behaviour of the Paderno d'Adda Bridge under moving problems. A detailed comparison between numerical and experimental responses shall be investigated in the future.

However, the present analysis proved the importance of including an algorithmic spring stiffness calibration and accounting for wheel-rail interaction. Ultimately, all these efforts shall support the development of a global Digital Twin of the San Michele Bridge (1889) designed to reproduce the bridge's real dynamic behaviour, providing an accurate digital representation of the real structure.

ACKNOWLEDGMENTS

The research conducted at the University of Bergamo was supported by the Italian Ministry of University and Research, through funding under the PRIN 2022 program and "Fondi di Ricerca d'Ateneo ex 60%". RFI, the bridge owner, is gratefully acknowledged for interaction and accessibility to the bridge. The third author would also like to acknowledge the Marie Skłodowska-Curie Actions (MSCA) program under the OMoRail project, Grant ID: 101104968.

REFERENCES

[1] J.S. Bendat, A.G. Piersol *Random Data: Analysis and Measurement Procedures, 4th Edition*. Wiley, 2010.

[2] R. Brincker, L. Zhang, P. Andersen, Modal identification from ambient responses using frequency domain decomposition. Society for Experimental Mechanics eds. *18th International Modal Analysis Conference (IMAC)*, San Antonio, Texas, USA, February 7-10, 2000.

[3] A.K. Chopra, *Dynamics of Structures - Theory and Applications to Earthquake Engineering*, 4th Edition. Prentice Hall, 2011.

[4] R.D. Cook, D.S. Malkus, M.E. Plesha, R.J. Witt, *Concepts and applications of finite element analysis, 4th Edition*, John Wiley & Sons, Inc., 2001.

[5] R. Ferrari, E. Rizzi, M.S. Brioschi, V. Dertimanis, Design of an effective structural health monitoring platform for Paderno d'Adda Bridge (1889). *Reconstruction and Restoration of Architectural Heritage 2021*, S. Sementsov, A. Leontyev, S. Huerta eds., CRC Press, London, 210–215, 2021. DOI: 10.1201/9781003136804-41.

[6] R. Ferrari, G. Cocchetti, E. Rizzi, Reference structural investigation on a 19th-century arch iron bridge loyal to design-stage conditions. *International Journal of Architectural Heritage*, **14**(10), 1425–1455, 2020. DOI:10.1080/15583058.2019.1613453.

[7] R. Ferrari, D. Froio, E. Rizzi, C. Gentile, E.N. Chatzi Model updating of a historic concrete bridge by sensitivity- and global optimization-based Latin Hypercube Sampling, *Engineering Structures*, **179**(2019), 139–160, 2019. DOI: 10.1016/j.engstruct.2018.08.004.

[8] C. Gentile, A. Saisi, Dynamic monitoring of the Paderno iron arch bridge (1889). B. Chen, J. Wei eds. *6th International Conference on Arch Bridges (ARCH'10)*, Fuzhou, China, October 11-13, 2010. Available at: https://hdl.handle.net/11311/569851.

[9] M. Guerini, L. Ermolli, C.D. Stoura, R. Ferrari, V.K. Dertimanis, E.N. Chatzi, E. Rizzi, Modal dynamic identification of historic San Michele bridge (Italy, 1889). In Proceedings of *COMPDYN 2025 - 10th International Conference on Computational Methods in Structural Dynamics and Earthquake Engineering*, Rhodes, Greece, June 15-18, 2025, ID 26049. https://2025.compdyn.org/proceedings.

[10] S.S. Law, J.Q. Bu, X.Q. Zhu, S.L. Chan, Vehicle axle loads identification using finite element method, *Engineering Structures*, **26**(8), 1143–1153, 2004. DOI: 10.1016/j.engstruct.2004.03.017.

[11] F. Pioldi, R. Ferrari, E. Rizzi, Seismic FDD modal identification and monitoring of building properties from real strong-motion structural response signals. *Structural Control and Health Monitoring*, **24**(11), 1–20, 2017. DOI: 10.1002/stc.1982.

[12] F. Pioldi, R. Ferrari, E. Rizzi, Earthquake structural modal estimates of multistorey frames by a refined Frequency Domain Decomposition algorithm. *Journal of Vibration and Control*, **23**(13), 2037–2063, 2015. DOI: 10.1177/1077546315608557.

[13] F. Pioldi, R. Ferrari, E. Rizzi, Output-only modal dynamic identification of frames by a refined FDD algorithm at seismic input and high damping. *Mechanical Systems and Signal Processing*, **68–69**(2016), 265–291, 2016. DOI: 10.1016/j.ymssp.2015.07.004.

[14] C.D. Stoura, E.G. Dimitrakopoulos EG, MDOF extension of the Modified Bridge System method for vehicle-bridge interaction. *Nonlinear Dynamics*, **102**(4), 2103–2123, 2020. DOI: 10.1007/s11071-020-06022-6.

[15] C.D. Stoura, E.G. Dimitrakopoulos, A Modified Bridge System method to characterize and decouple vehicle-bridge interaction. *Acta Mechanica* **231**(9), 3825–3845, 2020. DOI: 10.1007/s00707-020-02699-3.

[16] C.D. Stoura, E.G. Dimitrakopoulos, Additional damping effect on bridges because of vehicle-bridge interaction. *Journal of Sound and Vibration* **476**(2020), 115294, 2020. DOI: 10.1016/j.jsv.2020.115294.

[17] C.D. Stoura, V.K. Dertimanis, E.N. Chatzi, Identification of railway bridge modal properties via acceleration data from traversing trains. R. Platz, G. Flynn, K. Neal, S. Ouellette eds. *42nd IMAC, A Conference and Exposition on Structural Dynamics*, Orlando, Florida, USA, January 29 - February 1, 2024. DOI: 10.1007/978-3-031-68893-5_12.

[18] D. Froio, L. Verzeroli, R Ferrari, E. Rizzi, On the numerical modelization of moving load beam problems by a dedicated parallel computing FEM implementation, *Archives of Computational Methods in Engineering*, **28**(4), 2253–2314, 2021.

[19] S. Weng, Y. Xia, Y.L. Xu, H.P. Zhu, Substructure based approach to finite element model updating, *Computers and Structures*, **89**(9-10), 772–782, 2011. DOI: 10.1016/j.compstruc.2011.02.004.

[20] Y.B. Yang, J.D. Yau, Y.S. Wu, *Vehicle-Bridge Interaction Dynamics: with Applications to High-Speed Railways*. World Scientific, 2004.

UNCECOMP 2025
10th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

UPDATING DIGITAL TWINS SUBJECTED TO UNCERTAIN CORROSION CONDITIONS USING THE UNSCENTED KALMAN FILTER METHOD

Ioannis A. Prentzas[1], **Spyridon G. Diamantopoulos,** [1], **Alexandros D. Michailidis**[1], **and Michalis Fragiadakis**[1]

[1] National Technical University of Athens, School of Civil Engineering
address
e-mail: {gprentzas, sdiamadop, alex_michailidis, mfrag}@mail.ntua.gr

Abstract. *The paper presents a framework for updating models of reinforced concrete (RC) structures due to aging effects. This is accomplished through a nonlinear finite element model (FEM) updating procedure using the Unscented Kalman Filter (UKF) method. The UKF method recursively updates the FEM by integrating new information, resulting in a more realistic representation of the structure over time. Specifically, corrosion-related parameters, such as material degradation and the loss of reinforcement cross-sectional area, are updated based on measurements. The measurements have been generated using probabilistic models to ensure that structural analysis accounts for the progressive deterioration caused by chloride-induced corrosion in RC bridges. The efficiency of various measurement types is examined. Additionally, parametric studies are conducted to investigate how varying corrosion levels and seismic intensities influence the updating procedure. The updated FEM model is subsequently adopted to generate fragility curves that explicitly incorporate the effects of structural aging on performance. The developed framework is applied to a bridge pier in order to evaluate the effectiveness of the proposed methodology for updating the reliability assessments models in the case of aging. A bridge case study is considered in order to demonstrate how the approach effectively considers corrosion effects and seismic performance, providing valuable decision-making regarding bridge maintenance, repair, and retrofit.*

Keywords: Reliability analysis, Unscented Kalman Filter, Aging, Model updating, recursive filter

1 INTRODUCTION

Reinforced concrete (RC) structures deteriorate progressively due to aging, environmental exposure, and prior loading events. Among the various degradation mechanisms, reinforcement corrosion is both the most widespread and one of the most detrimental, especially in marine and coastal regions where chloride ingress accelerates its initiation and propagation. The loss of steel cross-section, reduction in mechanical properties, and associated cracking of the surrounding concrete significantly alter the strength, stiffness, and energy dissipation capacity of RC members. In seismically active regions, these corrosion-induced deficiencies accumulate over time, leaving structures more vulnerable to earthquakes and raising serious concerns about their long-term safety and serviceability.

This dual challenge of deterioration and seismic demand underscores the need for robust methods that can capture the evolving properties of corroded RC structures and provide reliable predictions of their seismic performance. Conventional assessment approaches often neglect deterioration or treat it deterministically, which limits their ability to quantify uncertainty and identify critical parameters. To overcome these limitations, finite-element model updating (FEMU) combined with probabilistic deterioration models provides a powerful framework for condition assessment. In particular, the Unscented Kalman Filter (UKF) [1] enables online nonlinear model updating, allowing for real-time estimation of corrosion-related parameters and improved fragility assessment of RC structures subjected to seismic loading.

Accurate estimation of corrosion-related parameters [3] through structural health monitoring or inspection is crucial for reliable seismic safety assessments and informed decision-making regarding repair, retrofit, or demolition. Model updating [2] using field measurements (e.g., vibration data) offers a viable alternative, as it facilitates the creation of digital twins for performance prediction and maintenance planning [4]. Many previous studies have used model updating to estimate corrosion damage; however, most relied on linear finite element (FE) models and low-amplitude ambient or service loads, rendering them unsuitable for highly nonlinear seismic scenarios [5, 6]. These methods fail to account for complex hysteretic behavior and thus are inadequate for predicting performance under earthquakes.

To address this gap, this study presents a methodology that integrates nonlinear finite element model updating (FEMU) with seismic data, focusing on estimating corrosion-related parameters in RC structures. The proposed approach employs a stochastic inference technique, specifically the Unscented Kalman Filter (UKF), which updates the parameters in real time based on recorded seismic responses [7, 8, 9].

In this study, a nonlinear fiber-based finite element model is developed in the *OpenSeesPy* framework to simulate the seismic response of corroded RC bridge piers. Corrosion-induced deterioration is incorporated by probabilistically reducing the steel reinforcement area and yield strength, while the concrete is modeled with nonlinear fiber sections to capture stiffness and strength degradation. Beyond reproducing the structural response, the framework integrates finite-element model updating with reliability analysis [10, 11], enabling the calibration of corrosion-related parameters and the subsequent updating of fragility curves. This combined approach allows not only accurate simulation of deterioration mechanisms but also improved assessment of the seismic reliability of aging RC piers. By means of simulated case studies incorporating different measurement noise type and ground motion intensities, this study shows that corrosion-related parameter changes—typically difficult to observe directly—can be successfully identified. The goal is to establish a robust, model-based framework that enhances seismic performance assessment and supports risk-informed decision-making for the manage-

ment of aging RC infrastructure.

2 DIGITAL UPDATING UPDATING FRAMEWORK FOR UNCERTAIN CORROSION CONDITIONS

The proposed framework uses Unscented Kalman Filter method to update crucial parameters of FEM model. Especially, corrosion-related parameters are of great interest. For this reason different corrosion induced scenario seismic measurements are produced using probabilistic corrosion models to vadilate the proposed framework. Utilizing recorded seismic data, the model's outputs are compared with observed responses through a Bayesian estimation process, such as the Unscented Kalman Filter (UKF). This model incorporates corrosion-affected parameters, which are initially unknown. The FE model, characterized by corrosion-related variables, is then iteratively updated to minimize discrepancies between simulated and observed responses, allowing for statistical inference of the unknown corrosion-induced properties. It is important to note that this approach necessitates advanced FE modeling techniques capable of capturing the complex effects of corrosion. Notably, it contributes to the refinement of fragility curves, which quantify the probability of reaching or exceeding different damage levels during earthquakes. Prior to discussing the modeling of corroded RC members, a concise overview of the UKF method—central to estimating corrosion-related parameters—is presented.

2.1 Nonlinear FE Model Updating using Uscented Kalman Filter

An online nonlinear FE model can be used to estimate the response of a structure subjected to dynamic excitation by the nonlinear differential equation:

$$\mathbf{M}(\theta)\ddot{\mathbf{u}}_{k+1}(\theta) + \mathbf{C}(\theta)\dot{\mathbf{u}}_{k+1}(\theta) + \mathbf{r}(\mathbf{u}_{k+1}, \boldsymbol{\theta}) = \mathbf{p}_{k+1} \tag{1}$$

where $\mathbf{u}_{k+1}$, $\dot{\mathbf{u}}_{k+1}$, $\ddot{\mathbf{u}}_{k+1} \in \mathbb{R}^n$ (relative displacement, velocity, and acceleration response vectors, respectively), $\mathbf{M} \in \mathbb{R}^{n\times n}$ (mass matrix), $\mathbf{C} \in \mathbb{R}^{n\times n}$ (damping matrix), $\mathbf{r}(\mathbf{q}_{k+1}, \boldsymbol{\theta})\mathbb{R}^n$ (resisting force vector), n = number of degrees of freedom (DOFs), $\boldsymbol{\theta} \in \mathbb{R}^{n\times 1}$ (vector of time-invariant model parameters), $\mathbf{p}_{k+1} = \mathbf{ML}\ddot{u}_{g,k+1}$, (dynamic load vector for rigid-base earthquake excitation), $\mathbf{L} \in \mathbb{R}^{n\times r}$ (influence matrix), $\ddot{u}_{g,k+1} \in \mathbb{R}^{r\times 1}$ (input ground acceleration), r = number of base excitation components, k = 0, 1, . . . , N-1, where N is the number of data samples of the input ground acceleration record.

Then, different types of sensors can be deployed on the structure to measure different response quantities $\mathbf{y} \in \mathbb{R}^{n_y}$, where n_y denotes the number of measured responses. These responses can also be estimated from the nonlinear FE model. The measured and FE-predicted response quantities at time t_{k+1} can be related by [9]:

$$\mathbf{y}_{k+1} = \hat{y}_{k+1}(\boldsymbol{\theta}) + \mathbf{v}_{k+1} = \mathbf{h}_{k+1}(\boldsymbol{\theta}, \ddot{\mathbf{u}}^g_{1:k+1}) + \mathbf{v}_{k+1} \tag{2}$$

where $\mathbf{y}_{k+1} \in \mathbb{R}^{n_y}$ is the measured response at time t_{k+1}, $\hat{y}_{k+1}(\boldsymbol{\theta}) = \mathbf{h}_{k+1}(\boldsymbol{\theta}, \ddot{\mathbf{u}}^g_{1:k+1}) \in \mathbb{R}^{n_y}$ is the FE-predicted response with $\mathbf{h}_{k+1}(\cdot)$ referred to as the nonlinear response function, $\ddot{\mathbf{u}}^g_{1:k+1} = [(\ddot{\mathbf{u}}^g_1)^T, (\ddot{\mathbf{u}}^g_2)^T, \ldots (\ddot{\mathbf{u}}^g_{k+1})^T]^T$ is the input ground motion time history from time t_1 to t_{k+1} and $\mathbf{v}_{k+1} \in \mathbb{R}^{n_y}$ is the prediction error vector assumed here to be white Gaussian with zero-mean and diagonal covariance matrix $\mathbf{R}_{k+1}$, i.e., $\mathbf{v}_{k+1} \sim N(0, \mathbf{R}_{k+1})$. Note that 2 accounts for model parameter uncertainty and uncertainty related to input and response measurements; however, it

does not explicitly account for model structure uncertainty, also referred to as modeling uncertainty. The model parameters to be estimated θ using UKF, can be modeled as a random walk, whose equation can be combined with Eq. 2 to define the following non-linear state-space equation

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \mathbf{w}_k \tag{3}$$

$$\mathbf{y}_{k+1} = \mathbf{h}_{k+1}(\boldsymbol{\theta}, \ddot{\mathbf{u}}^g_{1:k+1}) + \mathbf{v}_{k+1} \tag{4}$$

where $\mathbf{w}_k$ is the process noise assumed to be a white Gaussian with zero mean and diagonal covariance matrices $\mathbf{Q}_k$, i.e, $\mathbf{w} \sim N(\mathbf{0}, \mathbf{Q_k})$ and statistically uncorrelated with $\mathbf{v}_k$. In the proposed methodology, θ represents the unknown correlation-related properties and other model parameters, which are considered uncertain.

In this equation, $\mathbf{h}$ represents the simulated structure response until analysis step k, which is evaluated by analyzing the nonlinear FE model with the current estimate of model parameter vector θ_k, subjected to the seismic input characterized by earthquake ground motion acceleration series until the current time step k, i.e., $\ddot{u}_{g,1:k}$.

2.2 Probabilistic model of corroded RC piers

To predict the seismic response of corroded RC structures under extreme loading, a nonlinear finite-element (FE) model is required to evaluate $\mathbf{h}(\theta, \ddot{u}_g)$ in Eq. 2. Such a model must account for the influence of corrosion on both geometric and material properties. In this study, corrosion is represented as a time-dependent, chloride-induced process, and a probabilistic framework is adopted to generate measurement data that reflect the variability and uncertainty associated with deterioration. The DuraCrete model [12] is adopted in this study to estimate the corrosion initiation time (T_{cor-in}). According to Stewart and Rosowsky [13], the early stage of reinforcement corrosion can be described using the one-dimensional form of Fick's second law. In this framework, the chloride concentration C is defined as

$$\frac{\partial C(x,t)}{\partial t} = D\frac{\partial^2 C(x,t)}{\partial x^2}, \tag{5}$$

The solution to this equation is given by:

$$C(x,t) = C_s\left[1 - \text{erf}\left(\frac{x}{2\sqrt{Dt}}\right)\right]. \tag{6}$$

where $C(x,t)$ is the chloride concentration at depth x and exposure time t; D is the diffusion coefficient; C_s is the surface chloride concentration; and erf$(\cdot)$ is the Gaussian error function. A widely adopted assumption is that reinforcement corrosion initiates once the chloride ion concentration at the depth of the concrete cover exceeds a critical threshold. Based on this concept, the corrosion initiation time ($T_{\text{cor-in}}$) can be estimated using the DuraCrete model [12]:

$$T_{\text{cor-in}} = X_l\left\{\frac{d_c^2}{4k_e k_t k_c D_0 (t_0)^n}\left[\text{erf}^{-1}\left(1 - \frac{C_{\text{cr}}}{C_s}\right)\right]^{-2}\right\}^{\frac{1}{1-n}}. \tag{7}$$

where X_l is the modeling uncertainty factor; d_c is the reinforcement cover depth; k_e, k_c, and k_t are correction factors accounting for environmental conditions, curing time, and test methods,

Parameter	**Distribution**	**Mean**	**St. dev.**
d_c	Normal	38.1	11.4
k_e	Gamma	0.924	0.155
k_c	Beta	2.4	0.7
k_t	Normal	0.832	0.024
D_0	Normal	473	43.2 (10^{-12} mm^2/yr)
n	Beta	0.362	0.245
C_{cr}	Normal	0.90	0.15
C_{s} = Chloride surface concentration	—	(linear function of A_{cs} and ε_{cs} by weight of binder)	
A_{cs} = Parameter used in C_{s}	Normal	7.758	1.36
ε_{cs} = Parameter used in C_{s}	Normal	0	1.105

Table 1: Variables in the diffusion model used to estimate the corrosion initiation.

respectively; D_0 and t_0 are the reference diffusion coefficient and reference time; C_{cr} and C_s are the critical and surface chloride concentrations; and n is the aging factor.

Once corrosion initiates, the cross-sectional area of the reinforcing steel gradually decreases. Because this deterioration is time-dependent, it must be modeled as a function of exposure duration. Accordingly, a time-dependent expression is adopted to represent the evolution of the remaining diameter of the steel bar:

$$D_i(t) = \begin{cases} D_i, & t < T_{\text{cor-in}} \\ D_i - \alpha(t - T_{\text{cor-in}})^{0.71}, & T_{\text{cor-in}} \leq t \leq T_f \\ 0, & t > T_f, \end{cases} \tag{8}$$

where α is the deterioration factor, given by:

$$\alpha = \frac{1.0508 \times (1 - w/c) - 1.64}{d_c} \tag{9}$$

and w/c is the water-to-cement ratio.

$$T_f = T_{\text{cor-in}} + \frac{d_c \times D_i^{1/0.71}}{1.0508 \times (1 - w/c)^{-1.64}} \tag{10}$$

where D_i is the initial diameter of the ith reinforcing bar, T_f is the theoretical time at which D_i reduces to zero, and $D_i(t)$ is the time-dependent diameter. The corresponding mechanical properties of the corroded steel, accounting for the deterioration process, are expressed as:

$$f_y(t) = (1.0 - \alpha_y Q_{\text{corr}})\, f_{y0} \tag{11}$$

where f_{y0}, f_{u0}, and ϵ_{u0} are the yield strength, ultimate tensile strength, and ultimate elongation of the initial uncorroded steel bar, respectively. The term Q_{corr}, representing the averaged percentage of section loss, is given by

$$Q_{\text{corr}} = 100 \times \left\{ 1 - \left[\frac{D_i - D_i(t)}{D_i} \right]^2 \right\}. \tag{11}$$

3 FRAGILITY ASSESSMENT

Fragility curves are useful tools for assessing the seismic capacity of RC structures. The fragility of a system is the probability that an Engineering Demand Parameter, EDP, exceeds a threshold value, edp, and it is defined as follows:

$$F_R(IM) = P(EDP > edp|IM) \tag{12}$$

Engineering demand parameters, $EDPs$, are quantities that characterize the system response, while IM is the Intensity Measure of interest. The EDPs usually considered for RC piers performance assessment are member plastic hinge rotation, or the Interstorey Drift Ratio (IDR), both parameters are directly related to the limit-state demand. Suitable IMs for piers are the Peak Ground Acceleration PGA, or preferably the spectral acceleration at the fundamental period $S_a(T_1, 5\%)$. For the present study, the EDP and the IM adopted are the maximum interstorey drift IDR and the first mode spectral acceleration $S_a(T_1, 5\%)$.

If EDP values are known for given IM levels, assuming that the data follow a lognormal distribution, the probability of exceeding a limit-state threshold edp is given by the expression:

$$P(EDP \leq edp|IM) = \Phi\left[\frac{ln(EDP) - \mu_{lnEDP}}{\sigma_{lnEDP}}\right] \tag{13}$$

where Φ is the standard cumulative distribution function, σ_{lnEDP} is the logarithmic standard deviation of the logarithms of the EDP, and μ_{lnEDP} is the mean value of the logarithms of the EDP.

There are various approaches for calculating the fragility curves of Eq. 1. Figure 1 illustrates the different forms of engineering demand parameter–intensity measure (EDP–IM) data that can be used for fragility analysis, depending on the adopted methodology. Figure 1a presents a capacity curve, such as a force–displacement relationship obtained from seismic pushover analysis. In this format, the data directly describe the structural capacity as a function of demand, and fragility curves are derived by comparing capacity with seismic demand models. Figure 1b shows "stripped" data, where EDP values are provided conditionally at specified IM levels, or conversely, IM values are recorded for fixed EDP thresholds. This representation typically arises in incremental dynamic analysis (IDA) or stripe-based analysis, where ground motions are scaled to different intensity levels. The resulting horizontal or vertical bands of data facilitate direct estimation of fragility functions by quantifying the probability of exceeding predefined limit states at each IM level. Figure 1c illustrates the case where EDP–IM data form a random cloud, as is common in cloud analysis. Here, multiple nonlinear time-history analyses are performed using unscaled ground motions, producing scattered data points across the EDP–IM space. Statistical models, such as generalized linear regression or maximum likelihood estimation, are then employed to relate IM to the probability of exceeding a given EDP threshold. Together, these three formats—capacity, stripe, and cloud—represent the primary pathways to fragility calculation. The choice among them depends on the available data and the level of computational effort. In the present study, the cloud-type dataset is most relevant, as the simulated seismic responses of the corroded pier under varying input motions naturally generate scattered EDP–IM pairs, which can then be used to construct probabilistic fragility curves.

The approach followed in our study is based on the work of Fragiadakis and Vamvatsikos [14] who proposed an approximate method for estimating the median Incremental Dynamic Analysis (IDA) capacity curve and its fractiles from the backbone of the static pushover. Once the fractile IDAs are known, it is straightforward to calculate the fragility.

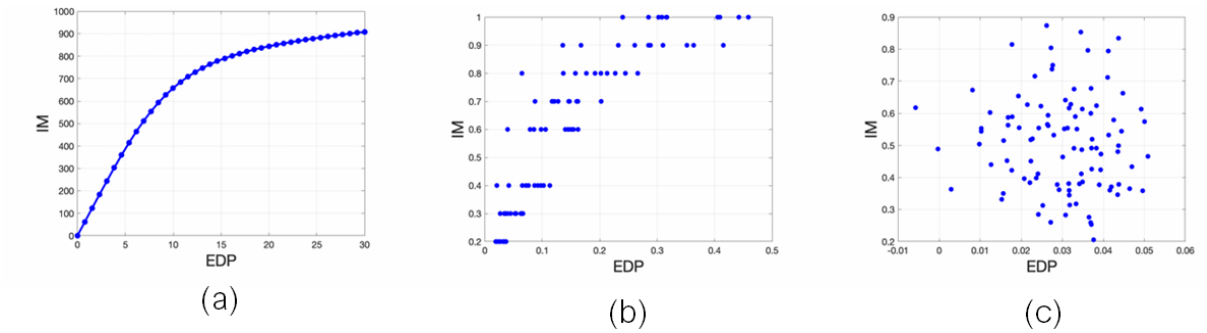


Figure 1: Different methods for creating fragility curves.

4 APPLICATION EXAMPLE

To test the proposed corrosion-related parameter estimation method, we apply it to a representative reinforced concrete (RC) bridge pier. Since field data on corroded RC piers under earthquakes are limited, we use numerical simulations instead. The pier is modeled in OpenSeesPy [15] and subjected to ground motions, and the resulting response histories—including the effects of corrosion—are treated as "measured" data for finite-element model updating. We consider one corrosion scenario, defined by different the initiation time and the exposure duration, under two seismic intensity levels, and perform a global sensitivity analysis to examine how well the key parameters can be identified.

4.1 Example RC bridge pier model

The RC pier examined in this study is representative of highway bridges constructed prior to the adoption of modern seismic detailing standards. It has a hollow rectangular cross-section 2 above a fixed foundation. The concrete, modeled using nonlinear fiber sections with the `Concrete01` material in OpenSeesPy, has a compressive strength of $f'_c = 25$ MPa, a Young's modulus of $E_c = 25$ GPa, and a Poisson's ratio of 0.2. The longitudinal reinforcement has a yield strength of $f_y = 415$ MPa. Rayleigh damping is applied to achieve 5% viscous damping. The pier is modelled using a force-based fiber-beam element to capture flexural and shear responses under cyclic loading. Figure 2 shows the cross-section, with longitudinal bars and transverse ties, as well as the elevation view with element discretization, clarifying how the material and geometric properties are represented in the finite-element model.

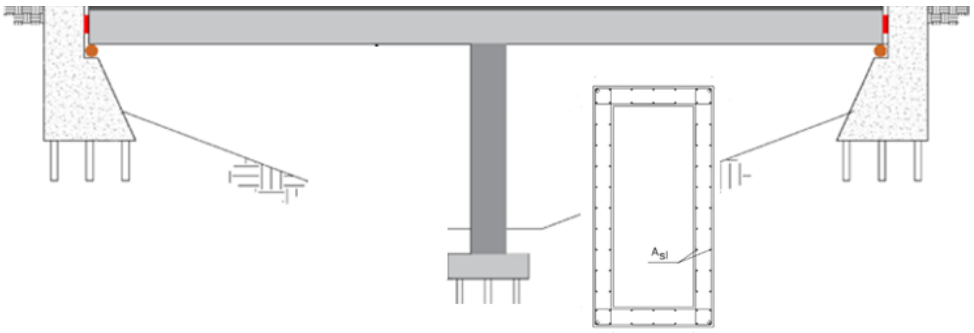
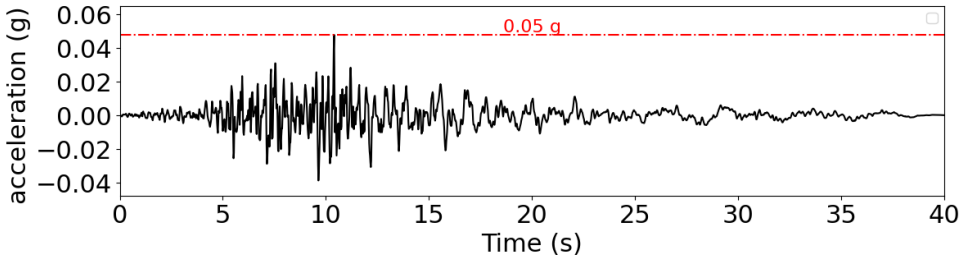


Figure 2: RC pier geometry: (a) cross-section showing longitudinal bars and transverse ties, (b) elevation view with element discretization.

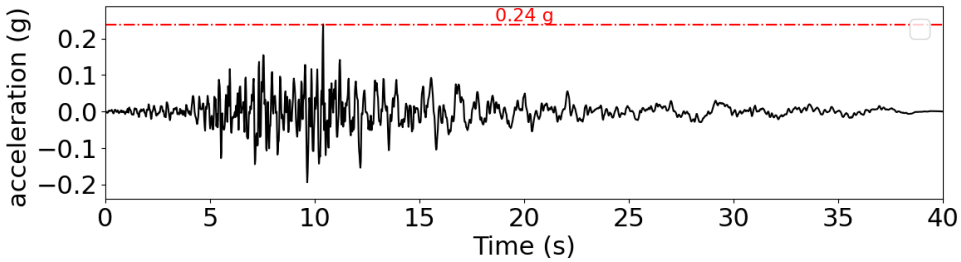
4.2 Corrosion scenario - Seismic input

Corrosion effects are modeled using the probabilistic framework described in Section 2.2, where chloride ingress governs depassivation at $T_{\text{corr-in}}$ and subsequent deterioration develops during the aging period t_{aging}. The considered scenario represents a typical deterioration range, with corrosion initiated after 10 years and progressing over a 50-year aging period. The steel cross-sectional area and yield strength are reduced based on lognormal distributions that capture uncertainty in corrosion rates and spatial variability. These degraded properties are directly assigned to the reinforcement material definitions in the FE model, allowing realistic simulation of stiffness and strength degradation.

To emulate field measurements, the corroded pier models are subjected to two scaled ground motions representative of low-to-moderate and high seismicity regions, corresponding to peak ground accelerations (PGAs) of $0.05g$ and $0.24g$, respectively. Each record is applied at the base in the longitudinal direction with a uniform time step of 0.005 s. Gaussian noise with a 2% standard deviation is superimposed on the signals to mimic sensor error and measurement variability. Figure 3a shows the acceleration time history for the mild corrosion scenario under the $0.05g$ and the $0.24g$ input.



(a) Acceleration time history (pier top) for PGA = $0.05g$ under mild corrosion.



(b) Acceleration time history at pier top for PGA = $0.24g$ and mild corrosion.

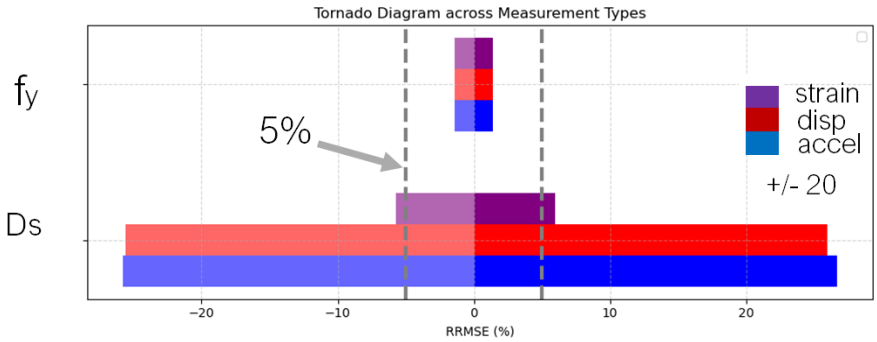
Figure 3: Comparison of synthetic acceleration responses at the pier top under two seismic intensities: (a) PGA = $0.05g$, (b) PGA = $0.24g$.

4.3 Identificability-sensitivity analysis

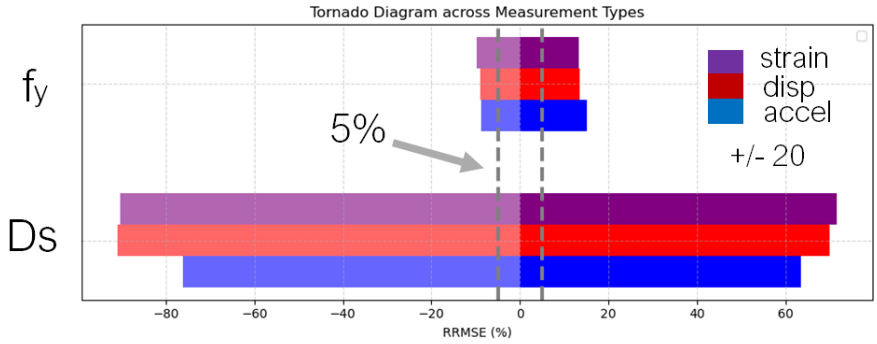
To determine which uncertain parameters most strongly influence the seismic response metrics—or can be inferred from specific measurements—we employ tornado diagrams. These diagrams are constructed by systematically perturbing each parameter by ±10% around its nominal value and quantifying the resulting change in peak acceleration. The parameters are then ranked according to their relative influence.

Figures 4a and 4b present the sensitivity results for the $0.05g$ and $0.24g$ input levels, respectively. In both cases, the diameter of the steel reinforcement and the yield strength f_y are the dominant contributors to response variability, confirming their importance as key corrosion-related parameters. At the lower $0.05g$ excitation, the pier response remains largely linear, which limits parameter identifiability to the rebar diameter. In contrast, under the higher $0.24g$ input, nonlinear effects are activated, allowing both the rebar diameter and yield strength to be identified with confidence. Subsequent figures further isolate the impact of these two parameters, showing the response variation under a ±10% perturbation at each intensity level.

From the perspective of finite-element model updating (FEMU), tornado diagrams provide a practical tool for prioritizing parameters: they highlight which properties are most likely to be identifiable from the available measurements and therefore should be targeted in the updating process.



(a) Tornado sensitivity diagram for peak acceleration under PGA = $0.05g$.



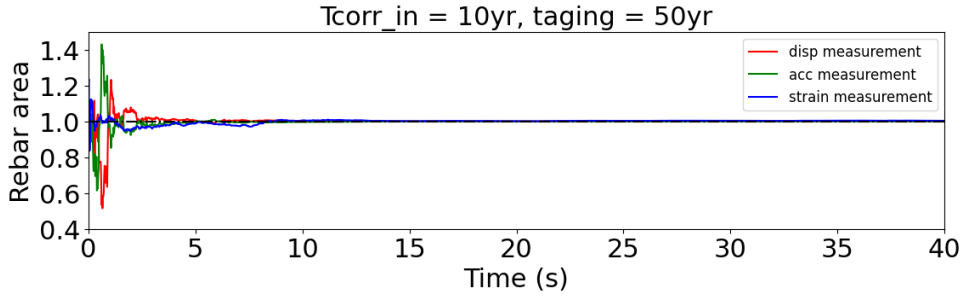
(b) Tornado sensitivity diagram for peak acceleration under PGA = $0.24g$.

Figure 4: Comparison of tornado sensitivity diagrams for two seismic intensities: (a) PGA = $0.05g$, (b) PGA = $0.24g$.

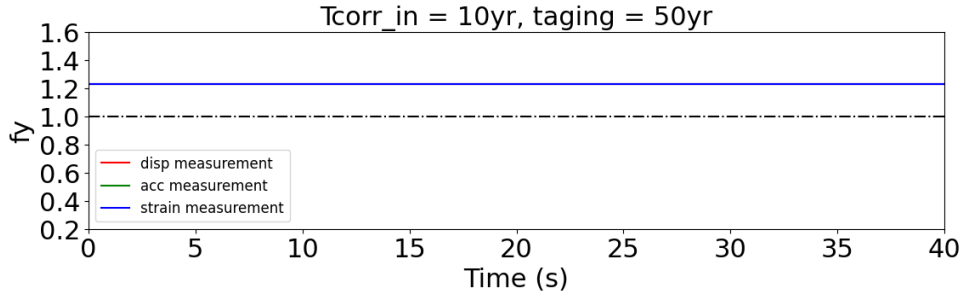
4.4 Results for two ground motion intensity levels

Under the low-intensity $0.05g$ input, the mildly corroded pier primarily exhibits linear behavior, and as a result, the nonlinear parameters of the model cannot be reliably identified, as illustrated by the tornado diagram in Figure 5. The sensitivity analysis indicates that only the diameter of the longitudinal steel reinforcement shows a measurable influence on the seismic response at this intensity level. In particular, the estimation of the yield strength f_y requires a higher seismic

intensity, since the pier must undergo nonlinear deformations before the effects of strength deterioration can be captured.



(a)



(b)

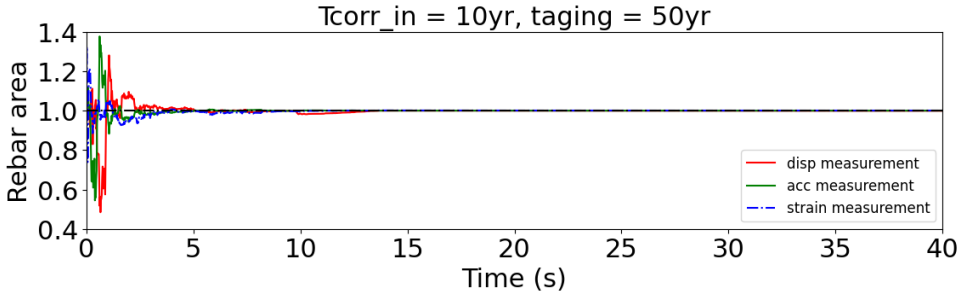
Figure 5: Convergence time-histories of the estimated unknown parameters (PGA = 0.05g)

Under the higher-intensity 0.24g input, the mildly corroded pier develops a distinctly nonlinear response, enabling the identification of parameters that cannot be captured under weaker excitation. Unlike the 0.05g case, where the response remained essentially linear and only the steel rebar diameter could be identified, the stronger shaking activates inelastic mechanisms and makes it possible to identify both the diameter of the steel rebar and its yield strength. The sensitivity analysis confirms that these two reinforcement-related parameters dominate the variability in the seismic response, demonstrating their critical role in capturing the effects of corrosion-induced deterioration. Figure 4b illustrates their relative influence under the 0.24g input, while Figures 6a and 6b show the congergence time-history for the rebar diameter and the yield strength, respectively.

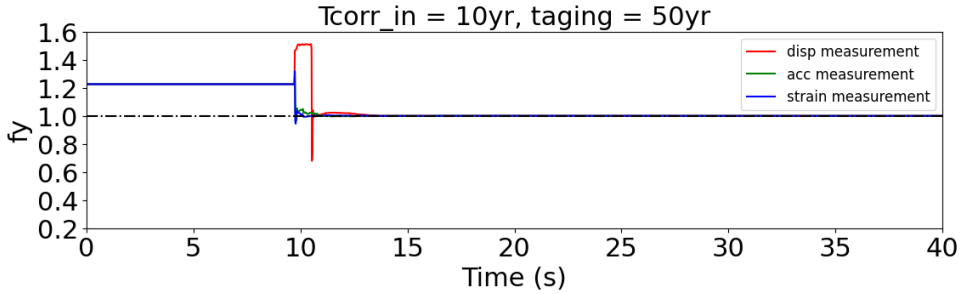
These results highlight the importance of seismic intensity for parameter identifiability. At low levels, FEMU is restricted to identifying parameters linked to elastic stiffness, but at higher intensities, it can also capture nonlinear properties such as reinforcement strength. This underscores the need for strong-motion data in structural health monitoring applications, since only under sufficient excitation can the key corrosion-related parameters be reliably extracted and calibrated.

4.5 Unforeseen data

Figure 7 compares the drift ratio time histories predicted by the initial and updated FE models for the 0.05g input. As shown in Figure 7a, the initial model exhibits noticeable discrepancies relative to the reference corroded response, particularly in terms of peak amplitudes and phase



(a)

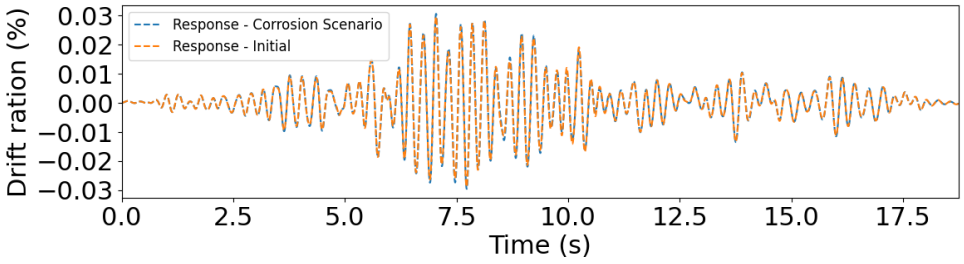


(b)

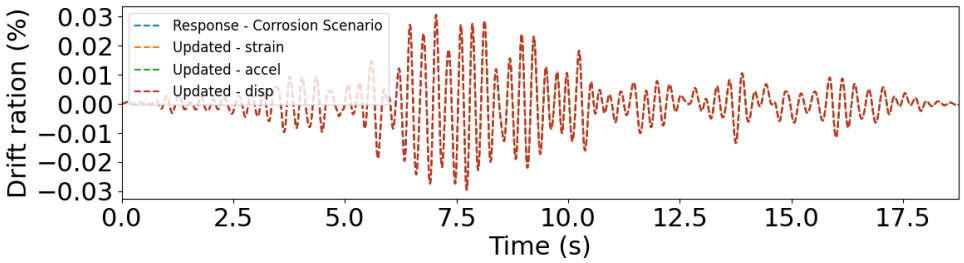
Figure 6: Convergence time-histories of the estimated unknown parameters (PGA = 0.24g).

alignment, highlighting its limited predictive capability when corrosion-induced degradation is not properly accounted for. In contrast, Figure 7b demonstrates the substantial improvement achieved through model updating: the updated predictions across strain, acceleration, and displacement closely track the reference signals, capturing both the amplitude and timing of the structural response. Quantitatively, the root-mean-square (RMS) error decreases from 8.5% in the initial model to just 0.05% after updating, indicating a nearly exact reproduction of the measured response. This dramatic reduction in error confirms the effectiveness of the FEMU procedure in calibrating corrosion-affected parameters and restoring predictive accuracy. Moreover, Figure 7 illustrates the generalization capability of the updated model when subjected to an unseen ground motion (PGA = 0.05g), further validating its robustness beyond the calibration dataset.

Figure 8 compares the drift ratio time histories predicted by the initial and updated FE models for the 0.24g input. As shown in Figure 8a, the initial model exhibits a pronounced mismatch with the reference corroded response, especially around peak drift values, where both the amplitude and phase deviate significantly. This discrepancy highlights the inability of the uncalibrated model to capture the combined effects of material deterioration and nonlinear cyclic behavior under stronger shaking. In contrast, Figure 8b illustrates the improved performance of the updated model, where predictions of strain, acceleration, and displacement align closely with the reference data across the entire response history. The root-mean-square (RMS) error is reduced from 46% with the initial model to just 0.10% after updating, underscoring the effectiveness of the FEMU procedure in calibrating key corrosion-related parameters. This substantial reduction not only confirms the accuracy of the updated model but also demonstrates its robustness when applied to higher seismic intensities, where nonlinearities and degradation effects are more pronounced.

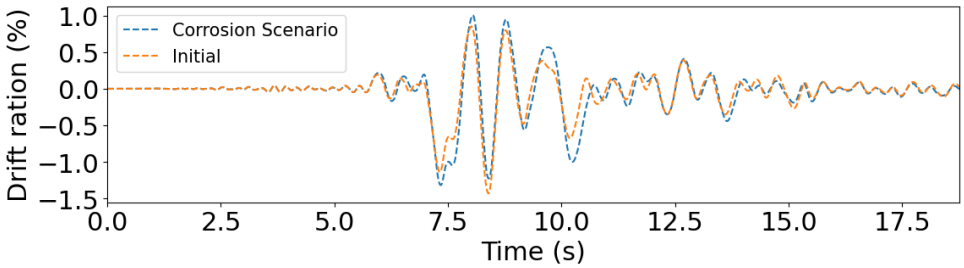


(a) Prediction to ground motion with the initial FEA model (PGA = 0.05g).

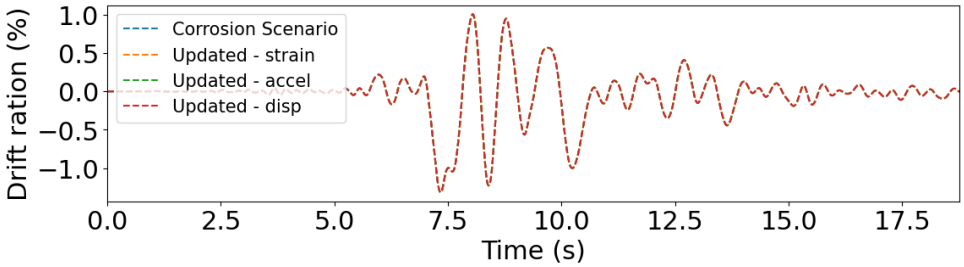


(b) Prediction to ground motion using the updated FEA model, in unforseen data (PGA = 0.05g).

Figure 7: Comparison of the prediction of the initial and updated FEA model to unforseen ground motion (PGA = 0.05g).



(a) Prediction to ground motion with the initial FEA model (PGA = 0.24g).



(b) Prediction to ground motion using the updated FEA model, in unforseen data (PGA = 0.24g).

Figure 8: Comparison of the prediction of the initial and updated FEA model to unforseen ground motion (PGA = 0.24g).

Figure 9 presents the fragility curves corresponding to a drift threshold of 0.020 (2%) for the case of PGA = 0.24g. The "Initial" curve, obtained prior to updating, shows a noticeably shifted response, underestimating failure probability at lower intensity levels and overestimating it at higher ones. After updating with different measurement types (strain, acceleration, and displacement), the fragility curves converge closely toward the "True" reference curve. This convergence demonstrates that model updating substantially improves the accuracy of fragility prediction, regardless of the measurement type employed, and reduces the bias present in the initial model.

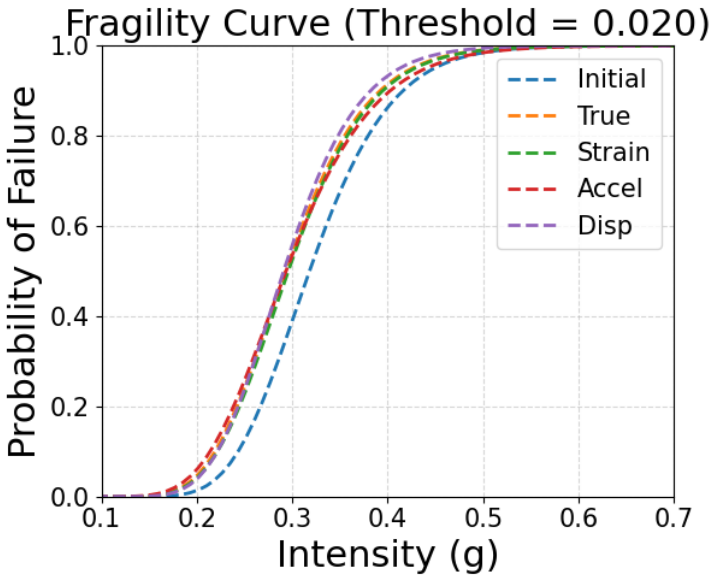


Figure 9: Initial and updated fragility curves at a drift-ratio threshold of 0.02 (2%) for different measurement types; case with PGA = 0.24g.

5 CONCLUSIONS

This work presented a FEMU framework for corrosion-affected RC bridge piers that combines an online nonlinear FE model with an Unscented Kalman Filter (UKF) and a probabilistic, time-dependent corrosion model. Using a representative pier, we generated synthetic "measured" responses under two seismic intensity levels (PGA = 0.05g and 0.24g) and quantified parameter influence via tornado diagrams built from $\pm 10\%$ perturbations. Identifiability, depended on excitation level—at 0.05g the response was largely linear and only the rebar diameter could be isolated with confidence, whereas at 0.24g the nonlinear response enabled reliable identification of both rebar diameter and f_y.

Model updating improved predictive accuracy on unforeseen ground motions. For the low-intensity case, the updated model reproduced the target response with an RMS error of 0.05% (down from 8.5% for the initial model), and for the higher-intensity case, the RMS error dropped from 46% to 0.10%. Beyond response matching, the framework also enhanced fragility prediction. As shown in Figure 9, the initial fragility curve deviated significantly from the reference, particularly at intermediate intensity levels. After updating with different measurement types (strain, acceleration, displacement), the fragility curves aligned closely with the "True" curve, demonstrating that the proposed methodology not only calibrates structural parameters but also yields accurate probabilistic capacity estimates.

Overall, the results confirm that (i) corrosion-related reinforcement parameters are both influential and identifiable, (ii) sufficient seismic demand is essential to activate nonlinear mechanisms needed for full parameter recovery, and (iii) the UKF-based FEMU framework substantially improves both deterministic response predictions and probabilistic fragility assessments. These findings highlight the potential of the method for structural health monitoring and seismic risk evaluation of aging RC bridge piers, with future work directed toward validation with field data and system-level applications.

6 ACKNOWLEDGMENT

The research project was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the "2nd Call for H.F.R.I. Research Projects to support Faculty Members & Researchers" (Project Number: 04140).

REFERENCES

[1] E.A. Wan, R. Van Der Merwe, The unscented Kalman filter for nonlinear estimation. *Proc. IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium*, Lake Louise, Canada (2000), 153–158. https://doi.org/10.1109/ASSPCC.2000.882463

[2] M. Fragiadakis, I. Prentzas, M. Georgioudakis, S. Diamantopoulos, O. Markogiannaki, S. Stefanidou, Integration of measured data for the seismic fragility assessment of deteriorated bridges. *EuroStruct 2023 Eur. Assoc. Qual. Control Bridges Struct. Digit. Transf. Sustain.*, 6(5) (2023), 1377–1381. https://doi.org/10.1002/cepa.2207

[3] S. Abtahi, Z. Liu, Y. Li, Corrosion-related parameter estimation for RC structures using UKF-based Bayesian nonlinear finite element model updating with seismic data. *Mech. Syst. Signal Process.*, 191 (2023), 110169. https://doi.org/10.1016/j.ymssp.2023.110169

[4] C.-S. Shim, N.-S. Dang, S. Lon, C.-H. Jeon, Development of a bridge maintenance system for prestressed concrete bridges using 3D digital twin model. *Struct. Infrastruct. Eng.*, **15**(10), 1319–1332, 2019.

[5] S. Jang, J. Li, B.F. Spencer, Corrosion estimation of a historic truss bridge using model updating. *J. Bridge Eng.*, **18**(7), 678–689, 2013.

[6] S.A. Faroz, N.N. Pujari, S. Ghosh, Reliability of a corroded RC beam based on Bayesian updating of the corrosion model. *Eng. Struct.*, **126**, 457–468, 2016.

[7] M. Hoshiya, E. Saito, Structural identification by extended Kalman filter. *J. Eng. Mech.*, **110**, 1757–1770, 1984.

[8] E.N. Chatzi, A.W. Smyth, The unscented Kalman filter and particle filter methods for nonlinear structural system identification with non-collocated heterogeneous sensing. *Struct. Control Health Monit.*, **16**, 99–123, 2009.

[9] R. Astroza, H. Ebrahimian, J.P. Conte, Material parameter identification in distributed plasticity FE models of frame-type structures using nonlinear stochastic filtering. *J. Eng. Mech.*, **141**, 4014149, 2015.

[10] I. Prentzas, M. Fragiadakis, Quantified active learning Kriging model for structural reliability analysis. *Probab. Eng. Mech.*, 78 (2024), 103699. https://doi.org/10.1016/j.probengmech.2024.103699

[11] I. Prentzas, S. Diamantopoulos, M. Fragiadakis, Multi-hazard fragility updating using structural health monitoring. *Proc. 18th World Conf. Earthq. Eng. (WCEE2024)*, Milan, Italy, June 30–July 5, 2024. https://proceedings-wcee.org/view.html?id=23310 & conference=18WCEE

[12] DuraCrete, Statistical quantification of the variables in the limit state functions. The European Union Brite EuRam 3 contract BRPR-CT95-0132 Project BE95-1347 Report no BE95-1347/R7. (2000).

[13] M.G. Stewart, D.V. Rosowsky, Time-dependent reliability of deteriorating reinforced concrete bridge decks. *Struct. Saf.*, 20(1) (1998), 91–109.

[14] M. Fragiadakis, D. Vamvatsikos, Fast performance uncertainty estimation via pushover and approximate IDA. *Earthq. Engng. Struct. Dyn.*, 39 (2010), 683–703. https://doi.org/10.1002/eqe.965

[15] M. Zhu, F. McKenna, M.H. Scott, OpenSeesPy: Python library for the OpenSees finite element framework. *SoftwareX*, 7 (2018), 6–11. https://doi.org/10.1016/j.softx.2017.10.009

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

RISQ: RANDOM AND INTERVAL SPATIAL UNCERTAINTY QUANTIFICATION SOFTWARE

Ben Van Bavel[1,2], **Robin Callens**[1,2], **David Moens**[1,2]

[1]KU Leuven, Department of Mechanical Engineering, Division LMSD
St.-Katelijne-Waver 2860, Belgium
e-mail: {ben.vanbavel, robin.callens, david.moens}@kuleuven.be
[2]Flanders Make@KU Leuven

Abstract. *This work presents a new software for spatial uncertainty quantification in high-dimensional finite element models called RISQ (Random and Interval Spatial uncertainty Quantification software). RISQ provides a wide range of spatial uncertainty tools for non-intrusive uncertainty propagation in an easy-to-use graphical user interface. The graphical user interface visualizes large finite element models and eliminates the required coding knowledge and implementation time required of end-users. Current features of RISQ include modeling of spatial uncertainty with local/global explicit interval fields and Gaussian/non-Gaussian homogeneous/heterogeneous random fields. Furthermore, RISQ facilitates the propagation of intervals or correlated random variables with standard methods such as vertex and Monte Carlo approaches. To showcase the potential of RISQ, a case study is presented to demonstrate its capabilities and efficiency in addressing spatial uncertainty in finite element models with non-Gaussian heterogeneous random fields.*

Keywords: Uncertainty Quantification, Spatial uncertainty, Random fields, Interval fields

1 INTRODUCTION

In the realm of modern manufacturing, the quest for optimizing product performance while minimizing production costs has become increasingly critical. This challenge is exacerbated by the inherent variability introduced during the production process, such as dimensional deviations [1] and material defects [2, 3], both of which can significantly affect the efficiency and reliability of the final product. Traditional approaches to manage this variability often involve stringent quality control measures and in-line testing. While effective, these often lead to increased costs and material waste.

User interface software packages [4, 5, 6] for uncertainty quantification (UQ) primarily focus on inter-variability, addressing global material parameters or environmental changes across different product realizations. These software packages often fall short in capturing intra-variability, i.e., the spatial variations and local defects within a single product. This gap in current UQ techniques limits the ability to identify and mitigate regions within a product that are particularly sensitive to variability, thereby hindering efforts to enhance both robustness and cost-efficiency.

The RISQ software aims to address these limitations by providing a comprehensive framework for spatial uncertainty quantification. By integrating advanced modeling and propagation techniques, RISQ enables the inclusion of intra-variability in the numerical evaluation of product performance. This approach not only offers a more detailed understanding of how spatial uncertainties affect product reliability but also paves the way for targeted design and manufacturing improvements. In addition, RISQ's graphical user interface aims to guide the user step-by-step through the uncertainty quantification process, making reliability analysis accessible even to those with limited expertise or limited coding experience.

This paper is organized as follows: Section 2 discusses the theoretical framework to perform uncertainty quantification. Section 3 discusses how the software architecture of RISQ implements this theoretical framework. Section 4 provides an example problem that outlines the steps required to perform an uncertainty quantification study in RISQ and highlights its spatial variability modeling capabilities. Section 5 discusses the current state of RISQ. Finally, Section 6 draws conclusions.

2 THEORETICAL UNCERTAINTY QUANTIFICATION FRAMEWORK

RISQ's high-level theoretical workflow hinges on an uncertainty quantification framework presented by de Rocquigny et al. [7]. Figure 1 shows that this framework divides uncertainty quantification into four broad steps:

1. **Model definition**: The first step requires the definition of a parametric (deterministic) numerical model, its input parameters and its output quantities of interest. An example of such a numerical model is the finite element representation of a mechanical component, whose displacement and stress state can act as output quantities of interest.

2. **Input uncertainty definition**: The second step requires the identification of uncertain inputs to the numerical model and quantify their uncertainty. The uncertain nature of the input (e.g., aleatory or epistemic) affects the modeling framework choice that is used to represent it (such as probability theory, fuzzy arithmetic or interval arithmetic).

3. **Uncertainty propagation**: Uncertainty propagation consists of repeatedly solving the numerical model defined in step 1 for different combinations of input parameters, sampled from step 2.

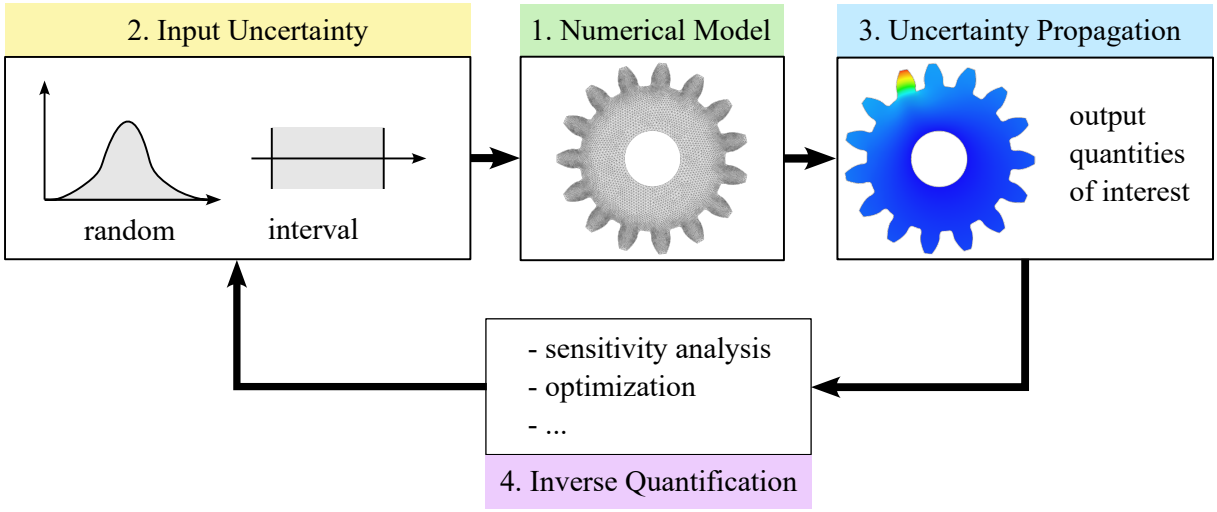


Figure 1: Uncertainty quantification framework schematic used by RISQ.

4. **Inverse quantification**: If needed, the output quantities of interest can be used to perform inverse quantification or sensitivity analysis on the input uncertainties.

By integrating these steps, RISQ provides a comprehensive framework for spatial uncertainty quantification.

3 FRAMEWORK IMPLEMENTATION AND RISQ ARCHITECTURE

The theoretical framework for uncertainty quantification, as outlined by de Rocquigny et al. [7] in Section 2, is implemented in the RISQ software to provide a practical and user-friendly approach to spatial uncertainty quantification in numerical simulations. The following steps detail how this framework is translated into an actionable workflow within RISQ. Coloured circles indicate how each step is linked to the theoretical framework in Section 2. Figure 2 provides a schematic illustration of these steps.

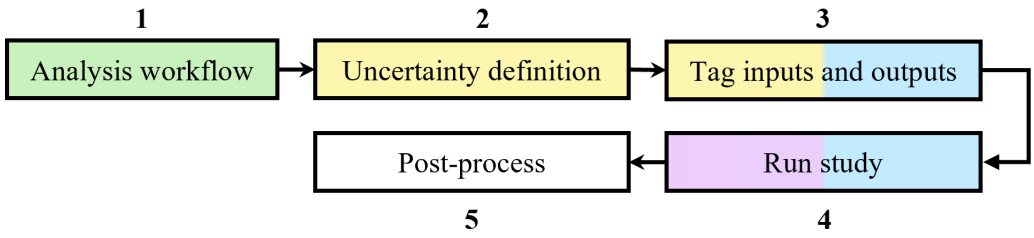


Figure 2: Schematic illustration of RISQ's uncertainty quantification workflow.

◯ **Define analysis workflow**: In RISQ, each analysis workflow is structured as a sequence of analyses (numerical simulations), with a clear execution order. An analysis is defined by its inputs and outputs, allowing users to map out the flow of data through the system. Pre-existing analysis portals help users interface with inputs and outputs of numerical models set up with other software, such as commercial finite element analysis software.

Define input uncertainty: Users specify the number of uncertain inputs within the workflow, and select an appropriate modeling framework for each, whether probabilistic, interval or parametric. For each modeling framework, additional tools are available to further specify the input uncertainty (e.g., definition of probability distribution and autocorrelation structure for probabilistic inputs).

Tag input uncertainty and output quantities of interest: Users can tag input uncertainties to specific analysis input parameters. In addition, tagging output quantities of interest allows RISQ to store these outputs for later post-processing.

Run study: Users select the type of study they wish to conduct, such as "forward propagation", "sensitivity analysis" or "parameter optimization". Users can then run the study when it has been properly set up. Tagged inputs and outputs can then be used for later post-processing within RISQ.

Post-process: RISQ offers a set of post-processing tools designed to help users visualize and interpret the results of their studies, such as histogram and simulation mesh plots.

By following these steps, RISQ translates the theoretical framework into a practical, user-friendly software solution. This implementation not only facilitates the quantification of spatial uncertainties, but also empowers users to make informed decisions to enhance product robustness and cost-efficiency.

4 EXAMPLE: PROBABILISTIC COMPOSITE PLATE BUCKLING

RISQ currently contains several tools for the uncertainty quantification of spatial variability. This Section studies the probabilistic buckling of a composite plate, subject to a spatially uncertain ply angle, through forward uncertainty propagation utilizing Monte Carlo sampling.

4.1 Analysis definition

Figure 3 shows the dimensions and mesh of a simple unidirectional carbon-fibre-reinforced composite plate. Table 1 provides the material properties of the composite material. Figure 3 illustrates that the plate is subject to an in-plane compressive force. Nastran SOL105 is used to calculate the first eigenfrequency and eigenmode of this buckling problem. The model formulation is defined as a Nastran input file (bulk data file .bdf) using Simcenter 3D as a pre-processor. Running the Nastran solver provides us with an ASCII output file (.op2) of the nominal (deterministic) problem that contains our output quantities of interest (i.e., the eigenfrequencies and eigenmodes). Within RISQ the Nastran analysis is run multiple times for uncertainty quantification.

Table 1: Material properties used for the composite plate in the example problem.

Property	Value
E_{11}	$139.27\,\mathrm{GPa}$
$E_{22/33}$	$7.70\,\mathrm{GPa}$
$\nu_{12/13}$	0.299
$G_{12/13}$	$4.03\,\mathrm{GPa}$
G_{23}	$2.74\,\mathrm{GPa}$

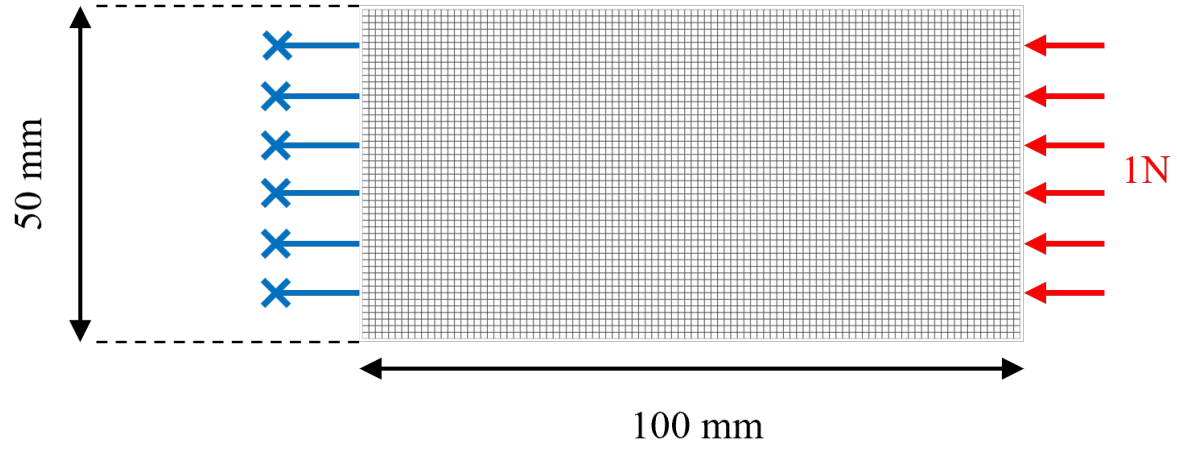


Figure 3: Unidirectional composite plate mesh with loads and constraints in red and blue, respectively.

4.2 Input uncertainty definition

The next step consists of specifying uncertain inputs within the analysis workflow. As an example, we consider the ply orientation of the unidirectional composite plate to be stochastic. We further specify the ply angle to follow a Gaussian distribution with zero mean (i.e., the nominal fiber orientation aligns with the load direction) and a standard deviation of $5°$. The ply angle exhibits spatial variation that we introduce by specifying a two-dimensional auto-correlation structure. The auto-correlation structure uses a modified exponential kernel K_{XX}:

$$K_{XX}(\tau) = \frac{e^{-|\tau|/b}}{1 + |\tau|/b}, \tag{1}$$

where τ represents the spatial lag between two positions and b is a parameter that determines the decay rate of the kernel. For each dimension of the random field, we specify a distance at which a certain correlation should remain. Both values are used by RISQ to iteratively calculate the parameter b internally, depending on the kernel choice of the user. The random field is discretized using the stepwise covariance matrix decomposition method [8].

4.3 Input and output tagging

By pointing our Nastran analysis to a pre-existing nominal input and output file, the analysis portal provides the user with GUI to select the input parameters contained within the Nastran analysis. In our case, the Nastran input file contains as input parameters: orthotropic material properties and laminate properties. We tag our “uncertain ply angle” to the “ply angle input parameter” of the first (and only) ply contained within our laminate. The tag changes the nominal value of the input parameter to an uncertain input. Likewise, RISQ provides the user with a graphical user interface to show the output parameters contained within the nominal Nastran analysis output file. In our case, the Nastran output file contains both displacement and modal results. We tag both the eigenmode and eigenvalue results as quantities of interest for later post-processing steps.

4.4 Run study

The next step consists of defining our uncertainty quantification study. In our example, we use standard Monte Carlo sampling with 100 samples in total with 5 parallel processes to speed

up the uncertainty quantification.

4.5 Post-processing

Figure 4 shows a histogram made with RISQ of the first eigenvalue for the 100 samples created in Section 4.4. Figure 5 shows a mesh plot of a realization of the spatially variable ply angle on the composite plate and its corresponding first eigenmode.

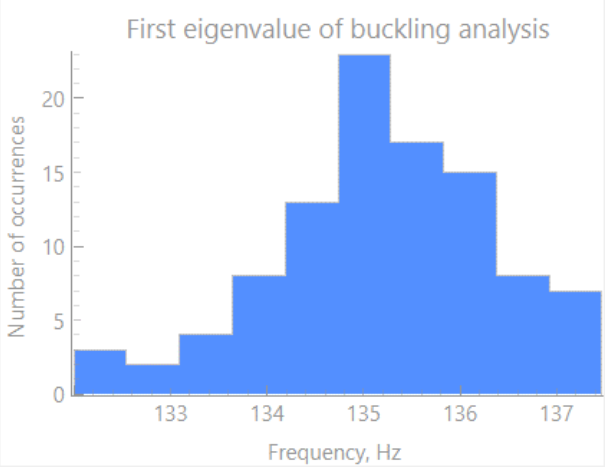
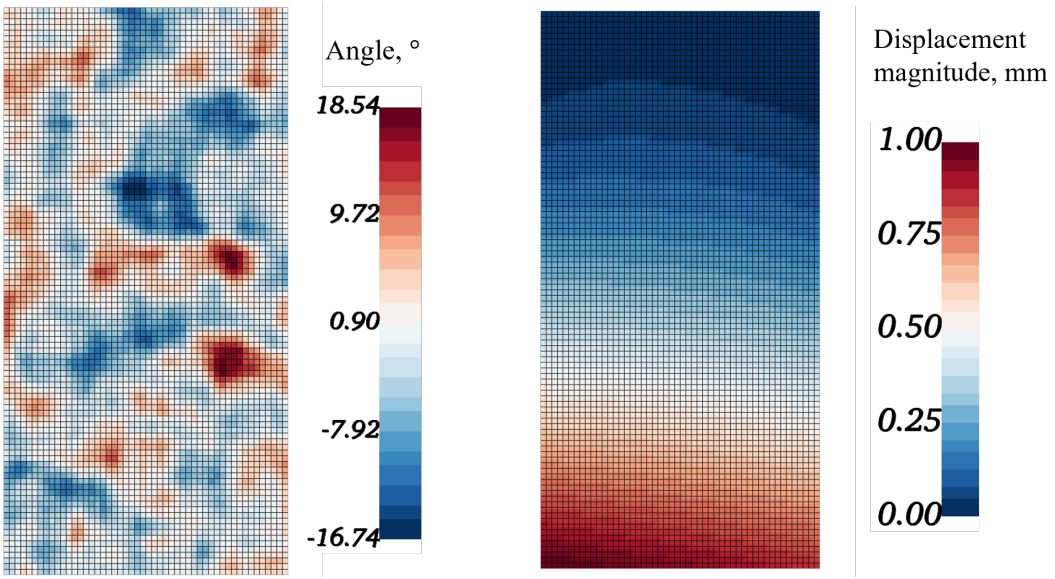


Figure 4: Histogram plot of 100 stochastic realizations of the first eigenvalue of the corresponding uncertain composite plate.



(a) Ply angle (b) First eigenmode

Figure 5: Mesh plots of (left) a ply angle random field realization and (right) the first eigenmode of the corresponding uncertain composite plate.

5 CURRENT STATE OF RISQ

The current state of RISQ provides the core structure required to perform spatial uncertainty quantification as outlined in Section 3. The non-intrusive architecture allows RISQ to interact

with a wide range of software packages, such as Nastran, Matlab and Python.

RISQ currently supports a wide range of (non-)spatial input uncertainties: (Probabilistic) Support for defining a wide range of univariate parametric distributions, cross-correlated multivariate distributions and auto-correlated multivariate distributions using Gaussian and non-Gaussian random field theory. (Interval) Support for defining both scalar intervals and local and global explicit interval fields. Advanced techniques are implemented to efficiently propagate these uncertainties the study workflow to the quantities of interest. Finally RISQ also supports a range of, inverse identification techniques, such as sensitivity analyses.

After the analysis, a wide range of visualization tools including mesh plots, histograms and scatter plots are available for post-processing.

6 CONCLUSION

This paper presents RISQ as a software capable of performing spatial uncertainty quantification. Its software architecture is based on the theoretical uncertainty quantification framework described by deRocquigny et al. [7]. Through its graphical user interface, RISQ aims to significantly reduce the expertise and coding effort required to perform comprehensive spatial uncertainty quantification on numerical simulations, rendering it useful to both academic and industrial actors. The case study of a composite plate buckling simulation with spatially uncertain composite properties illustrates RISQ's workflow and some of its features. Future developments will focus on extending software compatibility with commercial finite element analysis software, and further increasing its computational efficiency, opening up a wider audience for RISQ.

REFERENCES

[1] Nataly A. Manque, Jan Liedmann, Franz-Joseph Barthold, Marcos A. Valdebenito, and Matthias G.R. Faes. Interval isogeometric analysis for coping with geometric uncertainty. *Computer Methods in Applied Mechanics and Engineering*, 437:117773, March 2025.

[2] Robin R.P. Callens, Matthias G.R. Faes, and David Moens. Local explicit interval fields for non-stationary uncertainty modelling in finite element models. *Computer Methods in Applied Mechanics and Engineering*, 379:113735, June 2021.

[3] Ben Van Bavel, Dirk Vandepitte, and David Moens. *Reliability-Based Design of Filament Wound Composite Pressure Vessels: Incorporating Multiscale Spatial Material Variability*. PhD thesis, KU Leuven, 2025-03-03.

[4] Anon.˙ Optimus — Noesis Solutions — noesissolutions.com. `https://www.noesissolutions.com/our-products/optimus`. [Accessed 06-03-2025].

[5] Anon.˙ Simcenter HEEDS — plm.sw.siemens.com. `https://plm.sw.siemens.com/en-US/simcenter/integration-solutions/heeds/`. [Accessed 06-03-2025].

[6] Anon.˙ SmartUQ - AI for an Uncertain World — smartuq.com. `https://www.smartuq.com/`. [Accessed 06-03-2025].

[7] Etienne de Rocquigny, Nicolas Devictor, and Stefano Tarantola. *Uncertainty in Industrial Practice: A guide to Quantitative Uncertainty Management*. Wiley, April 2008.

[8] Dian-Qing Li, Te Xiao, Li-Min Zhang, and Zi-Jun Cao. Stepwise covariance matrix decomposition for efficient simulation of multivariate large-scale three-dimensional random fields. *Applied Mathematical Modelling*, 68:169–181, April 2019.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

BAYESIAN INVERSE UNCERTAINTY QUANTIFICATION FOR COMPUTATIONAL ENGINEERING USING SIMULATION-BASED INFERENCE

Maternus Herold[1,2,3], **Jan Boelts**[1]

[1] TransferLab, appliedAI Institute for Europe

[2] BMW Research and Development Center Munich, BMW Group

[3] Institute for Applied Mathematics and Scientific Computing, University of the Bundeswehr Munich

Abstract. *Inverse Uncertainty Quantification (IUQ) is crucial for analyzing complex systems in applied mechanics. Bayesian inference is a common approach to IUQ, as it allows for the systematic quantification of parameter uncertainty given prior knowledge and observations. However, many simulators and experiments in applied mechanics are analytically or computationally intractable, hindering the application of traditional Bayesian inference. To overcome this, we propose using Simulation-Based Inference (SBI), which circumvents the need for explicit likelihood calculations exploiting recent advances in probabilistic machine learning. We demonstrate the application of SBI for IUQ using the open-source* `sbi` *software package, showcasing its capabilities in an automotive passive safety example. We highlight the practical SBI workflow, including prior predictive checks, posterior inference, and posterior evaluation, and discuss key considerations for its application in computational mechanics.*

Keywords: Bayesian Inference, Inverse Uncertainty Quantification, Simulation-Based Inference, Neural Density Estimation, Automotive Passive Safety

1 INTRODUCTION

The target of robust and reliable system design in applied mechanics is hindered by the inherent presence of uncertainties across various facets, including material properties, geometric tolerances, and system parametrizations [26, 28, 37]. Underestimating these uncertainties can have sever consequences, particularly in safety-critical applications such as aerospace, civil engineering, and automotive passive safety. IUQ emerges as a critical methodology to address this challenge by inferring the distribution of uncertain system inputs from observed data. The ability of Bayesian inference to incorporate prior knowledge and systematically quantify uncertainty makes it a natural fit for IUQ, especially in mechanics, where prior knowledge of material properties and physical constraints is commonly available [25, 49, 50]. However, in applied mechanics models, computationally expensive simulations and the lack of an analytical likelihood often make classical approximate Bayesian methods—such as Markov Chain Monte Carlo or variational inference—impractical. To address this challenge, we propose using Simulation-Based Inference (SBI, [8]), which enables Bayesian inference without requiring an explicit likelihood. SBI trains neural networks as conditional density estimators [39] to approximate the posterior distribution directly from simulated data. This approach overcomes the limitations of likelihood-based methods in IUQ for applied mechanics, allowing systematic uncertainty estimation in model parameters.

SBI is an active research field that combines established methods with recent advances in probabilistic machine learning [8, 39, 12, 34]. However, practitioners often face challenges in applying these techniques, as research implementations are typically difficult to use and lack standardization [5]. To address this, several open source initiatives provide accessible and well-documented implementations of SBI methods, enabling broader adoption and reproducibility [47, 2, 41]. Such software packages play a crucial role in streamlining complex workflows, accelerating research cycles, and democratizing access to advanced inference techniques. By fostering collaboration and standardization, they ultimately contribute to more robust and insightful scientific outcomes.

In the following sections, we first discuss the application of Bayesian inference to IUQ in Section 2, highlighting the challenges posed by complex models and intractable likelihoods in applied mechanics, and thereby motivating the use of SBI. In Section 3, we introduce the concept of SBI, detailing its ability to overcome these challenges, and emphasize the need for versatile software packages that support the complete SBI workflow. Finally, in Section 4, we demonstrate the practical application of SBI for IUQ in an automotive passive safety example, showcasing the functionality and ease of use of the `sbi` toolkit [47, 2]. This numerical example illustrates the full SBI workflow, from prior predictive checks to posterior evaluation, using the `sbi` software package[1].

2 INVERSE UNCERTAINTY QUANTIFICATION VIA BAYESIAN INFERENCE

Relations between parameters and observations in mechanical systems are generally studied through the evaluation of a forward model $\mathcal{M} : \Theta \to \mathcal{X}$, where $\Theta \subseteq \mathbb{R}^d$ represents the parameter space and $\mathcal{X} \subseteq \mathbb{R}^m$ denotes the range of observable outputs. Depending on the field of application, the model $\mathcal{M}$ can be both, a simulation or a physical experiment. We denote the relationship between the parameters θ and observations $\mathbf{x}$ by the following equation:

[1]The `sbi` toolkit is an open-source Python package for the complete SBI workflow, described in Section 3. It is accessible via Python's package index and GitHub directly.

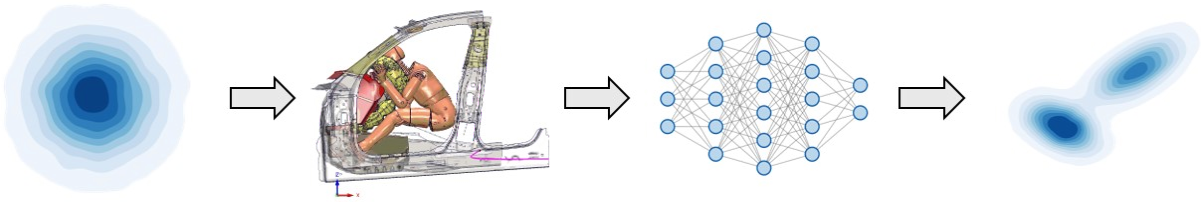


Figure 1: The general workflow of SBI involves sampling parameter values θ from the prior $\theta \sim p(\theta)$ and evaluating the stochastic model $\mathcal{M}$ on them, generating observations $\mathbf{x}$. Based on the generated data, a neural density estimator f_ϕ (see Section 3.2) is fitted, yielding a parametric posterior approximation $p_\phi(\theta \mid \mathbf{x}) \approx p(\theta \mid \mathbf{x})$.

$$\mathbf{x} = \mathcal{M}(\theta), \quad \theta \in \Theta, \mathbf{x} \in \mathcal{X}. \tag{1}$$

In many systems, noise and variability introduce uncertainty in both the parameters θ and the observations $\mathbf{x}$. This uncertainty can be described through a conditional distribution $p(\mathbf{x} \mid \theta)$, which captures how the system generates data given a set of parameters. In practice, however, the goal is often to estimate the parameters θ given observed data $\mathbf{x}$, while accounting for both system uncertainty and uncertainty of the parameter estimates. This is achieved by considering the inverse problem, i.e., obtaining the conditional distribution $p(\theta \mid \mathbf{x})$. This distribution, known as the *posterior*, is obtained using Bayes' theorem, which is given by

$$\begin{aligned} p(\theta \mid \mathbf{x}) &= \frac{p(\mathbf{x} \mid \theta) p(\theta)}{p(\mathbf{x})} \\ &\propto p(\mathbf{x} \mid \theta) p(\theta). \end{aligned} \tag{2}$$

In Equation (2), $p(\mathbf{x} \mid \theta)$ is the *likelihood*, $p(\theta)$ is the *prior* distribution on the parameters θ, and $p(\mathbf{x})$ is the marginal likelihood or evidence across the entire parameter space

$$p(\mathbf{x}) = \int_\Theta p(\mathbf{x} \mid \theta) p(\theta) \, d\theta. \tag{3}$$

The likelihood describes the probability of observing $\mathbf{x}$ given the parameters θ for the model $\mathcal{M}$. The prior distribution represents the initial belief about the parameters before observing the data. The marginal likelihood yields a constant normalization factor, which is generally a high-dimensional integral due to the dimensionality d of the parameter space Θ, making it intractable to compute exactly. Therefore, it is usually omitted and the posterior is obtained up to proportionality, as denoted in Equation (2), e.g., by obtaining posterior samples using established sampling algorithms like Markov Chain Monte Carlo (MCMC).

The approach to IUQ via the Bayesian inference is particularly suitable for applications in applied mechanics, as it provides a natural framework to incorporate expert knowledge and constraints on the parameters through the prior distribution. The likelihood (defined through the model) is then used to update the *prior beliefs* about the parameters by observing data under the assumed model, resulting in the posterior distribution. The posterior distribution provides a probabilistic description of *all* possible parameter values for yielding the observation conditioned on. Crucially, access to the full posterior covariance not only provides systematic uncertainty estimates for each parameter but also unveils their interdependencies. This allows engineers to understand which parameters exert the greatest influence on system behavior and

how they interact. This knowledge is particularly vital in applications such as vehicle safety, where the system's parameterization is constrained by its integration within a larger, more complex system.

Example 2.1 (Vehicle Restraint Systems)**.** *Consider the design of a vehicle restraint system $\mathcal{M}$, with the goal to minimize the risk of injury $\mathbf{x}$ of the occupants in the event of a collision (see Figure 2). The parameters of interest θ in this system include the deployment times of different airbags and the seatbelt characteristics. The posterior distribution $p(\theta \mid \mathbf{x})$ provides a probabilistic description of the uncertainty within the parameters, given the observed occupant safety $\mathbf{x}_o$.*

Additionally, different parameterizations $\theta_1, \theta_2, \ldots, \theta_n \in \Theta$ can lead to similar levels of occupant safety $\mathbf{x}_1 \approx \mathbf{x}_2 \approx \cdots \approx \mathbf{x}_n$, $\mathcal{M}(\theta_i) = \mathbf{x}_i \in \mathcal{X}$. The posterior distribution should ideally cover all of such solutions. In a downstream task, the different parameterizations might be assessed based on their cost or robustness [25].

While Bayesian inference provides a principled approach to IUQ, its direct application in applied mechanics is challenging. The likelihood function $p(\mathbf{x} \mid \theta)$ of a system $\mathcal{M}$ under consideration is usually not analytically available. This is due to such systems being complex mechanical models or simulations that can only be evaluated as black-box functions. Classical access to posterior samples via MCMC methods is therefore technically infeasible in the case of simulations and economically infeasible for many hardware tests [26, 28, 45]. Moreover, downstream tasks, such as those in Example 2.1, require evaluating and sampling the posterior for many observations $\mathbf{x}$. To enable this efficiently, a full parametric approximation—ideally in an amortized form—is desired, allowing rapid evaluation and sampling for any given observation.

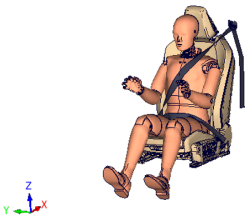


Figure 2: Illustration of a vehicle restraint system simulation depicting a crash test dummy protected by a seatbelt in a full frontal crash. Further restraint systems such as the airbags and the car's chassis are removed from the figure to reduce occlusion. Figure adapted from [26].

SBI offers a solution to these challenges. It approximates the posterior using data simulated from parameters sampled from the prior, without requiring an explicit likelihood function. Additionally, by performing conditional density estimation with neural networks, it enables amortized inference and efficient evaluation and sampling across many observations [8, 40, 18]. Unlike traditional MCMC methods, which have convergence guarantees, SBI is approximate in nature and requires careful evaluation. In the following section, we will explain the details of this approach and how it addresses the challenges in applied mechanics.

3 SIMULATION-BASED-INFERENCE

Simulation-based inference (SBI) enables Bayesian parameter estimation in scenarios where the likelihood function is intractable and has been used in various fields, such as astrophysics [9, 10], neuroscience [17, 20, 4, 3], or mechanical engineering [26, 28]. Unlike traditional methods that require an explicit likelihood function, SBI leverages methods from probabilistic machine learning to perform Bayesian inference for simulation-based models [8].

Early SBI approaches such as Approximate Bayesian Computation (ABC, [44]) address this challenge by simulating data and accepting parameters that yield close matches to observations.

However, ABC relies on comparing simulated data to observations using distance metrics and accepting parameters only if they fall below a user-defined threshold, leading to inefficiency, especially in high-dimensional problems. More recent neural network-based SBI methods improve upon these methods by using neural network-based density estimation techniques to efficiently approximate the posterior or the likelihood [39, 18, 10], making it particularly suitable for cases with high-dimensional dependencies and complex posterior shapes, such as multimodal distributions [8, 34].

The following section outlines the general workflow of SBI, starting with the most widely used method, Neural Posterior Estimation (NPE), which directly approximates the posterior distribution. We then compare it with Neural Likelihood Estimation (NLE), which learns the likelihood function, and Neural Ratio Estimation (NRE), which estimates the likelihood ratio, highlighting their trade-offs. In addition, we discuss performance evaluation, potential failure modes, such as poor posterior coverage, and the importance of robust software implementations to ensure reliability in practice.

The application of Bayesian inference, as outlined in the established Bayesian workflow by Gelman et al. [16], typically involves a sequence of steps: model specification, prior definition, posterior computation, and model checking. This general framework is also applicable to simulation-based inference (SBI) [5] and it aligns closely with the common steps employed in inverse uncertainty quantification (IUQ). Specifically, for IUQ via Bayesian inference using SBI, one first conducts *a priori checks* to verify the suitability of the problem and data. Subsequently, a specific SBI method is selected and applied. Finally, *a posteriori checks* and performance evaluations are performed to assess the quality and validity of the inference results.

3.1 A Priori Checks

In order to obtain valid inference results on observed data, the setup of the prior and simulator have to be assessed. While we generally assume a slight deviation of the model from the true data-generating process, the estimation should be able to reproduce the most important aspects of the data. This is known as the *model misspecification* problem. For example, in a finite element model, neglecting certain material nonlinearities could lead to model misspecification. While several methods to automatically detect and account for model misspecification in SBI have been proposed [14, 48, 29, 43], they often require additional computational resources. We therefore focus on the *prior predictive check*, which is a well interpretable and easy to implement alternative to assess the misspecification.

Prior predictive checks are conducted to assess whether the chosen prior distributions for a model's parameters are reasonable and result in plausible data w.r.t. obtained observations $\mathbf{x}_o$. The process involves defining prior distributions $p'(\theta)$ for each model parameter $\theta \in \Theta$, incorporating any available domain knowledge and system constraints. New data is then simulated by sampling parameter values from these prior distributions and then generating datasets using the model with the sampled parameters, see Algorithm 1. This simulation creates the prior predictive distribution

$$p'(\mathbf{x}) = \int p(\mathbf{x} \mid \theta) p'(\theta) d\theta \approx \int \mathcal{M}(\theta) p'(\theta) d\theta. \tag{4}$$

The data sampled from the prior predictive distribution $\mathbf{x}' \sim p'(\mathbf{x})$ is compared to the actual observed data $\mathbf{x}_o$. If the distribution of the simulated data is realistic and contains the observed data, the priors are considered appropriate. If the simulated data is unrealistic or fails to include the observed data, the priors must be adjusted. Depending on the dimensionality of the data,

visual checks, summary statistics like mean and variance, or statistical tests may be required. Even if the simulated and observed data do not match perfectly, the priors can still be acceptable if the simulated data is generally plausible. Note that a discrepancy can also suggest an issue with the simulation model itself, in which case more advanced techniques for identifying the misspecification need to be employed [43, 27]. Passing the prior predictive check allows one to proceed with the training of the neural density estimator.

3.2 Posterior Approximation

At the core of neural SBI stands the idea of using access to evaluate the forward simulation model (Equation (1), $\mathcal{M}$ ($\mathcal{M}(\theta) = \mathbf{x}$) to generate synthetic data $\mathbf{x}$ and bypass the requirement of analytical likelihood for traditional Bayesian inference [42]. The three most popular approaches are NPE [39], approximating the posterior distribution $p(\theta \mid \mathbf{x})$ directly, NLE [38], approximating the likelihood $p(\mathbf{x} \mid \theta)$, and NRE [22], learning the likelihood-to-evidence ratio $r(\theta, \mathbf{x}) = \frac{p(\mathbf{x}|\theta)}{p(\mathbf{x})}$. For all three approaches, a dataset from the joint distribution $\mathcal{D} = \{(\theta_i, \mathbf{x}_i)\}_{i=1}^N$, where $(\theta_i, \mathbf{x}_i) \sim p(\theta, \mathbf{x})$, is required to train the neural density estimator f_ϕ. In order to obtain the tuples $(\theta_i, \mathbf{x}_i)$, the forward model $\mathcal{M}$ is evaluated on a set of parameter values $\theta_i \sim p(\theta)$, as outlined in Algorithm 1.

Algorithm 1: Sampling from the joint distribution.

Input: Prior distribution $p(\theta)$, simulation model $\mathcal{M} : \theta \mapsto x$, number of samples N
Output: Sampled dataset from the joint distribution
$\mathcal{D} = \{(\theta_i, \mathbf{x}_i) \mid (\theta_i, \mathbf{x}_i) \sim p(\theta, \mathbf{x}), i = 1, \ldots, N\}$
Initialize an empty list of samples $\mathcal{D} \leftarrow [\,]$;
for $i \leftarrow 0$ **to** N **do**
  Sample parameter $\theta_i \sim p(\theta)$;
  Generate simulated data $x_i \leftarrow \mathcal{M}(\theta_i)$;
  Append (θ_i, x_i) to the list of samples $\mathcal{D}$;
end

In a subsequent step, a neural density estimator f_ϕ is trained on the dataset $\mathcal{D}$ to yield the approximate conditional distribution p_ϕ, in the case of NPE and NLE, or train a classifier, in the case of NRE. In the following, we will focus on NPE, as it is the most popular and widely used method by the community. The approaches of NLE and NRE run analogously with either a different targeted distribution or training task.

The fit of the neural density estimator is achieved by minimizing the forward Kullback-Leibler (KL) divergence between the true and approximate posterior distribution

$$\mathrm{KL}(p(\theta \mid \mathbf{x}) \,||\, p_\phi(\theta \mid \mathbf{x})) = \int p(\theta \mid \mathbf{x}) \log \left(\frac{p(\theta \mid \mathbf{x})}{p_\phi(\theta \mid \mathbf{x})} \right) d\theta. \tag{5}$$

The KL-divergence cannot be optimized directly as we generally don't have access to the true posterior $p(\theta \mid \mathbf{x})$. In order to minimize the KL-divergence implicitly, the negative log likelihood (NLL) is used as a surrogate loss function [1, 38]

$$\mathcal{L}_{\mathrm{NLL}}(\phi) = -\int p(\theta, \mathbf{x}) \log p_\phi(\theta \mid \mathbf{x}) d\theta \approx -\frac{1}{N} \sum_{i=1}^{N} \log f_\phi(\theta_i \mid \mathbf{x}_i). \tag{6}$$

As the approximate posterior $p_\phi(\theta \mid \mathbf{x})$ is encoded by a neural network, the conditional distribution for arbitrary observations $\mathbf{x}'$ can be obtained by evaluating the neural network on the new data point without retraining. This is commonly referred to as *amortized* inference as the initial high cost of training the density estimator is amortized later on [8, 39]. The steps taken to obtain a posterior approximation via NPE are summarized in Figure 1.

3.3 Quality Assessment of the Posterior Approximation

Evaluating the quality of a posterior approximation is crucial for ensuring reliable inference and security in downstream tasks. We will discuss a selection of these, emphasizing their relevance to applied mechanics.

To assess the discrepancy between the approximated and true posteriors, *sample-based tests* are commonly employed. Methods such as the *classifier two-sample test* (C2ST) [15, 32, 21], Maximum Mean Discrepancy (MMD, [19]), and Kernelized Stein Discrepancy [31] compare samples from the approximated posterior with (samples from the) true posterior. However, these methods require access to the true posterior, which is typically unknown in real-world applications. Consequently, their primary utility lies in evaluating toy models or benchmarking scenarios, as seen in works like Lueckmann et al. [34].

In contrast to two-sample tests, *Posterior Predictive Checks* (PPC) offer a practical means of evaluating the inference approximation that can performed without access to a reference posterior. PPC compares observed data to data generated from the posterior predictive distribution:

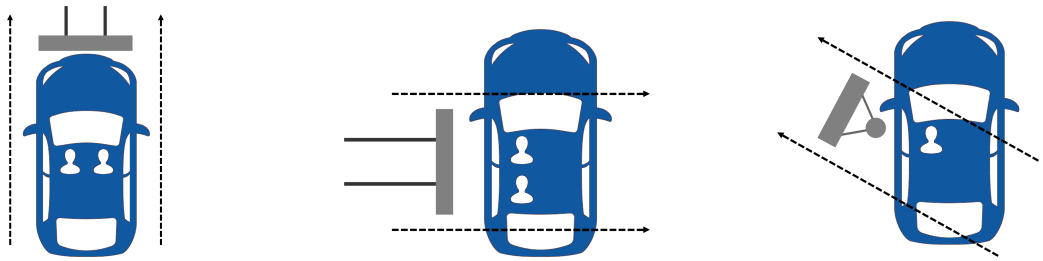
$$p(\mathbf{x}' \mid \mathbf{X}_o) = \int p(\mathbf{x}' \mid \theta) p(\theta \mid \mathbf{X}_o) d\theta \approx \int \mathcal{M}(\theta) p(\theta \mid \mathbf{X}_o) d\theta, \tag{7}$$

Samples from the posterior predictive distribution are obtained by sampling parameters from the approximated posterior and simulating (predictive) data $\mathbf{x}'$ with those parameters. The check is successful if the observed data $\mathbf{X}_o$ falls within the predictive distribution. Typically, this is checked by visual inspection or by calculating distance metrics between predictive and observed data. If the noise of the simulation model is known, the variance of the samples around the observation should match it. In computational mechanics, PPC helps assess whether the model accurately reproduces observed phenomena, such as deformation patterns or stress distributions.

Although PPC and C2ST are valuable tools for assessing the accuracy of the approximation, it is essential to additionally check whether the uncertainties represented by the posterior are well-calibrated. As Hermans et al. [23] have shown, estimators with high accuracy can still be poorly calibrated. To this end, simulation-based calibration techniques [7] allow evaluation of posterior statistical calibration without requiring access to the reference posterior. Essentially, these checks repeat the inference procedure many times on a simulated test set of parameters and corresponding (simulated) observed data, and check the coverage of the test parameters under the approximated posteriors. Common methods include *Simulation-Based Calibration* of the posterior marginals (SBC, [46]), calculation of expected coverage across all posterior dimensions [11, 13] or *Tests of Accuracy with Random Points* (TARP, [30]).

3.4 Trade-offs Between different SBI Methods

The three simulation-based inference (SBI) approaches—Neural Posterior Estimation (NPE), Neural Likelihood Estimation (NLE), and Neural Ratio Estimation (NRE)—address distinct learning problems and exhibit varying strengths in practical applications. NPE directly regresses the posterior density over the parameter space, conditioned on the observation $\mathbf{x}$. NLE,



(a) Full frontal. (b) Side impact. (c) Side impact.

Figure 3: Illustration of crash scenarios in the US NCAP protocol, according to [6]. In the full frontal crash, two crash dummies are seated in the front and the vehicle is accelerated to 56 km/h prior to impact. In the scenario of a side crash, the car is hit by a moving barrier at 55 km/h. In the last scenario, the vehicle impacts a pole at 32 km/h with one crash dummy in the driver's seat. Figure adapted from [24].

conversely, regresses the likelihood density over the observation space, conditioned on the parameter θ. NRE, unlike the others, does not perform density estimation; instead, it solves a classification problem to estimate the likelihood-to-evidence ratio over the joint parameter-observation space, taking both θ and $\mathbf{x}$ as input. This classification approach can offer computational benefits compared to density estimation.

The choice of approach depends significantly on the dimensionalities of the parameter space, $\dim(\theta)$, and the observation space, $\dim(\mathbf{x})$. When $\dim(\theta) \ll \dim(\mathbf{x})$, NPE may be preferred as it learns a density in a lower-dimensional space compared to NLE. Conversely, if $\dim(\theta) \gg \dim(\mathbf{x})$, NLE and NRE may be more suitable. However, while NLE and NRE can simplify the learning task, they require sampling techniques such as Markov Chain Monte Carlo (MCMC) or variational inference to obtain posterior samples for each new observation $\mathbf{x}$. Critically, only the training phase is amortized in NLE and NRE; posterior sampling is required for every new observation. Furthermore, MCMC methods do not allow for the direct evaluation of the posterior log probability (except for some variants of NRE [36]). In contrast, NPE is fully amortized, enabling direct sampling and evaluation of the posterior density through the learned neural network. This full amortization is particularly advantageous for simulation-based calibration methods, as it facilitates rapid repetition of inference across a test set, eliminating the need for individual sampling runs. While NLE and NRE can be beneficial when the parameter space is small relative to the observation space, MCMC in high-dimensional parameter spaces can be challenging.

For inference targeted at a specific observation $\mathbf{x}_o$, sequential or active learning approaches can improve efficiency [39, 35, 18, 33]. However, these methods introduce greater algorithmic complexity, require careful hyperparameter selection, and often demand use-case-specific tuning [34].

Following this overview of SBI methods, we will apply NPE, NLE, and NRE to a problem from automotive vehicle safety using the `sbi` toolkit [2, 47] in the subsequent section.

4 NUMERICAL EXAMPLES

To illustrate the application of SBI for IUQ in computational mechanics, we utilize an efficient simulator, allowing us to perform a substantial number of simulations without incurring excessive computational costs. Our primary objective is to showcase the versatility and ease of use of SBI via the `sbi` software package, rather than conducting a comprehensive benchmark

of different SBI methods for this specific task. For a detailed comparative analysis of various SBI methods on a set of benchmark tasks we refer the reader to Lueckmann et al. [34].

4.1 Problem Description

We employ a simulator based on the US New Car Assessment Program (US NCAP), as detailed in Herold et al. [24], to demonstrate the use of SBI for IUQ. The US NCAP encompasses a range of vehicle safety tests, collecting data from sensors on crash test dummies, vehicles, barriers, and advanced driver assistance systems. This data serves to assess the risk of severe injuries to vehicle occupants and other parties involved in accidents. An illustration of the crash scenarios is provided in Figure 3.

In this first problem, we focus exclusively on the crash tests within the US NCAP framework, which assess the risk of injury across a series of scenarios based on measurements collected from crash test dummies. The problem is modeled as follows:

$$\mathbf{x} = \mathcal{M}_{\text{US NCAP}}(\theta) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2), \tag{8}$$

where $\theta \in \Theta_{\text{US NCAP}} \subset \mathbb{R}^{21}$ represents a relatively high-dimensional parameter space, describing simulated aggregations of sensor measurements. Conversely, $\mathbf{x}_{\text{US NCAP}} \in \mathcal{X}_{\text{US NCAP}} \subset \mathbb{R}$ indicates a low-dimensional (scalar) data space, representing the final security rating. The additive Gaussian noise, ϵ, can be controlled by adjusting the variance parameter σ. Note that the simulator implements a strongly non-linear reduction in dimensionality, mapping the 21-dimensional vehicle parameters to a single, scalar security rating. This reduction is intentional, as the security rating is the primary metric of interest. However, as a consequence, the US NCAP problem might exhibit numerous complementary solutions distributed across the parameter space. Therefore, the posterior distribution must encompass a substantial portion of the parameter space and can possess a complex shape, as described by Herold et al. [24].

4.2 A Priori Checks

Before proceeding with posterior inference, we briefly address a priori checks. In this fully simulated scenario, the simulated observations naturally fall within the prior predictive distribution, confirming the suitability of our prior distributions (Figure 4). While this check is trivial in this context, it is a crucial first step in practical applications with real-world data, where validating the chosen prior distributions and ensuring the suitability of the simulation model is essential.

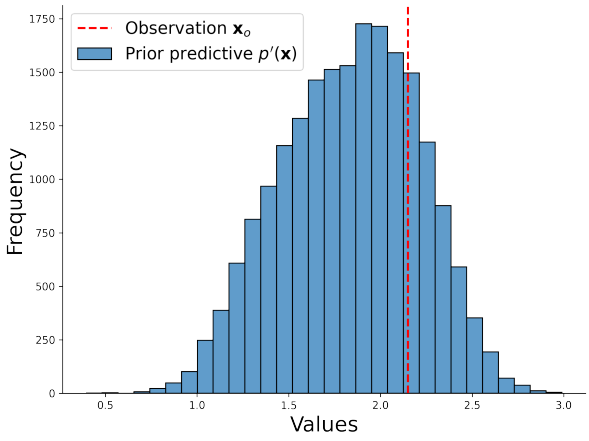


Figure 4: Prior predictive check showing the empirical distribution from samples of the prior predictive distribution Equation (4) and an observation $\mathbf{x}_o$ on top.

4.3 Posterior Approximation

After passing the a-priori check, we proceeded with the actual inference part and used NPE, NLE, and NRE to perform SBI on the US NCAP problem. To evaluate the impact of the size of the training dataset, each algo-

rithm was trained three times with different dataset sizes $N \in \{10^3, 10^4, 10^5\}$. Training was conducted until convergence as measured by a moving average of the loss on a held-out validation dataset of the same size as the training dataset. Additionally, a separate test dataset was generated for later comparison across training budgets and SBI methods (see Listing 1).

```
import sbi
import us_ncap_simulator

prior = sbi.util.torchutils.BoxUniform(...)
simulator = us_ncap_simulator.get_simulator()

thetas, thetas_val = prior.sample((N,)), prior.sample((N,))
xs, xs_val = simulator(thetas), simulator(thetas_val)

thetas_test = prior.sample((1_000,))
xs_test = simulator(thetas_test)
```

Listing 1: Exemplary code for data generation.

The posterior approximation was performed using the inference API provided by the `sbi` package (see Listing 2). When instantiating the inference object, the prior distribution must be specified for subsequent inference. Additionally, various configurations governing the inference process can be directly passed, such as the sampling method for MCMC-based inference in the case of NLE and NRE. By simply interchanging the inference method—i.e., replacing `NPE` with `NLE` or `NRE`—the posterior distribution can be approximated according to the different methods.

```
import sbi

inference = sbi.inference.NPE(prior=prior, ...)
inference = inference.append_simulations(thetas, xs)

density_estimator = inference.train(...)

posterior = inference.build_posterior(density_estimator)
```

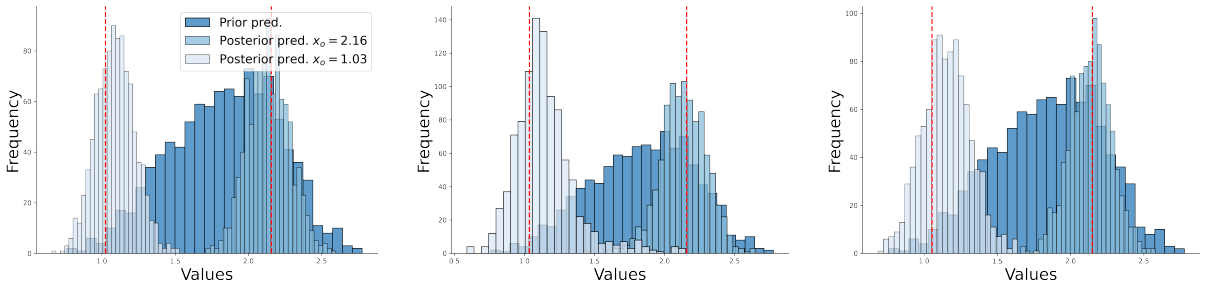
Listing 2: Exemplary code for neural posterior estimation using `sbi`.

```
from src.utils import ppc

posterior_samples = posterior.sample_batched((1000,), xs_test)

posterior_predictive_samples = simulator(posterior_samples)

median_dist = ppc(posterior_predictive_samples, xs_test)
```

Listing 3: Exemplary code for computing the PPC

4.4 Performance evaluation

We used several metrics to evaluate inference performance, tailored to each SBI method. For direct comparison between all methods and training budgets, we assessed PPC performance through visual comparison of prior and posterior predictive distributions and by calculating the median distance between test data and data generated from samples of the approximate posterior (see Listing 3). Additionally, we conducted calibration checks for all methods and budgets using SBC (see Listing 4). SBC can be performed for all marginals separately via ranking of



(a) NPE (b) NLE (c) NRE

Figure 5: Visual Posterior predictive check (PPC) for the US NCAP task using the three methods NPE, NLE, and NRE with training data budget of $N = 10^5$. Each subplot shows the prior predictive distribution as seen in Figure 4, two randomly chosen observations $\mathbf{x}_o^{(1)}$ and $\mathbf{x}_o^{(2)}$ (vertical lines), and the respective posterior predictive distributions.

parameter values [46] or via expected coverage using the log-probability of the approximate posterior [11]. We calculated SBC via the marginals in order to enable comparison across all three SBI methods, as NLE and NRE do not allow evaluation of the posterior.

```
    from sbi.analysis.plot import sbc_rank_plot
    from sbi.diagnostics import run_sbc

    ranks, _ = run_sbc(thetas_test, xs_test, posterior, \dots)

    # Plot the SBC as empirical CDFs.
    fig, ax = sbc_rank_plot(ranks, num_posterior_samples=1000, num_bins=50)
```

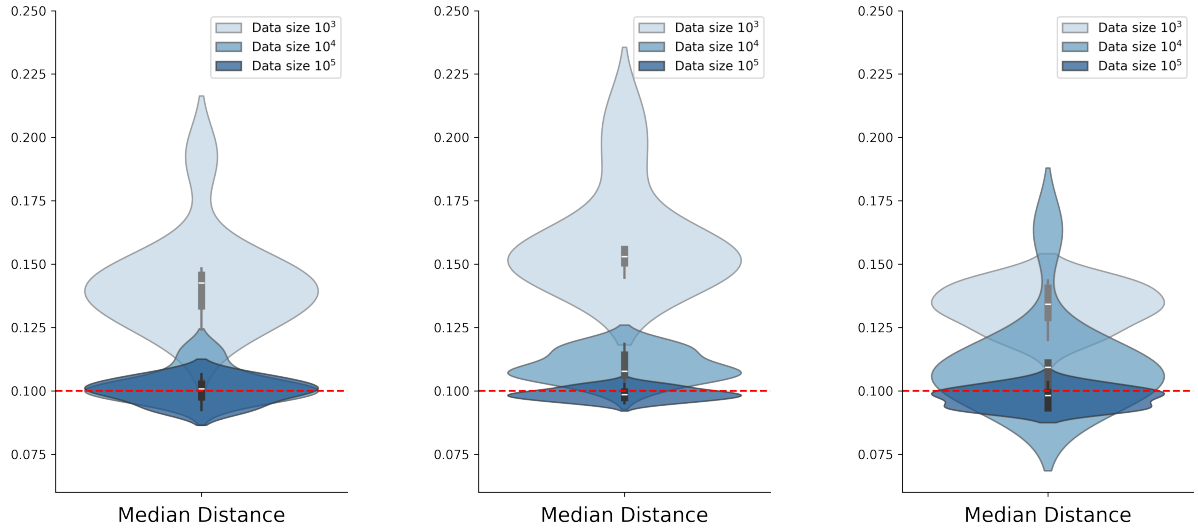
Listing 4: Exemplary code for performing SBC

Notably, all these auxiliary functions are readily available within the `sbi` package. Furthermore, the package provides additional functionality for quality analysis, including (local) classifier two-sample tests, SBC, TARP, pair plots, sensitivity analysis, and conditional density plots. For a comprehensive list of available functions, we direct the reader to the package documentation[2].

4.4.1 Posterior predictive performance

The visual Posterior Predictive Checks (PPC) demonstrated that the predictive distributions generated by all three SBI methods were concentrated around the observed data on which the posterior was conditioned (Figure 5). This indicates that the simulated data from the posterior predictive distributions closely matched the observed data. The more systematic evaluation of the median distances between the test data point and posterior predictive samples confirmed this observation, showing an improvement with increasing simulation budget and converging towards the reference value (Figure 6). In this comparison, NRE demonstrated slightly faster convergence in predictive performance compared to the other two methods.

[2]https://sbi-dev.github.io/sbi



(a) NPE (b) NLE (c) NRE

Figure 6: Posterior predictive check (PPC) for the US NCAP task using the three methods NPE, NLE, and NRE with training data sizes of $N \in \{10^3, 10^4, 10^5\}$. The plot shows the median distance between the posterior predictive samples and the true data for 200 test points with 10^3 samples from the posterior predictive, each. The closer the median distance to zero, the better the posterior approximation. Note, the variance σ^2 of the additive white noise of the simulator was set to 0.1, depicted by the horizontal red line.

4.4.2 Posterior calibration performance

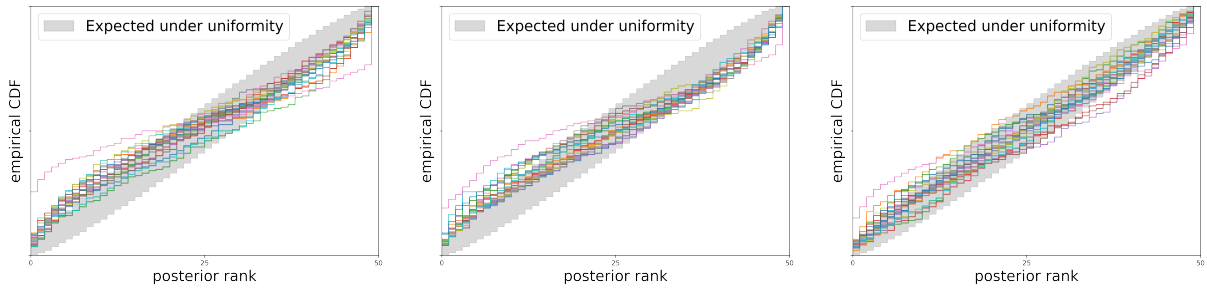
SBC results indicated that for NLE and NRE, all marginal distributions were well-calibrated. For NPE, one marginal distribution exhibited slight overconfidence, characterized by an excessively narrow credible interval (Figure 7). However, with increasing training budget, this overconfidence reduced, indicating that NPE might require more training data to achieve a well-calibrated posterior. We provide the code for all the experiments conducted for this work as repository on GitHub[3].

5 CONCLUSION

In computational mechanics, accurately quantifying uncertainties in material properties, geometric tolerances, and system parametrizations is essential for robust and reliable system design. Inverse Uncertainty Quantification (IUQ) is a crucial methodology for addressing these uncertainties, particularly in safety-critical applications. However, classical Bayesian inference methods often prove impractical for complex, computationally expensive models with intractable likelihoods. Simulation-Based Inference (SBI) offers a powerful alternative, enabling Bayesian inference without requiring explicit likelihood calculations.

This work demonstrated the application of Simulation-Based Inference (SBI) for Inverse Uncertainty Quantification (IUQ) within a Bayesian framework. A simplified example problem, based on US New Car Assessment Program (US NCAP) automotive safety simulations, was employed to illustrate the functionality of the sbi toolkit. Due to the absence of an analytically tractable likelihood function, traditional Bayesian inference methods were not applicable to

[3]Code provided at `github.com/aai-institute/uncecomp-2025-iuq-via-sbi`.



(a) 10^3 Training samples (b) 10^4 Training samples (c) 10^5 Training samples

Figure 7: SBC results of NPE per data size. In line with the PPC, 200 test samples with 10^3 posterior samples, each, are used to compute the ranks. A perfectly calibrated posterior would yield a uniform distribution of ranks. The 99% confidence interval of a uniform distribution is denoted by the gray-shaded area.

this problem. However, SBI facilitated the implementation of the standard Bayesian workflow, encompassing prior predictive checks, posterior inference, and posterior evaluation.

The results obtained from this example demonstrated general SBI workflow and showed a good predictive performance for the US NCAP model. However, we observed that the inference process resulted in only a slight constraint of the parameters, indicating that the posterior distribution remained relatively close to the prior. This is likely due to the low dimensionality of the data output compared to the high dimensionality of the parameter space (e.g., 1 vs. 21). To achieve a more constrained posterior distribution, it would be necessary to modify the simulator to generate a higher-dimensional data output. Additionally, the calibration checks revealed a slight mis-calibration for the NPE method, indicating the need for more training data to obtain fully calibrated posterior estimate. The fact that this occurred only for NPE can be explained by the difference in learning problems solved by the different approaches (see Section 3.4, e.g., NPE has to estimate a 21-dimensional posterior and NLE has to estimate a 1-dimensional likelihood.

More generally, in practical applications involving complex simulators, high-dimensional parameter spaces, or data outputs, SBI methodologies must be carefully adapted and tailored to the specific use case. This may involve sophisticated neural network architectures, advanced sampling techniques, or customized evaluation metrics. However, it is worth noting that the SBI field benefits from a very active and collaborative community of researchers and open-source developers, which can significantly aid in addressing these challenges.

In summary, SBI presents a viable and developing technique for IUQ in computational mechanics and related fields. Its capacity to handle complex models and intractable likelihoods offers new possibilities for uncertainty quantification, facilitating the design of more robust and reliable systems. Further advancements in SBI methodologies and their application across engineering disciplines are anticipated.

Acknowledgements

The authors would like to thank the community of developers behind the sbi toolkit. The authors also thank the TransferLab at the appliedAI Institute for Europe for supporting the research and submission. MH thanks the BMW Group for supporting the research, especially Dr. Jonas S. Jehle, and Prof. Dr. Matthias Gerdts from the University of the Bundeswehr Munich for their valuable discussions on the matter.

REFERENCES

[1] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer, New York, 2006.

[2] Jan Boelts, Michael Deistler, Manuel Gloeckler, Álvaro Tejero-Cantero, Jan-Matthis Lueckmann, Guy Moss, Peter Steinbach, Thomas Moreau, Fabio Muratore, Julia Linhart, Conor Durkan, Julius Vetter, Benjamin Kurt Miller, Maternus Herold, Abolfazl Ziaeemehr, Matthijs Pals, Theo Gruner, Sebastian Bischoff, Nastya Krouglova, Richard Gao, Janne K. Lappalainen, Bálint Mucsányi, Felix Pei, Auguste Schulz, Zinovia Stefanidi, Pedro Rodrigues, Cornelius Schröder, Faried Abu Zaid, Jonas Beck, Jaivardhan Kapoor, David S. Greenberg, Pedro J. Gonçalves, and Jakob H. Macke. Sbi reloaded: A toolkit for simulation-based inference workflows, November 2024.

[3] Jan Boelts, Philipp Harth, Richard Gao, Daniel Udvary, Felipe Yáñez, Daniel Baum, Hans-Christian Hege, Marcel Oberlaender, and Jakob H. Macke. Simulation-based inference for efficient identification of generative models in computational connectomics. *PLOS Computational Biology*, 19(9):e1011406, September 2023.

[4] Jan Boelts, Jan-Matthis Lueckmann, Richard Gao, and Jakob H Macke. Flexible and efficient simulation-based inference for models of decision-making. *eLife*, 11:e77220, July 2022.

[5] Jan Bölts. *Advancing Methods and Applicability of Simulation-Based Inference in Neuroscience*. PhD thesis, Universität Tübingen, July 2023.

[6] carhs. SafetyCompanion 2024. 2024.

[7] Samantha R. Cook, Andrew Gelman, and Donald B. Rubin. Validation of Software for Bayesian Models Using Posterior Quantiles. *Journal of Computational and Graphical Statistics*, 15(3):675–692, 2006.

[8] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, December 2020.

[9] Miles D. Cranmer, Richard Galvez, Lauren Anderson, David N. Spergel, and Shirley Ho. Modeling the Gaia Color-Magnitude Diagram with Bayesian Neural Flows to Constrain Distance Estimates, August 2019.

[10] Maximilian Dax, Jonas Wildberger, Simon Buchholz, Stephen R. Green, Jakob H. Macke, and Bernhard Schölkopf. Flow Matching for Scalable Simulation-Based Inference, May 2023.

[11] Michael Deistler, Pedro J. Goncalves, and Jakob H. Macke. Truncated proposals for scalable and hassle-free simulation-based inference, November 2022.

[12] Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural Spline Flows. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[13] Maciej Falkiewicz, Naoya Takeishi, Imahn Shekhzadeh, Antoine Wehenkel, Arnaud Delaunoy, Gilles Louppe, and Alexandros Kalousis. Calibrating Neural Simulation-Based Inference with Differentiable Coverage Probability, October 2023.

[14] David T. Frazier and Christopher Drovandi. Robust Approximate Bayesian Inference With Synthetic Likelihood. *Journal of Computational and Graphical Statistics*, 30(4):958–976, October 2021.

[15] J H Friedman. On multivariate goodness of fit and two sample testing. 2003.

[16] Andrew Gelman, Aki Vehtari, Daniel Simpson, Charles C. Margossian, Bob Carpenter, Yuling Yao, Lauren Kennedy, Jonah Gabry, Paul-Christian Bürkner, and Martin Modrák. Bayesian Workflow, November 2020.

[17] Pedro J Gonçalves, Jan-Matthis Lueckmann, Michael Deistler, Marcel Nonnenmacher, Kaan Öcal, Giacomo Bassetto, Chaitanya Chintaluri, William F Podlaski, Sara A Haddad, Tim P Vogels, David S Greenberg, and Jakob H Macke. Training deep neural density estimators to identify mechanistic models of neural dynamics. *eLife*, 9:e56261, September 2020.

[18] David S. Greenberg, Marcel Nonnenmacher, and Jakob H. Macke. Automatic Posterior Transformation for Likelihood-Free Inference, May 2019.

[19] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A Kernel Two-Sample Test. *Journal of Machine Learning Research*, 13(25):723–773, 2012.

[20] Lukas N. Groschner, Jonatan G. Malis, Birte Zuidinga, and Alexander Borst. A biophysical account of multiplication by a single neuron. *Nature*, 603(7899):119–123, March 2022.

[21] Michael U. Gutmann, Ritabrata Dutta, Samuel Kaski, and Jukka Corander. Likelihood-free inference via classification. *Statistics and Computing*, 28(2):411–425, March 2018.

[22] Joeri Hermans, Volodimir Begy, and Gilles Louppe. Likelihood-free MCMC with Amortized Approximate Ratio Estimators, June 2020.

[23] Joeri Hermans, Arnaud Delaunoy, François Rozet, Antoine Wehenkel, Volodimir Begy, and Gilles Louppe. A Trust Crisis In Simulation-Based Inference? Your Posterior Approximations Can Be Unfaithful, December 2022.

[24] Maternus Herold, Jonas Siegfried Jehle, and Matthias Gerdts. Tractable Inverse Uncertainty Quantification via Probabilistic Circuits for Applied Mechanics. *under review*.

[25] Maternus Herold, Jonas Siegfried Jehle, and Matthias Gerdts. Inverse Uncertainty Quantification in Passive Automobile Safety: Bayesian Inversion for Optimal Safety Ratings with Robustness Quantification. In *9th European Congress on Computational Methods in Applied Sciences and Engineering*, Lisbon, October 2024.

[26] Maternus Herold, Anna Veselovska, Jonas Jehle, and Felix Krahmer. Non-intrusive surrogate modelling using sparse random features with applications in crashworthiness analysis. *CoRR*, abs/2212.14507, 2022.

[27] Daolang Huang, Ayush Bharti, Amauri Souza, Luigi Acerbi, and Samuel Kaski. Learning Robust Statistics for Simulation-based Inference under Model Misspecification, May 2023.

[28] Jonas Siegfried Jehle. *Uncertainty Management Framework for Automotive Crash Applications*. PhD thesis, Universität der Bundeswehr München, 2022.

[29] Ryan P. Kelly, David J. Nott, David Tyler Frazier, David J. Warne, and Christopher Drovandi. Misspecification-robust Sequential Neural Likelihood for Simulation-based Inference. *Transactions on Machine Learning Research*, March 2024.

[30] Pablo Lemos, Adam Coogan, Yashar Hezaveh, and Laurence Perreault-Levasseur. Sampling-Based Accuracy Testing of Posterior Estimators for General Inference, June 2023.

[31] Qiang Liu, Jason Lee, and Michael Jordan. A Kernelized Stein Discrepancy for Goodness-of-fit Tests. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 276–284. PMLR, June 2016.

[32] David Lopez-Paz and Maxime Oquab. Revisiting Classifier Two-Sample Tests. In *International Conference on Learning Representations*, February 2017.

[33] Jan-Matthis Lueckmann, Giacomo Bassetto, Theofanis Karaletsos, and Jakob H. Macke. Likelihood-free inference with emulator networks. In *Proceedings of The 1st Symposium on Advances in Approximate Bayesian Inference*, pages 32–53. PMLR, January 2019.

[34] Jan-Matthis Lueckmann, Jan Boelts, David S. Greenberg, Pedro J. Gonçalves, and Jakob H. Macke. Benchmarking Simulation-Based Inference, April 2021.

[35] Jan-Matthis Lueckmann, Pedro J. Goncalves, Giacomo Bassetto, Kaan Öcal, Marcel Nonnenmacher, and Jakob H. Macke. Flexible statistical inference for mechanistic models of neural dynamics. arXiv, November 2017.

[36] Benjamin Kurt Miller, Christoph Weniger, and Patrick Forré. Contrastive Neural Ratio Estimation, October 2022.

[37] Joseph B. Nagel and Bruno Sudret. Bayesian Multilevel Model Calibration for Inverse Problems Under Uncertainty with Perfect Data. *Journal of Aerospace Information Systems*, 12(1):97–113, January 2015.

[38] George Papamakarios. Neural Density Estimation and Likelihood-free Inference, October 2019.

[39] George Papamakarios and Iain Murray. Fast ϵ-free Inference of Simulation Models with Bayesian Conditional Density Estimation. 2016.

[40] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[41] Stefan T. Radev, Marvin Schmitt, Lukas Schumacher, Lasse Elsemüller, Valentin Pratz, Yannik Schälte, Ullrich Köthe, and Paul-Christian Bürkner. Bayesflow: Amortized bayesian workflows with neural networks. *Journal of Open Source Software*, 8(89):5702, 2023.

[42] Donald B. Rubin. Bayesianly Justifiable and Relevant Frequency Calculations for the Applied Statistician. *The Annals of Statistics*, 12(4):1151–1172, December 1984.

[43] Marvin Schmitt, Paul-Christian Bürkner, Ullrich Köthe, and Stefan T. Radev. Detecting Model Misspecification in Amortized Bayesian Inference with Neural Networks: An Extended Investigation, June 2024.

[44] Scott A. Sisson, Yanan Fan, and Mark Beaumont. *Handbook of Approximate Bayesian Computation*. CRC Press, September 2018.

[45] T.J. Sullivan. *Introduction to Uncertainty Quantification*, volume 63 of *Texts in Applied Mathematics*. Springer International Publishing, Cham, 2015.

[46] Sean Talts, Michael Betancourt, Daniel Simpson, Aki Vehtari, and Andrew Gelman. Validating Bayesian Inference Algorithms with Simulation-Based Calibration, October 2020.

[47] Alvaro Tejero-Cantero, Jan Boelts, Michael Deistler, Jan-Matthis Lueckmann, Conor Durkan, Pedro J. Gonçalves, David S. Greenberg, and Jakob H. Macke. Sbi: A toolkit for simulation-based inference. *Journal of Open Source Software*, 5(52):2505, August 2020.

[48] Daniel Ward, Patrick Cannon, Mark Beaumont, Matteo Fasiolo, and Sebastian M. Schmon. Robust neural posterior estimation and statistical model criticism. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, pages 33845–33859, Red Hook, NY, USA, November 2022. Curran Associates Inc.

[49] Xu Wu, Tomasz Kozlowski, Hadi Meidani, and Koroush Shirvan. Inverse uncertainty quantification using the modular Bayesian approach based on Gaussian process, Part 1: Theory. *Nuclear Engineering and Design*, 335:339–355, August 2018.

[50] Xu Wu, Tomasz Kozlowski, Hadi Meidani, and Koroush Shirvan. Inverse uncertainty quantification using the modular Bayesian approach based on Gaussian Process, Part 2: Application to TRACE. *Nuclear Engineering and Design*, 335:417–431, August 2018.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

POLYUQ: A SOFTWARE TOOL FOR THE SIMULTANEOUS CONSIDERATION OF ALEATORY AND EPISTEMIC UNCERTAINTIES IN ENGINEERING PROBLEMS

Martin Drieschner[1]

[1]Chair of Structural Analysis and Dynamics, BTU Cottbus-Senftenberg
Konrad-Wachsmann-Allee 2, 03046 Cottbus, Germany
e-mail: martin.drieschner@b-tu.de

Abstract. *The adequate consideration of uncertain parameters in engineering applications is essential for a realistic prediction of state variables, such as displacements or stresses. Uncertainty is omnipresent and polymorphic. The source of the uncertainty and the quality of available information determine whether the uncertainty is aleatory, epistemic or a combination of both.*

Aleatory uncertainty is inherent, unavoidable, and irreducible. Sufficient statistical information is available for the corresponding parameters, enabling them to be modeled and quantified using stochastic variables within the framework of probability theory. Epistemic uncertainty, on the other hand, is avoidable, as it can generally be reduced by better models or additional data. Various uncertainty quantification approaches exist, such as Bayesian inference, random sets, p-boxes, or representations using non-stochastic variables based on possibility theory.

We present a software tool that allows the simultaneous definition of stochastic variables, interval variables, fuzzy variables, and mixed forms as input variables. Different input dependencies can also be considered before samples are automatically generated based on user-defined solution methods. For a given computational model, the desired output variables are then determined and graphically displayed using different evaluation methods in each uncertainty space. The influence of the uncertainty description on the results will be investigated on an academic example first. Then, results from studies on various engineering problems will be presented, followed by an outlook on further development steps.

Keywords: Polymorphic Uncertainty Modeling, Aleatory Uncertainty, Epistemic Uncertainty, Fuzzy-Stochastic Variables, MATLAB.

1 INTRODUCTION

The numerical investigation of real-world problems always requires modeling. Good numerical models are as accurate as possible, but as complex as necessary, so an optimal balance between the opposing aspects of efficiency and accuracy should be found. Based on models that are as realistic as possible, reliable predictions can then be made. The efficiency and accuracy of these predictions depend, on the one hand, on the underlying structural model and, on the other hand, on the input variables to be considered. The model is always an approximation of reality, based on simplifications and subjective decisions in the modeling process, which result in unavoidable model uncertainties. Some input variables cannot be represented in the model or are deliberately neglected, leading to additional model uncertainty. Other input variables are integrated (simplified) taking into account their existing data uncertainty [10].

The influence of uncertainty has been studied for decades in various fields of application, such as weather and climate forecasting [20], geosciences [13], and nuclear energy [2]. In structural engineering, uncertainty quantification is mainly related to the design and optimization problems [3, 14], geotechnics [15] or system and parameter identification [21]. As part of the SPP 1886, funded by the DFG, several scientists across Germany have worked on various engineering questions regarding polymorphic ($\pi o \lambda \acute{\upsilon} \mu o \rho \varphi o \varsigma$ (Greek) = multifaceted) uncertainty modeling in recent years [5, 19]. The foundation is that uncertainty is omnipresent and diverse. A classification into aleatory (*alea* (Latin) = chance, risk, randomness) and epistemic ($\epsilon \pi \iota \sigma \tau \acute{\eta} \mu \eta$ (Greek) = science, knowledge, insight) uncertainty is common, depending on the existing source of uncertainty, as shown in Fig. 1.

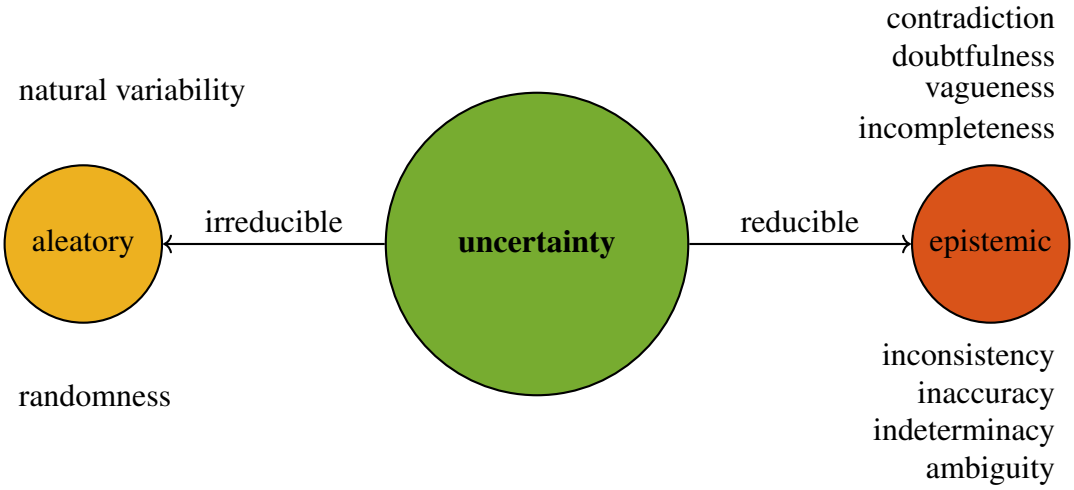


Figure 1: Uncertainty types and sources.

Aleatory uncertainty is an inherent uncertainty, as it is based on randomness and natural variability. Objectively speaking, aleatory uncertainty is therefore unavoidable and irreducible. Its treatment using probabilistic methods based on probability theory is scientifically established, as substantial statistical information, such as in the form of a histogram, is available [17, 18]. The experience with this modeling approach is already reflected in design codes of practice [4]. The probability density function (pdf) $f_X\left(x, \boldsymbol{\lambda}\right)$ and the cumulative distribution function (cdf)

$F_X\left(x, \boldsymbol{\lambda}\right)$ with shape parameters $\boldsymbol{\lambda}$ are connected through the indefinite integral:

$$P(X \leqslant x) = F_X\left(x, \boldsymbol{\lambda}\right) = \int_{-\infty}^{x} f_X\left(t, \boldsymbol{\lambda}\right) \, \mathrm{d}t \tag{1}$$

As an example of many possible distributions, consider the normal distribution $x \sim \mathcal{N}\left(\mu, \sigma^2\right)$ in Fig. 2 with $f_X\left(x\right) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$ as pdf and $\mu = \mathrm{E}(X)$ and $\sigma^2 = \mathrm{E}\left(\left(X-\mu\right)^2\right)$ as shape parameters.

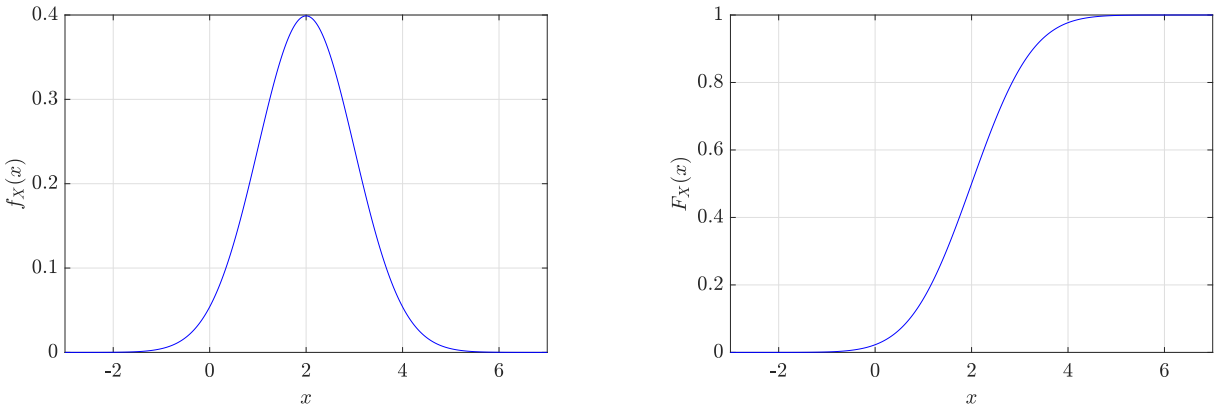


Figure 2: Modeling of aleatory uncertainty using probabilistic methods, e. g. the normal distribution $x \sim \mathcal{N}\left(2, 1\right)$. Left: Probability density function (pdf). Right: Cumulative distribution function (cdf).

Epistemic uncertainty is based on limited, inaccurate, or inconsistent data, which can result from expert knowledge, subjectivity, and model simplifications. As a result, epistemic uncertainty is reducible, provided more information becomes available and/or model improvements are made. Therefore, modeling within the framework of classical probability theory is no longer recommended, and alternative approaches should be chosen, such as Bayesian inference, random sets, p-boxes, or representations using non-stochastic variables based on possibility theory [1]. If only limits or linguistic expressions are available for a parameter, a set-theoretic approach based on possibility theory [11, 23] can be chosen, where the possibility of a value is quantified using a possibility distribution function, also called a membership function. The simplest form describes membership through the Boolean expressions $\{\mathrm{true}, \mathrm{false}\}$, meaning full membership with the value one or no membership with the value zero. These two events are disjoint sets, and operations can be performed based on classical set theory. If limits are provided for an epistemically uncertain value, quantification using an interval variable $\bar{X} = \left[x_\mathrm{l}, x_\mathrm{r}\right] = \left\{\bar{x} \in \mathbb{R} \middle| \, x_\mathrm{l} \leqslant \bar{x} \leqslant x_\mathrm{r}\right\}$ is sensible. If an interval variable is considered too restrictive, a gradual, non-Boolean extension of classical set theory is appropriate, from which fuzzy set theory [22] developed. The membership function $\mu(\tilde{x})$ is usually normalized so that a real-valued fuzzy variable $\tilde{x}$ can be described on the fuzzy set $\tilde{X} = \left\{\left(\tilde{x}, \mu\left(\tilde{x}\right)\right) \middle| \, \tilde{x} \in \mathbb{R}, \, 0 \leqslant \mu\left(\tilde{x}\right) \leqslant 1\right\}$. In Fig. 3, an interval variable and a triangular symmetric membership function of a fuzzy number ($\mathrm{TFN}\langle x_\mathrm{l}, x_\mathrm{m}, x_\mathrm{r}\rangle$ = triangular fuzzy number) are exemplarily shown.

It is also possible for an input variable to be both aleatorically and epistemically uncertain at the same time. In this case, the combination of monomorphic uncertainty models leads to a polymorphic uncertainty model. Furthermore, polymorphic uncertainty may only become visible in the output variables when different aleatory and epistemic uncertainties have been

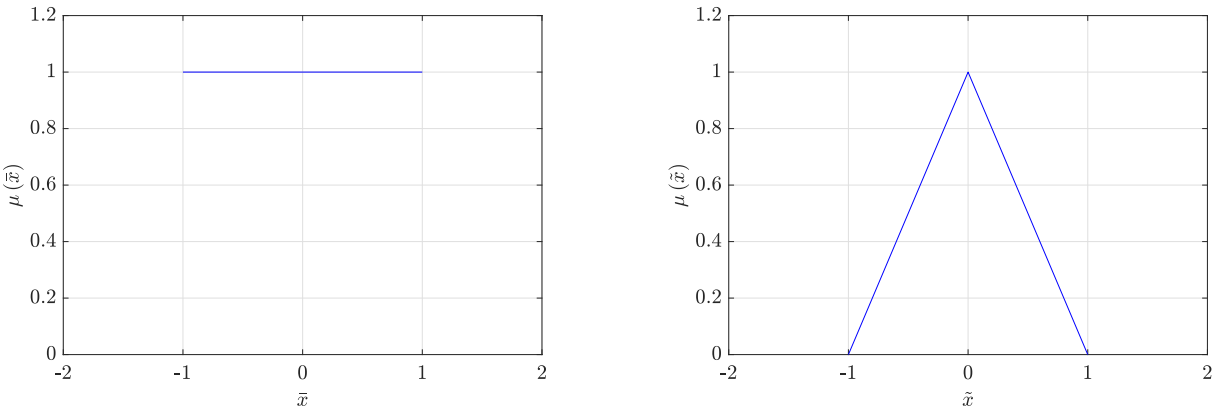


Figure 3: Modeling of epistemic uncertainty using non-probabilistic methods. Left: Interval variable $\bar{x} = [-1, 1]$. Right: Fuzzy variable $\tilde{x} = \mathrm{TFN}\langle -1, 0, 1 \rangle$.

integrated into the model. Regardless, in general, the calculation of an underlying model considering various uncertainties is necessary. This leads to the embedding of the computational model in a multi-layered uncertainty space, as shown in Fig. 4.

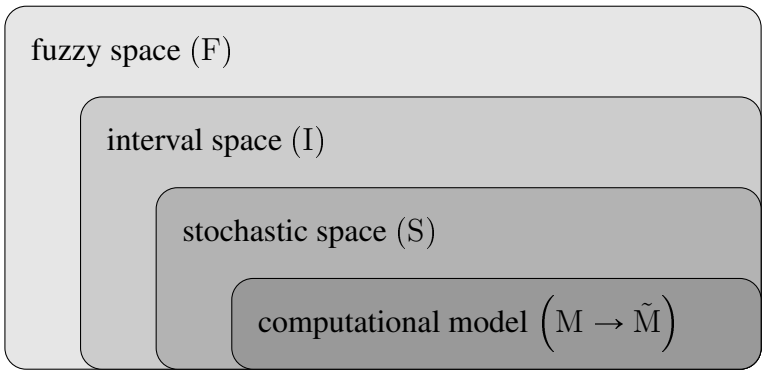


Figure 4: Scheme of a multi-layered uncertainty space with an embedded computational model.

The software tool PolyUQ, developed for the resulting nested analysis, is introduced in Section 2. An academic example is presented in Section 3. Section 4 includes various engineering applications. Section 5 finally provides the conclusions and offers an outlook on further development steps.

2 POLYUQ

The software tool PolyUQ has been developed to solve various problems under polymorphic uncertainty. Thereby, a computational model is embedded in the uncertainty space. First, the existing uncertainties are data-driven modeled and quantified. Then by choosing appropriate solution methods, efficient uncertainty propagation is possible. Finally, evaluation methods are applied to the uncertain output variables. The user will have the results in both tabular and graphical form. A general workflow is shown in Fig. 5.

The following provides a general overview of the user inputs and the functionalities. A detailed description can be found in [6]. All files can be downloaded from here.

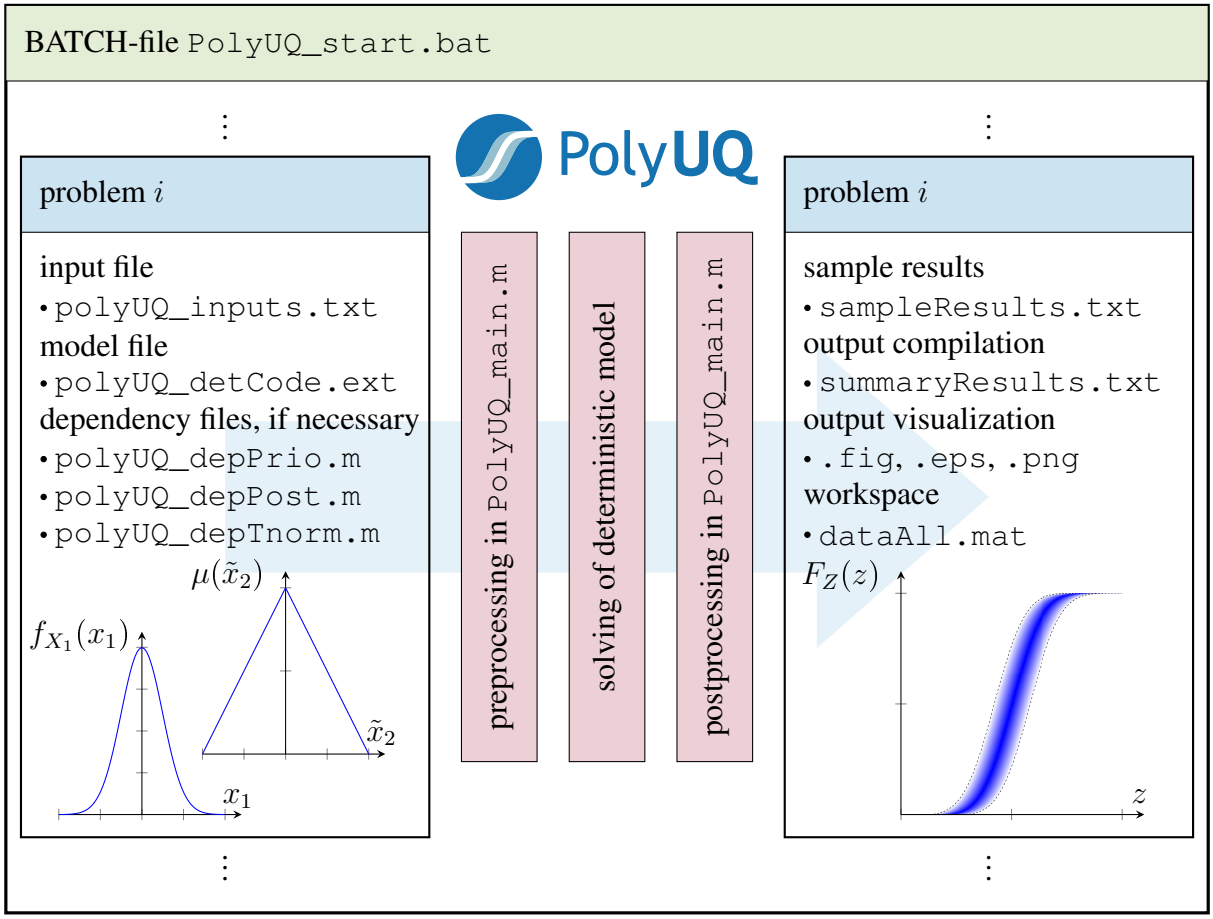


Figure 5: PolyUQ workflow [6].

2.1 User input

The user can define multiple problems, each consisting of an input file, a model file and input dependency files, if input dependencies are present. The BATCH file `PolyUQ_start.bat` controls the computation. The MATLAB main file is called `PolyUQ_main.m`. The MATLAB functions are saved in the folder `PolyUQ_functions`. All three components have to be saved in the same directory.

2.1.1 BATCH-file

The BATCH-file `PolyUQ_start.bat` controls the computation of multiple models. The user must define a model path `pathModel[i]` and a solver `solver[i]` for each model. If the computational model is not directly defined in MATLAB, additional information must be provided for the solver, such as the `exe`-file or the license. After that, the user inputs are automatically controlled. If all inputs are correctly defined, the models are serially solved. For each model, the start time and the end time is displayed in the command window for the user.

2.1.2 Input file

In the input file `polyUQ_inputs.txt` created in `pathModel[i]` for the i-th model, the definition of the uncertain input variables, the solution methods, the output variables, the output evaluation methods and the output evaluation is necessary.

Uncertain input variables Uncertain input variables can be quantified either as stochastic, interval, fuzzy, interval-stochastic or fuzzy-stochastic variables. For each variable, a unique name and a unit have to be defined.

- Stochastic variables require a distribution type, the associated distribution parameter(s) and, if desired, truncation limits.
- Interval variables are defined by a lower and an upper bound.
- Fuzzy variables require a fuzzy type and corresponding parameters.
- Interval-stochastic and fuzzy-stochastic variables are combinations of the single parts.

Solution methods The need of the definition of solution methods depends on the uncertain input variables which have been defined. It is further possible to define mutliple solution methods which are combined line by line. Thus, the influence of different solution methods on the result can be examined.

Output variables Multiple output variables can be defined. For each variable, a unique name and a unit have to be defined.

Output evaluation methods The need of the definition of output evaluation methods depends on the uncertain input variables which have been defined. It is further possible to define mutliple output evaluation methods which are combined line by line. Thus, the influence of different output evaluation methods on the result can be examined.

Output evaluation It can be specified whether the results on all evaluated points or only the evaluated result(s) in the individual space are at disposal for the user.

2.1.3 Model file

In the model file created in `pathModel[i]` for the i-th model, the user has to define the deterministic model. The model file has to be named `polyUQ_detCode` with the associated extension for the desired solver.

2.1.4 Input dependency files

Dependencies of the uncertain input variables can be defined either as a-priori dependencies (`polyUQ_depPrio.m`) or a-posteriori dependencies (`polyUQ_depPost.m`). Between fuzzy variables, also a fuzzy t-norm (`polyUQ_depTnorm.m`) is possible. No dependencies are present, if the dependency files are not defined by the user.

2.1.5 Preprocessing

After the subfolders `0_automation`, `1_preprocessing`, `2_solving` and `3_postprocessing` are automatically created in the model path `pathModel[i]` for the i-th model, the input file, the model file and, if present, the input dependency files are checked. Furthermore, the user input in the input file is read in and saved in appropriate variables. If

errors are present in the files, they are displayed in the command window to indicate necessary corrections. A formally corrected, optimal input is also displayed.

Uncertain input variables The defined uncertain input variables are displayed and saved in the subfolder `1_preprocessing`. In addition, a tabular compilation is saved in the file `summaryInputs.txt` in the same subfolder.

Solution methods The solution methods are summarized in the file `summarySoluMethods.txt` in the subfolder `2_solving`.

Output variables and evaluation methods The output variables and evaluation methods are summarized in the file `summaryResults.txt` in the subfolder `3_postprocessing`.

2.2 Solving

The solution phase is subdivided into two steps. First, all evaluation points in all non-empty uncertainty spaces are generated based on the defined input variables, input dependencies and solution methods. The procedure is conducted top-down referring to Fig. 4. Then, the deterministic model is solved on each evaluation point depending on the applied solver.

2.3 Postprocessing

The file `sampleResults.txt` is automatically created in the subfolder `3_postprocessing`. The results on all sampling points of all output variables by all solution method combinations are summarized. Based on these values, the output evaluation is conducted which depends on the defined input variables and the output evaluation methods. The results on the sampling points are evaluated bottom-up referring to Fig. 4, whereby an empty space is skipped. In addition to the text file, the output variables are displayed and saved in the same subfolder. Finally, the workspace is saved as `dataAll.mat` in the subfolder `0_automation`. A summary of the PolyUQ workflow is saved as `PolyUQ_output.txt` in the same subfolder.

3 ACADEMIC EXAMPLE

The complete PolyUQ workflow is presented in the following using an academic example. It concerns a two-dimensional frame structure in the x-z plane with a system length L and a system height H, as shown in Fig. 6. The girder is subjected to a horizontal point load λF_H and a vertical point load λF_V applied at the center. The rectangular cross-sections have a width b and a height h. The frame is built of steel S 235.

The input variables are defined partially as deterministic and partially as uncertain, using different uncertainty models, as shown in Tab. 1. The characteristic value of the yield stress is $f_\mathrm{yk} = 235\,\mathrm{N/mm^2}$, that is the $5\,\%$ quantile value of the statistical distribution. According to the Probabilistic Model Code [16], lognormal distributions are recommended to be applied with a coefficient of variation of $3\,\%$ for the Young's modulus E and with a coefficient of variation of $7\,\%$ for the yield stress f_y. The vertical point load F_V is also integrated as a stochastic variable into the model, this time as a right-sided truncated normal distribution with a coefficient of variation of $10\,\%$. The cross-sectional parameters b and h may exhibit deviations from a target value due to manufacturing tolerances. However, quality management ensures minimal and

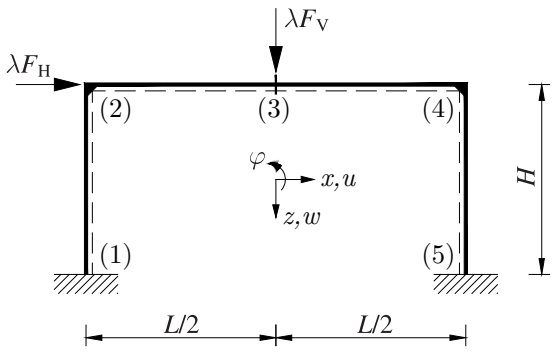


Figure 6: Frame structure.

maximal possible values $5\,\%$ around the target value. The modeling is done here using fuzzy variables with a triangular membership function.

Table 1: Input variables.

variable	unit	deterministic	stochastic	fuzzy
L	$[\mathrm{m}]$	10		
H	$[\mathrm{m}]$	5		
E	$[\mathrm{N/mm^2}]$		$\mathcal{LN}(210000, 6300)$	
f_y	$[\mathrm{N/mm^2}]$		$\mathcal{LN}(264.27, 18.50)$	
F_H	$[\mathrm{N}]$	5000		
F_V	$[\mathrm{N}]$		$\mathcal{TN}(6000, 600, -\infty, 7000)$	
b	$[\mathrm{mm}]$			$\mathrm{TFN}\langle 114, 120, 126\rangle$
h	$[\mathrm{mm}]$			$\mathrm{TFN}\langle 95, 100, 105\rangle$

The computational model M is directly generated in MATLAB. The solution methods chosen are the Monte Carlo simulation with 1000 samples in the stochastic space and the reduced transformation method with eleven equidistantly distributed α-cuts in the fuzzy space. For the output evaluation, the mean (μ) in the stochastic space and the center of gravity (COG) in the fuzzy space are calculated.

3.1 Determination of internal forces and displacements

In the first step, the maximum bending moment $M_{y,\max}\,[\mathrm{kNm}]$ and the horizontal girder displacement $u_\mathrm{G}\,[\mathrm{mm}]$ are considered as output variables for a load factor of $\lambda = 1.0$. In Fig. 7, the bending moments and the system displacements are shown as examples for the sample with mean and core values. The maximum bending moment $M_{y,\max}^\mathrm{det} = 10.81\,\mathrm{kNm}$ is located here at the support of the right column. The horizontal beam displacement is $u_\mathrm{G}^\mathrm{det} = 21.7\,\mathrm{mm}$.

Due to the existing uncertainties, the use of PolyUQ results in an uncertain output, as shown in Fig. 8.

It is evident that the maximum bending moment does not exhibit epistemic uncertainty, and is thus independent of the fuzzy input parameters b and h. The uncertainty regarding the stochastic input parameters (here, only with respect to F_V) is present. Values between $10.0\,\mathrm{kNm}$ and $11.4\,\mathrm{kNm}$ are probable. Due to the skewed pdf of F_V, the mean $M_{y,\max}^\mu = 10.78\,\mathrm{kNm}$ is

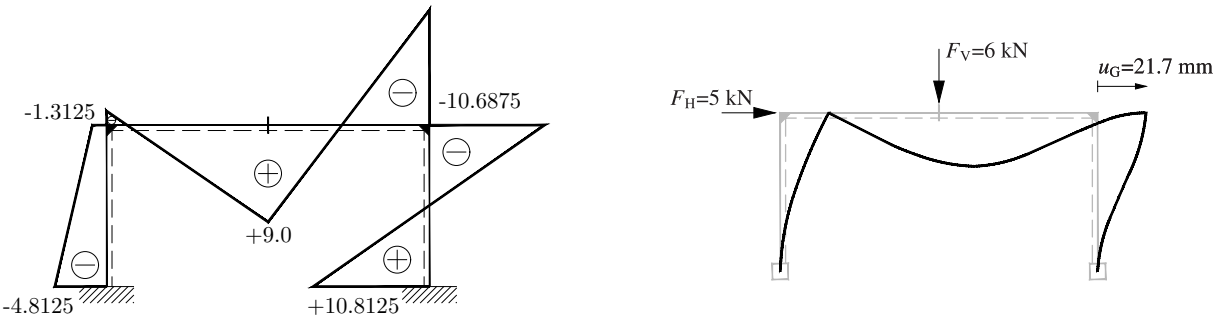


Figure 7: Deterministic results with mean and core values. Left: Bending moments. Right: Displacements.

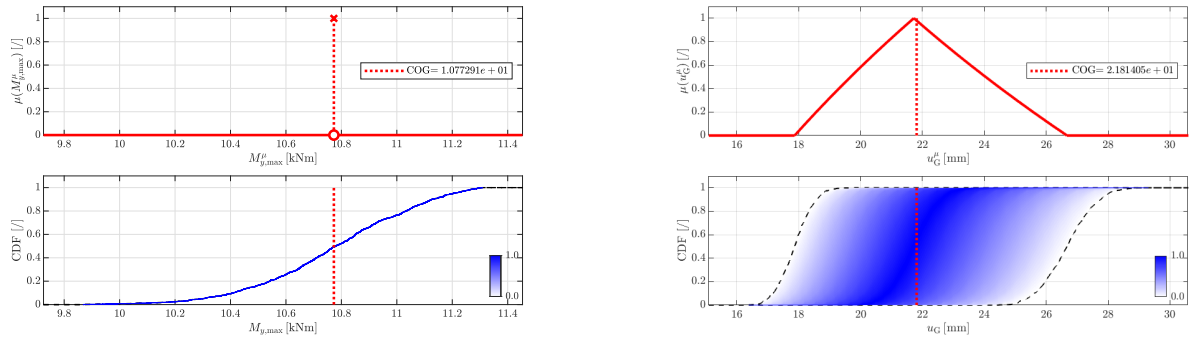


Figure 8: Uncertain outputs caused by present aleatory and epistemic uncertainties. Left: Maximum bending moment $M_{y,\max}$ [kNm]. Right: Horizontal girder displacement u_G [mm].

slightly smaller than the deterministic value $M_{y,\max}^{\det} = 10.81\,\mathrm{kNm}$. In the horizontal girder displacement, both the influences of the aleatory and epistemic uncertain parameters are visible. On the support ($\mu = 0$), values between $16.0\,\mathrm{mm}$ and $28.0\,\mathrm{mm}$ are probable, with the mean $u_G^{\mathrm{supp}(\mu)} = [17.8, 26.6]\,\mathrm{mm}$. On the core ($\mu = 0$), values between $20.0\,\mathrm{mm}$ and $24.0\,\mathrm{mm}$ are probable, with a mean of $u_G^{\mathrm{core}(\mu)} = 21.7\,\mathrm{mm}$. The defuzzified value $u_G^{\mathrm{COG}(\mu)}$ is also $21.7\,\mathrm{mm}$. Based on the evaluation of the uncertain results, decisions can be made in the next step, taking the uncertainty into account.

3.2 Determination of the load-deformation curve

The problem is extended to account for physically nonlinear effects. The load-deformation curve is sought, which is derived using cross-section and system reserves with the first-order yield line theory. Again, the uncertainties in the input variables are propagated through the model and influence the output. In Fig. 9, the final solution with mean and core values is illustrated on the left, having an ultimate load factor of $\lambda_{\max} = 8.65$. With the results automatically saved in `sampleResults.txt`, the load-displacement diagram on the right could be created. The mean values are shown in blue and the $5\,\%$ and $95\,\%$ fractile values on the support with dashed lines.

The deterministic solution is enveloped by the solution that takes the uncertainty in the input parameters into account. It is clearly evident that there is a strong dependence of the curve on the uncertain input parameters, both in the u_G- and λ-direction. As an example, the ultimate load factor $\lambda_{\max}$ is analyzed. The mean on the support range between 7.48 and 10.10, with the $5\,\%$ and $95\,\%$ fractile values even ranging from 6.53 to 11.60. The mean on the core is quite

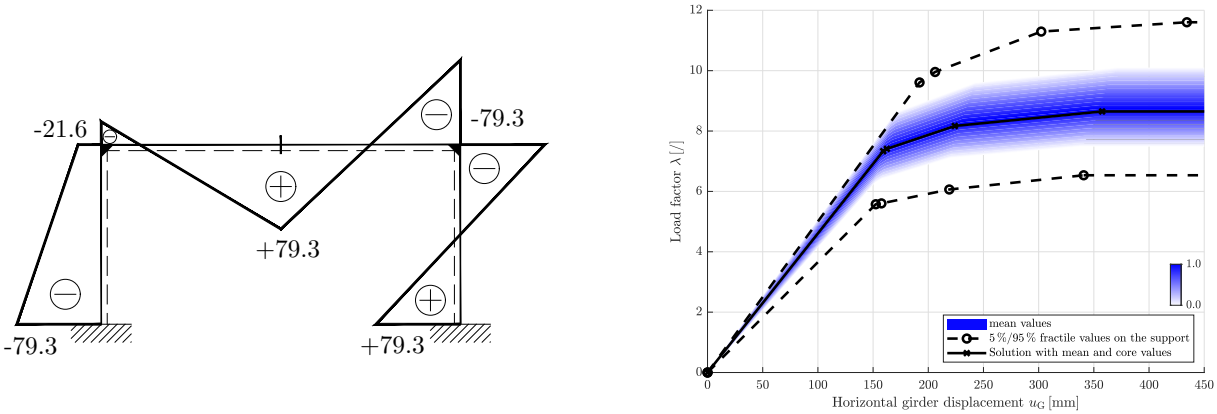


Figure 9: Consideration of physically nonlinear effects. Left: Final bending moments $M_y(\lambda_{\max})$ [kNm] with mean and core values. Right: Uncertain load-displacement diagram.

similar to the defuzzified value of 8.74. It is worth mentioning that qualitative differences in system behavior may also occur which have to be analyzed in further studies.

4 ENGINEERING APPLICATIONS

The computational model (M) is fully interchangeable, and this does not affect the definition of the (multi-layered) uncertainty space. Thus, the application to various engineering problems and beyond is possible, as shown in the following sections.

4.1 Surrogate modeling

In [8], a structure made of carbon fiber-reinforced plastic (CFRP) with an Ω-shaped cross-section was subjected to a three-point bending test until nonlinear stability failure. The focus was on the tilted webs, which have a reduced thickness compared to the flanges, and therefore, lower load-bearing capacity. Relevant uncertain input variables were modeled using both stochastic and non-stochastic approaches and integrated into the model. The underlying computational model (M) is a finite element model, which leads to a high computational effort in the context of a nonlinear analysis. Therefore, a surrogate model ($\tilde{\mathrm{M}}$) based on artificial neural networks (ANN) was used. The numerical predictions of local strains and the entire displacement field were compared with experimentally obtained data, as shown in Fig. 10.

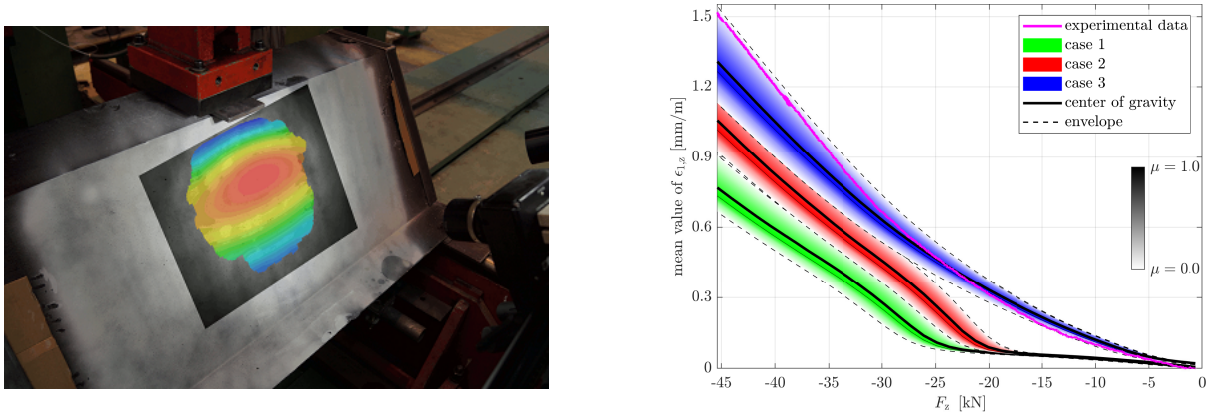


Figure 10: Numerical and experimental results using a surrogate model ($\tilde{\mathrm{M}}$). Left: Displacement field $\boldsymbol{u}_{\mathrm{norm}}(\boldsymbol{x})$. Right: Local horizontal strain $\epsilon_{1,z}$.

4.2 Multicriteria optimization

In [7], a beam made of glass fiber-reinforced plastic (GFRP) with bonds made of epoxy resin (EP) has been investigated in a four-point bending test. The structure was optimized in terms of cost minimization and compliance minimization, for which an embedding of the PolyUQ framework into an optimization routine was carried out [12]. Different topologies regarding the number of bonding layers were examined, and constraints concerning structural failure and maximum deformations were defined. Uncertain cross-sectional and material parameters were integrated into the model. The Pareto optimal points for a topology with four bonds are exemplarily shown in Fig. 11.

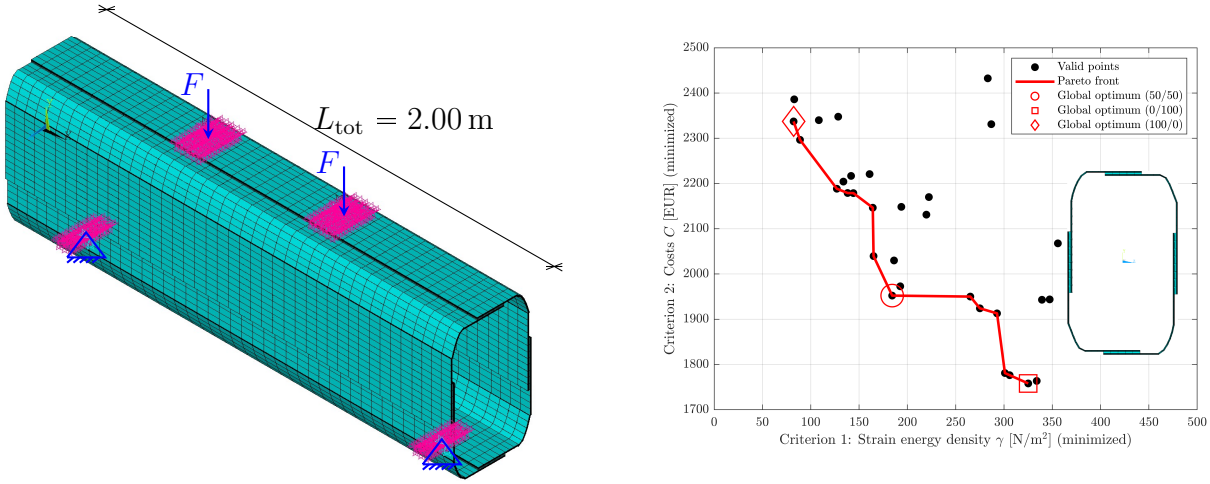


Figure 11: MuScaBlaDes beam. Left: Isometric view with boundary conditions. Right: Pareto optimal points after final optimization step of beam with four bonds.

4.3 Data assimilation

In [9], the focus was on considering numerical predictions and real measurement data of system responses, as well as the simultaneous estimation of unknown model parameters under the presence of aleatory and epistemic uncertainty. The data assimilation approach was used, with the EnKF (ensemble Kalman filter) being employed, which, in its extended form, can simultaneously estimate state variables $\boldsymbol{x}$ and unknown model parameters $\boldsymbol{b}$. For this purpose, the EnKF was directly embedded into the stochastic space generated within the PolyUQ framework. The real test structure was a stainless steel beam with a hollow box cross-section, on which vertical accelerations were experimentally determined during oscillation tests using five accelerometers, as shown in Fig. 12.

5 CONCLUSIONS AND OUTLOOK

PolyUQ is a software tool that was developed in-house and is made available for download to everyone. A detailed documentation can be found in [6]. It is used to solve problems involving aleatory and epistemic uncertainty, which are modeled using stochastic and non-stochastic variables. Although various application examples from engineering have been presented in this paper, extensions to the current version are necessary, particularly with regard to input options, user-friendliness, solution methods, and interfaces to external solvers. For any questions, further ideas or interest in cooperation, please contact the corresponding author via e-mail.

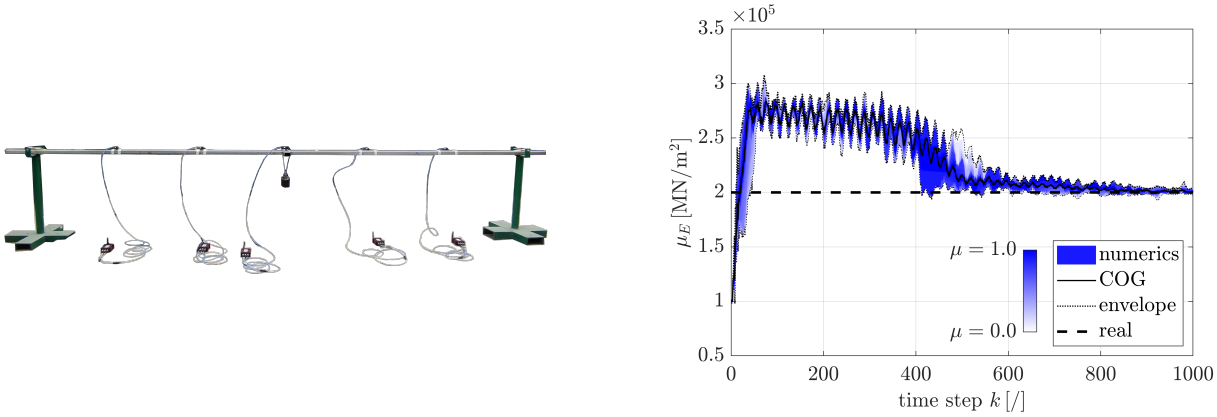


Figure 12: Application of the EnKF. Left: Real structure with five accelerometers. Right: Prediction of the mean of Young's modulus $\mu_E(t)$.

REFERENCES

[1] M. Beer, S. Ferson, V. Kreinovich, Imprecise probabilities in engineering analyses. *Mechanical Systems and Signal Processing*, **37**, 4–29, 2013.

[2] F. Bostelmann, G. Strydom, Nuclear data uncertainty and sensitivity analysis of the VHTRC benchmark using SCALE. *Annals of Nuclear Energy*, **110**, 317–329, 2017.

[3] A. Csébfalvi, Structural optimization under uncertainty in loading directions: Benchmark results. *Advances in Engineering Software*, **120**, 68–78, 2018.

[4] DIN Deutsches Institut für Normung e.V., *DIN EN 1990, Eurocode 0: Grundlagen der Tragwerksplanung*. Beuth Verlag GmbH, 2010.

[5] M. Drieschner, H. G. Matthies, T.-V. Hoang et al., Analysis of polymorphic data uncertainties in engineering applications. *GAMM-Mitteilungen*, **42**, e201900010, 2019.

[6] M. Drieschner, L. Eichner, F. Seiffarth et al., PolyUQ: framework for solving problems with polymorphic uncertainties. *Preprint-Reihe des Fachgebiets Statik und Dynamik, Technische Universität Berlin*, **2021**, 01, 2021.

[7] M. Drieschner, Y. Petryna, L. Eichner, Design of a 3D composite structure considering data uncertainties and various failure mechanisms. *PAMM*, **21**, e202100030, 2021.

[8] M. Drieschner, C. Wolf, F. Seiffarth et al., Combined ANN-FEM approach for spatial–temporal structural response prediction: Method and experimental validation. *Thin-Walled Structures*, **189**, 110800, 2023.

[9] M. Drieschner, C. Herrmann, Y. Petryna, The Data Assimilation Approach in a Multilayered Uncertainty Space. *Modelling*, **4**, 529–547, 2023.

[10] M. Drieschner, *Effiziente experimentell gestützte Modellierung unter polymorpher Unschärfe: Herausforderungen von der Erstellung bis zur Bewertung*. Technische Universität Berlin (Dissertation), 2024.

[11] D. Dubois, H. Prade, Possibility Theory and Its Applications: Where Do We Stand? J. Kacprzyk, W. Pedrycz eds. *Springer Handbook of Computational Intelligence*, 31–60, 2015.

[12] L. Eichner, *Mehrzweck-Optimierung einer dreidimensionalen Kompositstruktur unter Berücksichtigung der Parameterunschärfe*. Technische Universität Berlin (master thesis), 2021.

[13] S. J. Fletcher, *Data assimilation for the geosciences: from theory to application*. Elsevier, 2017.

[14] M. Guedri, S. Cogan, N. Bouhaddi, Robustness of structural reliability analyses to epistemic uncertainties. *Mechanical Systems and Signal Processing*, **28**, 458–469, 2012.

[15] M. Huber, Reducing forecast uncertainty by using observations in geotechnical engineering. *Probabilistic Engineering Mechanics*, **45**, 212–219, 2016.

[16] JCCS: Joint Committee on Structural Safety, *Probabilistic Model Code, Part 3, Resistance Models. 3.02: Structural steel*. Technical University of Denmark, 2001.

[17] A. D. Kiureghian, O. Ditlevsen, Aleatory or epistemic? Does it matter? *Structural Safety*, **31**, 105–112, 2009.

[18] W. L. Oberkampf, J. C. Helton, C. A. Joslyn, S. F. Wojtkiewicz, S. Ferson, Challenge problems: uncertainty in system response given uncertain parameters. *Reliability Engineering and System Safety*, **85**, 11–19, 2004.

[19] I. Papaioannou, M. Daub, M. Drieschner et al., Assessment and design of an engineering structure with polymorphic uncertainty quantification. *GAMM-Mitteilungen*, **42**, e201900009, 2019.

[20] J. Slingo, T. Palmer, Uncertainty in weather and climate prediction. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **369**, 4751–4767, 2011.

[21] K. Worden, R.J. Barthorpe, E.J. Cross et al., On evolutionary system identification with applications to nonlinear benchmarks. *Mechanical Systems and Signal Processing*, **112**, 194–232, 2018.

[22] L. A. Zadeh, Fuzzy sets. *Information and Control*, **8**, 338–353, 1965.

[23] L. A. Zadeh, Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, **100**, 9–34, 1999.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

MULTI-DISCIPLINARY DESIGN OPTIMIZATION UNDER UNCERTAINTY: THE OPEN SOURCE CAPABILITIES OF GEMSEO

Matthias De Lozzo[1], Clément Laboulfie[1], Olivier Sapin[1], Nicolas Roussouly[1], François Gallard[1], Amine Aziz-Alaoui[1], Antoine Dechaume[1], and Anne Gazaix[1]

[1]IRT Saint Exupéry
Toulouse, France, 31400
e-mail: matthias.delozzo@irt-saintexupery.com

Abstract. *The increasing complexity of systems involved in engineering applications requires a breakthrough in the management of coupled models, also called disciplines, to be able to design new products in line with nowadays constraints. This design can be supported by multi-disciplinary optimization (MDO) methods, in order to handle optimization and coupling problems together by means of MDO formulations, a.k.a. MDO architectures. The Python library GEMSEO automates the MDO process generation from the MDO problem specification and a given MDO formulation. GEMSEO can also be used for surrogate modeling, to visualize optimization profiles, to manage data as the process progresses, etc. Over the last years, the GEMSEO team put a major effort to add uncertainty quantification (UQ) techniques into GEMSEO, and at the core of this, the extension of the MDO formulations to uncertainty. A unified way to extend existing deterministic MDO formulations (MDF, IDF, multi-level ones) has been proposed. Besides, the whole MDO suite has been extended to account for uncertain effects: defining and visualizing uncertainties, propagating them through multi-disciplinary systems, and performing statistical analysis of the quantities of interest, such as sensitivity analysis. In the last years, GEMSEO plugins have also been enriched to allow batch acquisition and quantile estimation with active learning methods, and to solve MDO problems under uncertainty, in order to promote the use of UQ techniques, not only in academia but also in industry. In this work, we present how GEMSEO and its plugins can be useful to carry out multi-disciplinary studies with uncertainties. We first give a brief overview of GEMSEO. Then, we demonstrate some capabilities for propagating uncertainties through weakly and strongly coupled disciplines, and solving MDO problems under uncertainty, with a focus on the use of surrogate-based variance reduction methods. In this work, we also show how GEMSEO can be enriched with new algorithms and software libraries, through intensive use of the object-oriented paradigm. Finally, we give some ideas for the development of UQ capabilities in GEMSEO in the coming years.*

Keywords: Multi-Disciplinary Optimization, Uncertainty Quantification, Robust Optimization, Sensitivity Analysis, Surrogate Models, Open Source Software.

1 INTRODUCTION

Optimization problems are common to design the product or the strategy that minimizes a cost (or maximizes a performance) while respecting constraints. An optimization problem can be written in a standard way as

$$\begin{array}{lll} \min\limits_{x} & f(x,u) & \\ \text{s.t.} & g(x,u) & \leq \ 0 \\ & h(x,u) & = \ 0 \\ & x \in \mathcal{X} & \end{array} \tag{1}$$

where x groups the optimization variables, f is the objective function (either the cost or the inverse of the performance), g are inequality constrained functions, h are equality constrained functions, and u groups various parameters, typically, in the following, uncertain variables.

1.1 Multi-disciplinary design optimization

In the context of multi-disciplinary design optimization (MDO), the optimization problems are generated from processes, which are composed of several models called disciplines. This specificity comes from the fact that disciplines are linked by coupling variables y, some weakly (or one-way coupling), others strongly (or two-way coupling, i.e., more rigorously, when the disciplines are involved in cycles of the input/output dependency graph). The handling of these couplings, combined with optimization, is addressed by the so-called MDO formulations or MDO architectures [14]. An MDO architecture is the combination of an optimization problem and a computational process that computes the objective function and constraints for given values of the optimization variables. There are several MDO formulations, which reformulate the original design problem, called the SAND formulation, into computationally tractable ones. The most widely used is MDF (multi-disciplinary feasible), which formulates an optimization problem that is very similar to Problem (1). The coupling variables do not appear in this problem because the process associated with MDF is the multi-disciplinary analysis (MDA), that computes the coupling variables at each iteration of the optimization. They are implicit functions of the design variables. Another very common formulation, because of its simplicity, is the individual design feasible (IDF) formulation, which executes all disciplines in parallel at each iteration, where the coupling consistency being ensured, at convergence, by specific equality constraints added to the optimization problem, and coupling variable targets y_t being added to the optimization variables. The optimization problem associated with IDF is also similar to Problem (1), when x is replaced by x, y_t and consistency constraints $c(x, y_t, u) = y(x, y_t, u) - y_t = 0$ are added to the problem, in addition to the user equality constraints $h(x, u) = 0$.

Some MDO formulations involve multiple optimization problems, and are then called distributed, or multi-level when the problems are nested such as CSSO and BLISS, each with its own advantages and disadvantages depending on the use case [29]. More recently, [9] proposed a bi-level formulation that is particularly suited to studies already equipped with disciplinary optimizers, e.g. structure sizing, or to studies where it would be a good idea to separate the optimization problem by type of variables, e.g. continuous vs. discrete variables. These formulations also generate optimization problems that are similar to Problem (1), since they are derived from MDF.

1.2 The open source Python library GEMSEO

In the face of this, the open source Python library GEMSEO allows the user to set up the optimization problem (1) from a collection of disciplines and a catalog of MDO formulations, including bi-level ones [9], without having to connect the disciplines by hand. It also enables the process to be automatically reconfigured when any element of the problem is changed, e.g. the design space, a discipline or the MDO formulation. In addition to the MDO formulations, optimization and coupling algorithms mentioned above, GEMSEO can be used to create surrogate models, to visualize optimization profiles, to manage data as the process progresses, etc. Any type of model can be connected to GEMSEO, through specific wrappers of Python functions, external executables, Excel spreadsheets, Matlab scripts or others. It also offers several features for parallel computation and remote execution via SSH, HTTPS, or HPC job schedulers.

GEMSEO was initially created in 2015 within the MDO competence center of IRT Saint Exupéry. This center is dedicated to the development of process automation technologies encompassing a wide range of disciplines, and their usage in a range of applications. Originally known as GEMS (Generic Engine for MDO Scenarios), this Python package became GEMSEO in 2021 when IRT Saint Exupéry and its partners decided to release it as an open source project with the aim to make it widely collaborative. GEMSEO is the acronym for generic engine for multi-disciplinary scenarios, exploration and optimization. It is the result of successive projects carried by IRT Saint Exupéry, funded by both Investments for the Future Programme (french acronym: PIA) and companies and involving both academic and industrial partners. It has been enriched by contributions from European projects. The GEMSEO team organized the first annual User Day on November 28, 2024 to bring the community together and share experiences.

GEMSEO is open source under the GNU LGPL v3.0 license for the source code, BSD 0-Clause for the examples and CB BY-SA 4.0 license for the documentation. It gives users the right to to use the software, study and analyze its source code, to modify the software and thus its source code, to distribute the original software or its derivatives. The project is hosted on gitlab (`www.gitlab.com/gemseo/dev/gemseo`), the main documentation can be accessed at `www.gemseo.org` and the packages are available on pypi. To improve the quality of GEMSEO and release it faster, the core development team uses the agile-based practices: short incremental cycles, daily meeting, conception review before coding, review before integrating changes and software development forge. The source code comes with automated and mandatory testing (continuous integration), reproducible and isolated testing (automation project *tox*, Python environments, controlled dependencies), on all supported Python versions, on major target OS (Linux, Windows) and with high code coverage ($>$90%). All the tests must always pass on the reference source code (develop branch). In terms of code quality, automatic verifications are made for every code changes (pre-commit hooks) and refactoring is done continuously. Lastly, documenting is done continuously, close to the source code and reviewed like the code.

1.3 GEMSEO and UQ

In the recent years, accounting for uncertainties in optimization problems has become a major research topic, including MDO problems [6]. Typically, physical constants usually assumed to be known are in fact uncertain, the optimum design (exact or found numerically) is not exactly the one used in practice, the initial conditions of the disciplines are poorly known, etc. And all these sources of uncertainty, whether they come from one or several disciplines, actually impact a large number of disciplines, since they are coupled. This leads to algorithmic

problems in easily propagating uncertainties, especially in the presence of strong couplings. For that reason, the GEMSEO team has developed uncertainty quantification (UQ) capabilities, using the framework of probability theory, and has completed its package `gemseo` with `gemseo-umdo` (documentation) and `gemseo-mlearning` (documentation), two specialized plugins (by GEMSEO, we mean `gemseo` and its plugins). These packages are based on specialized open source Python libraries, such as OpenTURNS [3] for UQ, SMT [5] and scikit-learn [17] for surrogate modelling and NumPy and SciPy for applied mathematics.

Figure 1 illustrates the main steps of an MDO study under uncertainty. The aim of GEMSEO is to carry out these steps with a unique software environment, to standardize the user experience, avoid the need to master different software libraries and facilitate the transition from MDO to UQ, which are two potentially complex fields of applied mathematics in engineering departments. First (Step A), we define the MDO problem under uncertainty: which input uncertainties, which statistical objective to optimize, which statistical constraints to satisfy, which design variables to control, which disciplines to compute the f, g and h from x and U and which MDO formulation to generate the execution process. Then (Step B), we need to define the probability distributions characterizing the uncertain input variables. In a third stage (Step C), we have to propagate these uncertainties through the multi-disciplinary process, using sampling typically. Possibly (Step C'), the contribution of the uncertain inputs to the quantities of interest is studied, variables are sorted and non-influences are set to nominal values. Lastly (Step D), Steps C and potentially C' are repeated in an optimization loop until a stopping criterion is met (hoping for optimum value). Note that Figure 1 is adapted from one classically used in the UQ literature [20], to which we have added MDO features (disciplines, MDO formulation, objective, constraints and design variables) and Step D.

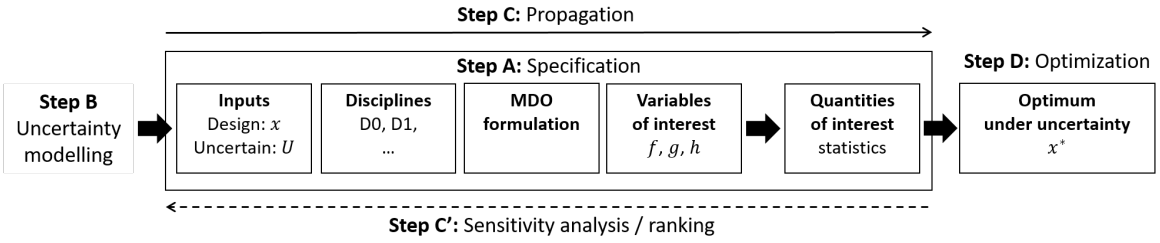


Figure 1: Breakdown of an MDO study under uncertainty (adapted from [20]).

1.4 Outline

In what follows, Section 2 presents the main concepts of GEMSEO on which the UQ elements have been built. Then, Section 3 shows how to define and propagate uncertainties through a multi-disciplinary system, and study the sensitivity of quantities of interest to sources of uncertainty. Section 4 focuses on the definition and resolution of an MDO problem under uncertainty, with different kinds of statistic estimators, illustrates these capabilities on an academic use case and proposes benchmark problems to compare U-MDO techniques. Finally, Section 5 recalls the main points and gives a vision of the GEMSEO roadmap concerning UQ and U-MDO.

2 BASIC CONCEPTS OF GEMSEO

This section provides a brief overwiev of the main Python classes of GEMSEO on which the UQ elements rely. It concludes with an UML class diagram summarizing these elements.

In this paper, the expressions *NumPy array* and *float number* will be replaced by *array* and *float* for the sake of brevity. In other respects, bear in mind that about everything in GEMSEO can be enriched by deriving a Python class, either directly in `gemseo` or in its existing plugins or in a new public or private plugin. Your new class will then be automatically recognized by GEMSEO, via a factory mechanism. As an example, if a `Color` class has an associated `ColorFactory`, then this factory can create an instance of any class derived from `Foo` without knowing its import path. This child class of `Color` can be defined either in `gemseo`, in a public plugin or in a module of your Python path if you want to keep your development private. So, in practice, a `Square` class equipped with a `ColorFactory` can generate `Square` objects from any color class derived from `Color`. You only have to know the class name, and instantiate the class as `blue_square = Square("BlueSquare")`.

MDOFunction The most basic concept of GEMSEO is the `MDOFunction`, which wraps a callable object, e.g. a Python function, taking an array as argument and returning either an array or a float. It can also wrap the function computing its Jacobian matrix expressed as an array from an array. An `MDOFunction` has methods to compute the output value (array or float), and the Jacobian matrix (array), from the input value (array). It is equipped with algebraic operators, namely addition, subtraction, multiplication, division and negation, in order to rewrite the functions of the optimization problem in a standardized way: the objective functions as cost functions, the inequality constraint functions as negativity constraints, and the constraint functions with a zero threshold.

DesignSpace A `DesignSpace` is a dictionary-like object to define scalar and vector variables from a name, a lower bound, an upper bound, a type (integer or real) and a current value. A point in the design space can be represented either as a dictionary {`variable_name: variable_value, ...`}, where `variable_value` is an array, or as a full array concatenating the sub-arrays. It offers easy-to-use methods for switching from one representation to another, as well as others to express the point according to a particular transformation, such as normalization into the unit hypercube from bounds.

EvaluationProblem An `EvaluationProblem` is set of `MDOFunctions`, a `DesignSpace` and a method to evaluate these `MDOFunctions` using a mechanism to check that the evaluation point respects the bounds of the `DesignSpace`. It also proposes a method for replacing the original `MDOFunctions` by specific `MDOFunctions`, called `ProblemFunctions`, that can count their evaluations, transform the evaluation point according to the `DesignSpace`, store their evaluations in a `Database` to avoid evaluating an `MDOFunction` twice at a same evaluation point. This `Database` can be converted into a `Dataset`, which is a multi-index pandas DataFrame [28] easier to handle.

OptimizationProblem An `OptimizationProblem` is an `EvaluationProblem` whose `MDOFunctions` are labelled as *objective*, *constraints* and *observables*.

DriverLibrary `BaseDriverLibrary` is the base class to interface a library of algorithms, called drivers. Once instantiated from an algorithm name, its `execute` method uses this algorithm to evaluate the `MDOFunctions` of an `EvaluationProblem` from algorithm settings. In particular, `BaseDOELibrary` is the base class to interface design of experiments (DOE)

algorithms, while `BaseOptimizationLibrary` is the base class to interface optimization algorithms and is limited to `OptimizationProblems`. GEMSEO gives access to various libraries, such as SciPy, pyDOE3 and OpenTURNS for DOEs and SciPy, NLopt and pymoo for optimizers.

Discipline A discipline is defined by a `Discipline` whose `execute` method computes labeled output data of the form {`output_name: output_value, ...`} from labeled input data of the form {`input_name: input_value, ...`}. The `Discipline` can also compute the Jacobian of the form {`output_name: {input_name: jac_value, ...}, ...`} when the discipline is differentiable, either analytically or by approximation, e.g. finite differences. The input and output variables of a discipline are defined by grammars, in terms of names, types and compulsoriness. At execution, the `Discipline` checks that the input and output data are consistent with these grammars. If the input data are incomplete, the `Discipline` completes them with default input values.

DependencyGraph A `DependencyGraph` analyzes the input and output variable names of a collection of `Disciplines` to infer how these depend on each other, by using the convention: two homonymous variables represent the same variable. It produces the sequence of execution of the disciplines, the coupling variables between two disciplines and a directed coupling graph. `DependencyGraph` can also display this graph, either in full form, with one node per discipline and one edge by coupling variable, condensed form, with one node per weakly coupled discipline or group of strongly coupled discipline. The user is led to use the concept of `Discipline` rather than that of `MDOFunction` which is dedicated to the process layer.

BaseFormulation The base class `BaseFormulation` defines the concept of MDO formulation, which is both the definition of the design space and objective and constraints functions, and the overall organization of the process. This `BaseFormulation` is composed of an `OptimizationProblem`, created from a `DesignSpace`, a set of `Disciplines`, names of discipline output variables to minimize or maximize, and names of discipline output variables to to be constrained. More precisely, the `BaseFormulation` plugs `MDOFunctions` into the `OptimizationProblem` and these `MDOFunctions` wrap the `Disciplines` according to the overall process. This process can be composed of several levels, and in some cases, sub-processes can be parallelized. `MDF`, `IDF`, `BiLevel` and `DisciplinaryOpt` are concrete classes deriving from `BaseFormulation`. `DisciplinaryOpt` defines a process to execute the disciplines in the order in which they are given, while the others implement the eponymous MDO formulations presented in Section 1.1 using a `DependencyGraph` to identify the coupling variables and potentially deduce the sequence of execution of the disciplines. In the latter case, the formulations MDF and BiLevel also include sub-processes implemented as `ProcessDiscipline` objects, such as sub-optimizations and MDAs.

MDOScenario Finally, `MDOScenario` is the *key class* for the end user. It aims to gather all elements necessary to define, analyze, solve and post-process an MDO problem: a `DesignSpace`, a set of `Disciplines`, the names of the discipline output variables to minimize or maximize, the names of the discipline output variables to be constrained, the name and settings

of an MDO formulation, and the name and settings of a driver. Here is an example to define and solve the Sellar problem [25] using the MDF formulation:

```
disciplines = [
    AnalyticDiscipline({"y1: (z1**2+z2+x-0.2*y2)**0.5"}, name="Sellar1"),
    AnalyticDiscipline({"y2": "abs(y1)+z1+z2"}, name="Sellar2"),
    AnalyticDiscipline(
        {
            "f": "x**2+z2+y1**2+exp(-y2)",
            "c1": "3.16-y1**2",
            "c2": "y2-24",
        },
        name="SellarSystem"
    )
]

design_space = DesignSpace()
design_space.add_variable("x", lower_bound=0.0, upper_bound=10.0)
design_space.add_variable("z1", lower_bound=-10.0, upper_bound=10.0)
design_space.add_variable("z2", lower_bound=0.0, upper_bound=10.0)

scenario = create_scenario(
    disciplines, "obj", design_space, formulation_name="MDF"
)
scenario.add_constraint("c1", constraint_type="ineq")
scenario.add_constraint("c2", constraint_type="ineq")
scenario.execute(algo_name="SLSQP", max_iter=10)
```

Figure 2 summarizes this entire section in the form of a class diagram where classes are represented by boxes. The abstract classes have dashed borders, the edges with diamonds represent composition relationships (the class with the diamond is made up of the class at the other end) and the edges with triangles represent inheritance relationships (the class with the triangle is the parent of the class at the other end). The edges with arrows represent a certain causal relationship (a non-pointed class uses a pointed class or generates an instance of this class).

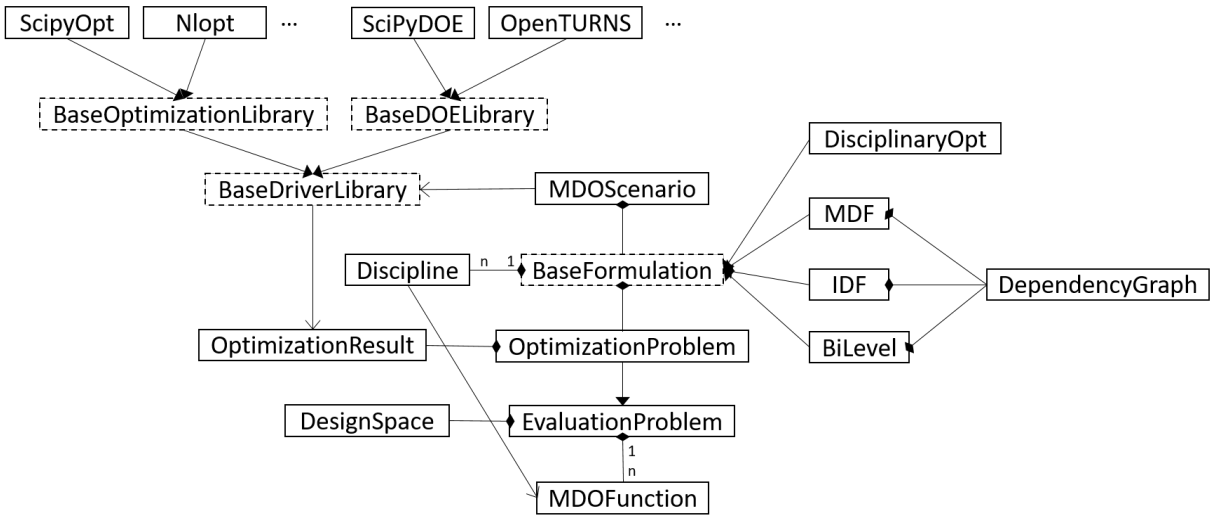


Figure 2: Class diagram of the basic concepts of GEMSEO.

3 UQ IN MULTI-DISCIPLINARY SYSTEMS

Although its main area of applied mathematics is optimization, as illustrated by Figure 2 with the `OptimizationProblem` and `OptimizationResult` classes for example, the technological core of GEMSEO is the automation of multi-disciplinary system creation, execution and reconfiguration. This is illustrated in Figure 3, where an `EvaluationProblem` is automatically created from a `DesignSpace` and a set of `Disciplines`. This makes it possible to evaluate any output of a multi-disciplinary system from the value of one of its inputs, which multiplies the possibilities of use in other mathematical fields (simple evaluation of the multi-disciplinary system, surrogate model of this system, visualization of an output of interest, sensitivity of the output of a discipline with respect to the input of another discipline, etc.). In particular, the sub-package `gemseo.uncertainty` was originally added not to optimize in presence of uncertainties, as we will see in Section 4, but to estimate statistics for structural composite models defined as modular discipline chains [13], using the MDF formulation, as illustrated in Figure 4.

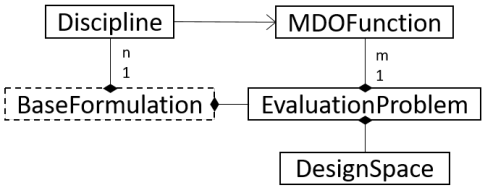


Figure 3: Minimal class diagram for defining a multi-disciplinary evaluation problem.

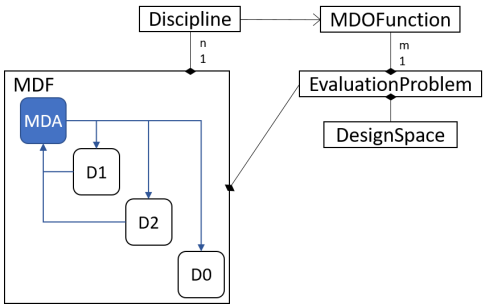


Figure 4: Minimal class diagram for defining a multi-disciplinary evaluation problem in an MDF way.

3.1 Uncertainty propagation through a multi-disciplinary system

Given a set of disciplines, an input variable space and a collection of output variables of interest, `gemseo` is able to generate a collection of `MDOFunctions` mapping from the input variable space to an output variable space and evaluate them, as explained in Section 2. This is typically what is done when the user creates an `MDOScenario` or calls the `sample_disciplines` function to sample a multi-disciplinary system over the input space when the notions of *objective* and *constraint* are not useful (in other words, we simply want to calculate "input values - output values" pairs). In the case of sampling, the input variables are usually defined by bounds and thus a `DesignSpace` is perfectly suited to gather them. When these variables are random, a `ParameterSpace` deriving from `DesignSpace` is more appropriate as it can also define scalar and vector variables from their probability distributions. `gemseo` offers an

interface to all the distributions available in OpenTURNS and SciPy, through the generic classes `OTDistribution` and `SPDistribution`, as illustrated in Figure 5. There are also specific classes for specific probability distribution, e.g. `OTNormalDistribution`. Note that the developer can interface other UQ libraries by deriving the base class `BaseDistribution`. The class will then be automatically recognized by `gemseo` and accessible via its name and arguments:

```
uncertain_space = create_parameter_space()
uncertain_space.add_random_variable("u", "OTNormalDistribution", sigma=0.8)
```

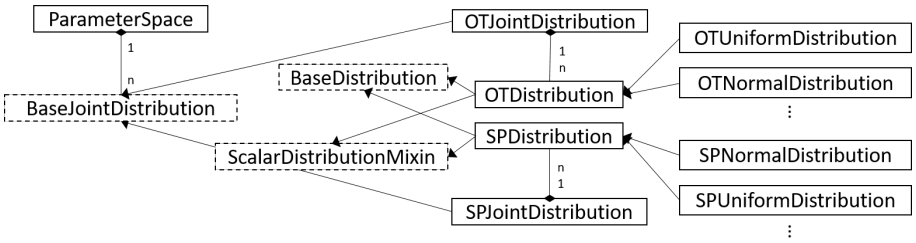


Figure 5: Class diagram for probability distributions and their use by `ParameterSpace`. OT and ST stand for OpenTURNS and SciPy respectively. `OTDistribution` can represent any probability distribution available in OpenTURNS. `OTNormalDistribution` represents the normal distibution available in OpenTURNS and `OTJointDistribution` is the probability distribution of a random vector whose marginals are `OTDistributions`.

As mentioned in Figure 5, random vectors can also be added to a `ParameterSpace`. By default, only the marginal probability distributions are defined and the components of a random vector are stochastically independent. In the case of OpenTURNS distributions, the dependence structure can be defined using a copula function, e.g. Gaussian copula.

Then, just as the `DesignSpace` can apply a geometric transformation to project a point into the unit hypercube, the `ParameterSpace` can apply the inverse transformation method to sample over the unit hypercube and project these intermediate points into the probabilized input space. Here is an illustration of sampling where the `sample_disciplines` function returns a `Dataset`:

```
samples = sample_disciplines(
    disciplines, uncertain_space, output_names, n_samples=100
)
```

Once the uncertainty sources propagated, the classes `EmpiricalStatistics` and `ParametricStatistics`, deriving from the base class `BaseStatistics`, come into play to estimate the statistics of interest, including tolerance intervals (especially A- and B-value). The first one uses empirical estimators:

```
empirical_statistics = create_statistics(samples)
std = empirical_statistics.compute_standard_deviation()
```

while the second one uses the statistics of the best probability distributions, fitted from the output samples by either maximum likelihood or the method of moments, and selected from a list of candidates by a statistical hypothesis test or criterion:

```
parametric_statistics = create_statistics(
    samples,
    tested_distributions=["Exponential", "Normal", "Uniform"],
```

```
    fitting_criterion="Kolmogorov"
)
q10 = parametric_statistics.compute_quantile(0.1)
```

The `EmpiricalStatistics` class also provides the user with visualization tools, such as boxplots and plots of empirical probability density and cumulative probability functions:

```
boxplots = empirical_statistics.plot_boxplot()
```

All the figures have matplotlib [10] versions that can be modified locally to match the user's graphical presentation of results (color, line styles, labels, etc.). This applies to all the figures that can be generated by GEMSEO. An effort is underway to port these figures to plotly in order to make them dynamic in a web page and improve the user experience.

3.2 Quantile estimation using active learning algorithms

Because of the slow convergence of the Monte Carlo method, the estimates of some statistics can be flawed with a large variance, in particular when only a limited number of evaluations is available. A naive approach is to replace the function of interest by a surrogate model and sample it intensively. However, this requires a precise surrogate model, and so possibly a large training dataset. Another approach is to use active learning techniques that are iterative methods aiming to sequentially estimate various quantities of interest: optimas, contours, failure probabilities, quantiles, expected values, etc. These methods generally start by constructing a rough surrogate model (an approximation of a costly exact physical model much faster to run). Then, they iteratively look for new training points in order to update the surrogate model and increase the accuracy of the target quantity until a budget or accuracy criterion is reached. This search is guided by an acquisition criterion, also called infill criterion, to be minimized or maximized. This approach is inspired by the efficient global optimization (EGO) algorithm [12]. Another potential application of active learning techniques is the estimation of level-sets, i.e. the input values for which the model output is equal to a specific value. In particular, this specific value can be the α-quantile for a specific $\alpha \in]0, 1[$ [13].

`gemseo-mlearning` proposes the `ActiveLearningAlgo` class to perform an active learning study. This algorithm instantiates a `BaseAcquisitionCriterion` belonging to a `BaseAcquisitionCriterionFamily`, e.g. `EI` (expected improvement), `EF` (expected feasibility) or `U` (U-function) from the `Quantile` family [23, 19], in the case of quantile estimation. The `ActiveLearningAlgo` can also be used to explore the input space, optimize a function or estimate a level set. As illustrated in Figure 6, the `ActiveLearningAlgo` can use any DOE or optimization algorithm available in a `BaseDriverLibrary` to optimize the infill criterion and find the next point to enrich the surrogate model. It should be noted that point acquisition can be carried out sequentially or in batch mode, to take advantage of parallelized calculations. Lastly, the surrogate model used by the `ActiveLearningAlgo` can be any `BaseRegressor`, which is the base class for regression models in GEMSEO, and interfaced from any machine learning library. In fact, even if most often active learning technique uses a Gaussian process regressor, (`gemseo` proposes `GaussianProcessRegressor` based on scikit-learn and `OTGaussianProcessRegressor` based on OpenTURNS), it can be applied to a wide range of surrogate models [4].

3.3 Sensitivity analyses

Once the uncertainties have been propagated and the quantity of interest estimated, the user may want to perform a sensitivity analysis (SA) to explain this quantity from the dif-

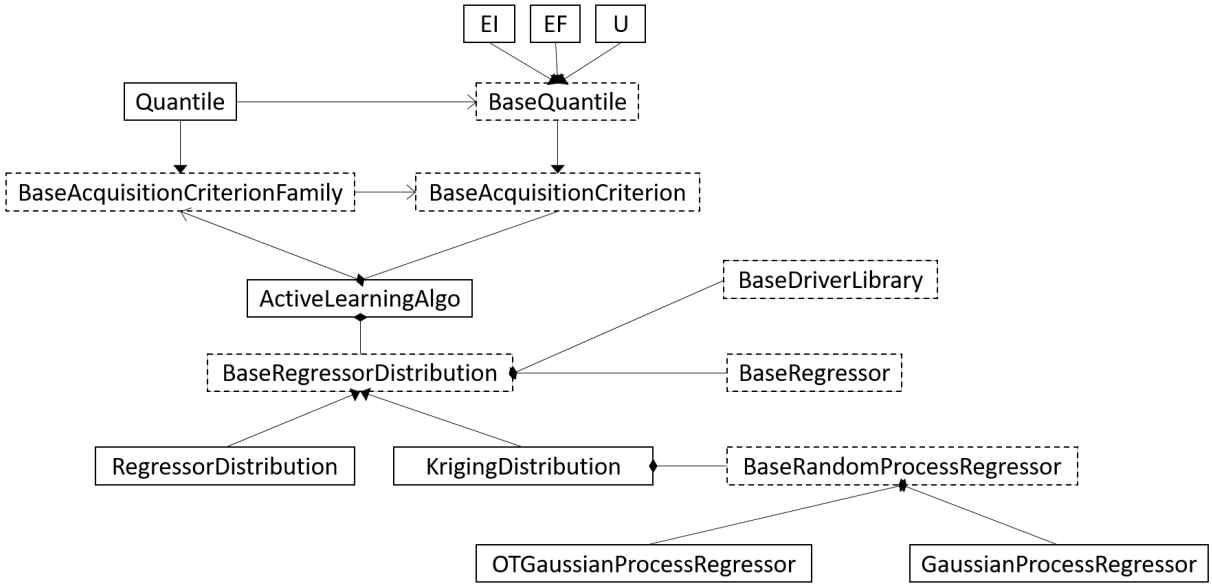


Figure 6: Class diagram of the active learning algorithm to estimate a quantile.

ferent sources of uncertainty. Figure 8 gives an overview of the classes implementing SA in `gemseo`. They derive from the base class `BaseSensitivityAnalysis`. Its method `compute_samples` is used to sample the multi-disciplinary system over an uncertain space using an `MDOFormulation`, a DOE which may be specific to the type of SA and a maximum number of evaluations. This is the costly task in a SA, and for this reason it is advisable to save the samples generated by this method. This means you can later instantiate a new SA from these samples, without having to regenerate them. Note also that by default, the sampling is performed for all outputs of all disciplines, including coupling variables, so that we can calculate sensitivity indices for all these variables. However, it is possible to select a subset of variables of interest, to avoid storing data that we will never use. Then, the method `compute_indices` computes the sensitivity indices from these samples. The indices can be displayed using generic graphs, such as bar plot and radar chart, or specific ones, whether related to the type of sensitivity analysis (see Figure 7) or to the functional nature of the quantity of interest. It is also possible to sort the sources of uncertainty, to prioritize uncertainty management, and compare the results of two different analyses on the same graph, to see whether or not they agree with the conclusions.

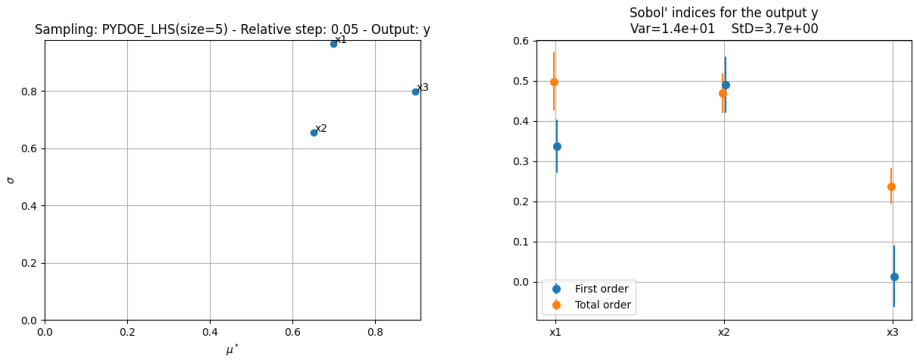


Figure 7: Visualization of the Morris indices (left) and Sobol indices (right) for the Ishigami function [11].

If we take a closer look, most of `SobolAnalysis`, `CorrelationAnalysis` and `HSICAnalysis` wrap features proposed by OpenTURNS while `MorrisAnalysis` is an in-house code. About the latter, you can perform a Morris analysis [15] using any DOE algorithm to place the trajectories in the input space, and must specify either the number of samples required, or the number of trajectories. You can also use the Morris sampling technique, or the underlying one-at-a-time (OAT) one in charge to generate a trajectory, independently of the `MorrisAnalysis`, thanks to the DOE algorithms `MorrisDOE` and `OATDOE`. Regarding `CorrelationAnalysis` and `HSICAnalysis`, they do not use a specific DOE algorithm (just Monte Carlo sampling), which makes them popular methods. The method `compute_indices` of `HSICAnalysis` proposes three SA variants, namely global SA (default) to identify the input variables most likely to cause the output to deviate from a nominal value, target SA to identify the input variables most likely to cause the output to reach a certain domain, and conditional SA to identify the input variables most likely to cause the output to deviate from a nominal value under the condition that the considered output is in a certain domain. Lastly, `SobolAnalysis` proposes different state-of-the-art algorithms to estimate the first-order, second-order and total indices from the samples. In the case of the Saltelli algorithm [24], the user can reduce the variance of the estimator at no extra cost, by providing cheap disciplines to be used as control variates, including surrogate models in the spirit of [7]. Using control variates is uncommon in UQ software libraries while the literature has demonstrated the value of these variance reduction methods, sometimes with several orders of magnitude.

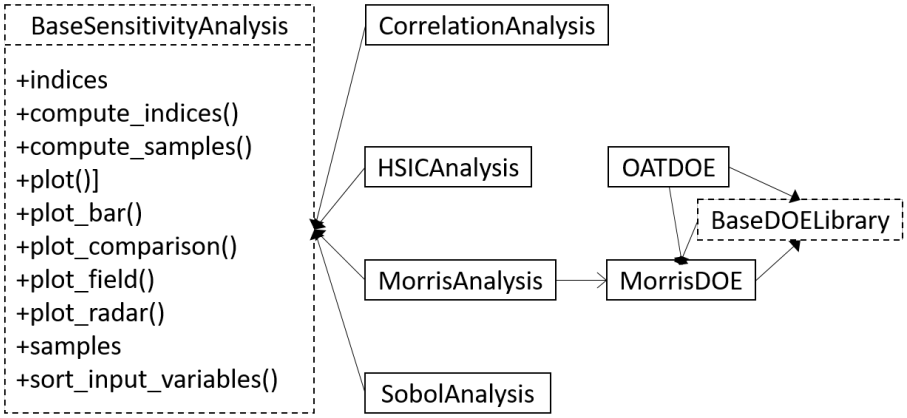


Figure 8: Class diagram of the sensitivity analysis algorithms.

3.4 Networks to visualize uncertainties

Calculating statistics is one thing, visualizing them is quite another, especially in high dimension, for example in presence of a large number of sources of uncertainty or many coupling variables. Networks are often used to represent this kind of data.

3.4.1 First-, second- and total-order Sobol' indices

Some ten years ago, [16] proposed the FANOVA graph to display the first-, second- and total-order Sobol' indices at the same time. A node represents an uncertain variable whose name is written inside, followed by its first-order and total-order Sobol' indices. The thickness of a node is proportional to the total-order Sobol' index of the variable while the thick-

ness of an edge is proportional to the second-order Sobol' index of the corresponding pair of variables. The `SobolGraph` class of the package `gemseo-umdo` implements the FANOVA graph, either from a `SobolAnalysis` or from the indices directly. The user can change the maximum thickness of a line and the significance level below which a second-order index is not displayed. Figure 9 illustrates this feature using the Ishigami function $f(X_1, X_2, X_3) = \sin(X_1) + 7\sin(X_2)^2 + 0.1X_3^4\sin(X_1)$ where X_1, X_2 and X_3 are independent and identically distributed according to the uniform distribution over $[-\pi, \pi]$ [11]. This clearly shows the first-order index of zero for X_3 and its interaction with X_1 which leads to a total index of 24%.

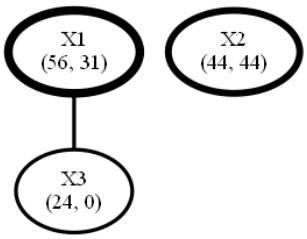


Figure 9: FANOVA graph for the Ishigami function.

3.4.2 Uncertain coupling variables

Usually, we perform a sensitivity analysis to identify the impact of uncertain inputs on a quantity of interest. In the case of an optimization problem, this quantity could be an objective or a constraint. In the case of a multi-disciplinary system, some disciplines may be less subject to uncertainties than others. Some disciplines might even be insensitive to uncertainties. Then, it would be wise not to re-evaluate them when sampling this system to estimate statistics of interest. For that reason, `gemseo-umdo` has an `UncertainCouplingGraph` class that identifies the coupling variables that are not impacted by the uncertain inputs, and so the disciplines that are not impacted by the uncertain inputs.

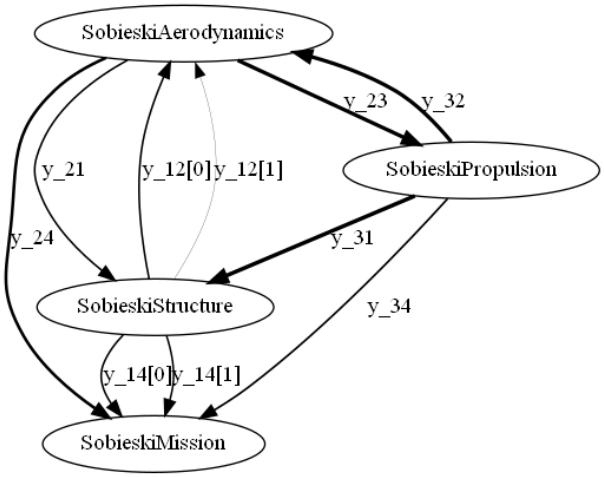


Figure 10: Uncertain coupling graph for the Sobieski's SSBJ problem.

Figure 10 illustrates this concept from the Sobieski's SSBJ problem [26] that includes three strongly coupled disciplines, called Aerodynamics, Propulsion and Structure, computing the

disciplinary constraints, and a fourth one, called Mission, computing the objective function from the others. In this example, the design variables are transformed into independent random variables and uniformly distributed over an interval of $\pm$ 5% around the optimum. In this network, the nodes represent the disciplines while the edges represent the coupling variables. The thickness of an edge is proportional to the absolute value of the dispersion of the corresponding coupling variable. A node connected only by ultra-thin edges, e.g. `y_12[1]`, will therefore be judged to be very insensitive to uncertain inputs. In terms of options, the user can select the dispersion measure (either the coefficient of variation or the quartile coefficient of dispersion) and the coupling variables of interest (displaying a lot of coupling variables can look like a plate of spaghetti).

4 MDO PROBLEMS UNDER UNCERTAINTY

When the sources of uncertainty have a significant impact on the objective and constraints, which are the variables of interest of the MDO problem, it is important to take them into account when defining the optimization problem. Originally, the GEMSEO team has created the `gemseo-umdo` plug-in to define and solve optimization problems under uncertainty of the form

$$\begin{array}{lll} \min\limits_{x} & \mathbb{K}_f[f(x,U)] & \\ \text{s.t.} & \mathbb{K}_g[g(x,U)] & \leq 0 \\ & \mathbb{K}_{h^-}[h(x,U)] & \leq 0 \\ & \mathbb{K}_{h^+}[h(x,U)] & \geq 0 \\ & x \in \mathcal{X} & \end{array} \tag{2}$$

where $\mathbb{K}_f$, $\mathbb{K}_g$ and $\mathbb{K}_h$ are statistical operators, e.g. expectation or variance, and f, g and h are objective and inequality constrained functions defined on both the design and uncertain spaces. Note that the uncertain parameters u are now replaced by random parameters U. The question of equality constraints under uncertainty remains an open problem, and we have chosen not to provide any particular capability to deal with them. Answering this question is part of the roadmap of `gemseo-umdo`. A naive approach is to replace them with both negativity and positivity constraints, as described in Problem (2). A more complex way would be to consider equality constraints for different realizations of U or for different statistics of $h(x,U)$ [6]. In the case of the IDF formulation, the consistency constraints would be addressed through this prism. However, this would increase significantly the dimension of the optimization, especially in the case of the approach with realizations.

4.1 An intuitive interface for UQ novices

The `UMDOScenario` class can be used to rewrite any MDO problem under uncertainty as Problem (2). Its interface is very similar to that of `MDOScenario` to ease the transition from a deterministic to an uncertain world, whether in terms of problem definition, resolution or results management. As in the case without uncertainties, the user must specify

- the disciplines outputting the variables of interest,
- the disciplinary output f to minimize or maximize,
- the disciplinary outputs g and h to be constrained,
- the design space in which to perform the optimization,

- the MDO formulation to create the multi-disciplinary process to evaluate the functions f, g and h.

In the specific context of uncertainties, the user must also specify:

- the uncertain space,
- the statistical operators for the constraints and objective,
- the statistics estimation technique (see Section 4.4).

As you can see in the following code, it is easy to go from an MDO problem without uncertainty (left) to one with uncertainty (right):

```
scenario = MDOScenario(
    [disciplines],
    "obj",
    design_space,


    formulation_name="MDF",



)  # min obj(x,u_bar) over x_space
scenario.add_constraint(
    "g"
)  # s.t. g(x,u_bar) <= 0
```

```
scenario = UMDOScenario(
    [disciplines],
    "obj",
    design_space,
    uncertain_space,
    "Mean",
    formulation_name="MDF",
    statistic_estimation_parameters={
        "n_samples": 100
    },
)  # min E[obj(x,U)] over x_space
scenario.add_constraint(
    "g", "Margin", factor=3.0
)  # s.t. E[g(x,U)]+3S[g(x,U)] <= 0
```

Once created, the user can execute this `UMDOScenario` with any optimization or DOE algorithm interfaced to `gemseo` and then post-process the results, in the same way as in the case without uncertainty:

```
scenario.execute(algo_name="NLOPT_COBYLA", max_iter=50)
scenario.post_process("OptHistoryView")
```

4.2 U-MDO formulation engine

Section 4.1 showed that the user must define a pair of MDO formulation and statistics estimation technique. `gemseo-umdo` calls this duo a U-MDO formulation. It can be seen as a nested multi-disciplinary process. The inner process is orchestrated by the MDO formulation and generates the functions

$$u \mapsto f(x,u) \qquad u \mapsto g(x,u) \qquad u \mapsto h(x,u)$$

defined over the uncertain space for a given value of x. The outer process is orchestrated by the statistics estimation technique and generates the functions

$$x \mapsto \widehat{\mathbb{K}}_f[f(x,U)] \qquad x \mapsto \widehat{\mathbb{K}}_g[g(x,U)] \qquad x \mapsto \widehat{\mathbb{K}}_h[h(x,U)]$$

where $\widehat{\mathbb{K}}_f$, $\widehat{\mathbb{K}}_g$ and $\widehat{\mathbb{K}}_h$ are the estimators of the statistics $\mathbb{K}_f$, $\mathbb{K}_g$ and $\mathbb{K}_h$ respectively. Finally, the estimators $\widehat{\mathbb{K}}_f$, $\widehat{\mathbb{K}}_g$ and $\widehat{\mathbb{K}}_h$ are plugged into the Problem (2) which is ready to be solved numerically. We can thus see that a so-called U-MDO formulation is indeed an MDO formulation in the MDO sense, namely both a workflow to evaluate the objective and constraints functions

expressed as statistics estimators, and a dataflow to connect the disciplines and sub-processes using labelled data. Therefore, `UMDOScenario` is a unified way to extend the MDO formulations usable with `gemseo` (MDF, IDF, multi-level ones) to the frame of U-MDO. In the same way as `gemseo` offers an MDO formulation engine, `gemseo-umdo` offers a U-MDO formulation engine, illustrated by the class diagram in Figure 11.

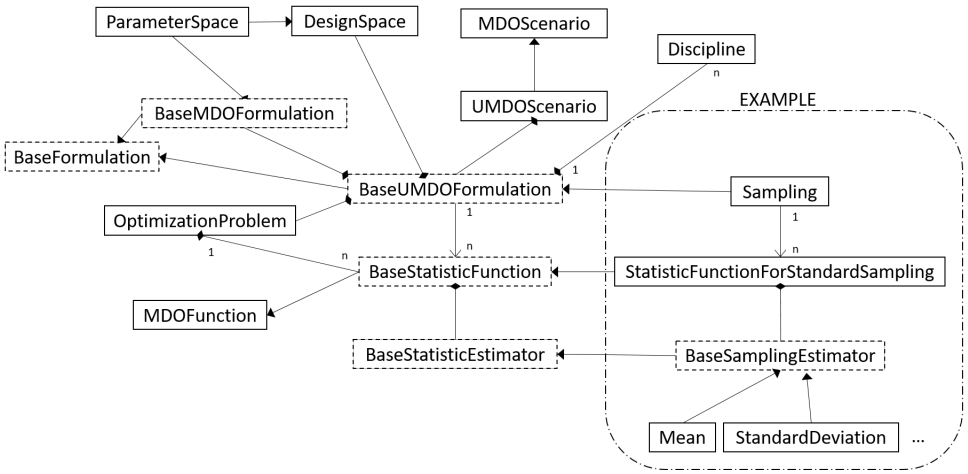


Figure 11: Class diagram of the U-MDO formulation engine. A `UMDOScenario` is a `MDOScenario` whose `BaseFormulation` is a `BaseUMDOFormulation`, e.g. `Sampling` when estimating the statistics using a sampling technique. This `BaseUMDOFormulation` is also composed of a `BaseMDOFormulation`, e.g. `MDF`. The `BaseMDOFormulation` includes a `ParameterSpace`, defining the uncertain variables, while the `BaseUMDOFormulation` includes a `DesignSpace`, defining the design variables. The `OptimizationProblem` composing the `BaseUMDOFormulation` uses `BaseStatisticFunctions` as `MDOFunctions`, estimating the statistics helped by a `BaseStatisticEstimator`. `BaseStatisticFunction` and `BaseStatisticEstimator` are subclassed for a specific `BaseUMDOFormulation`, e.g. `StatisticFunctionForStandardSampling` and `BaseSamplingEstimator`. Finally, `BaseSamplingEstimator` derives into one class per statistic, e.g. `Mean` and `StandardDeviation`, whose class names are used as keywords by the end-user (see `"Mean"` and `"Margin"` in the previous code lines).

In an easier-to-read format, Figure 12 is a graphical view of this feature when combining Monte Carlo (MC) sampling (outer loop in orange) and an MDF formulation (inner loop in blue), with a Jacobi algorithm for the MDA. In this example, the MDO problem is unconstrained and the expectation is the statistical operator for the objective. In the absence of uncertainty, the outer loop would not appear and the discipline D_0 would directly return $f(x, u)$ to the optimizer at top left. In the next sub-section, we illustrate it with an application to the Sobieski's Super-Sonic Business Jet (SSBJ) problem [26].

4.3 Illustration on an aeronautical application

In [26], the goal was to maximize the range y_4 of a SSBJ under the constraints $g_i \leq 0,\ 1 \leq i \leq 3$, with respect to different design variables, such as the shared design variables x_{shared}. In this study, we consider this design variable as uncertain and model by the random variable $X_{shared} = dv_{x_{shared}} + U_{x_{shared}}$ where $dv_{x_{shared}}$ is the design vector controlled by the optimizer and $U_{x_{shared}}$ is a random vector with zero mean. The minimum drag coefficient C_4 is also considered as uncertain. Now, the goal is to maximize the expectation of the range y_4 under the constraints $\mathbb{E}[g_i] + 2\mathbb{S}[g_i] \leq 0,\ 1 \leq i \leq 3$ and the uncertain variables $U_{x_{shared}}$ and C_4, with respect to different design variables. The following lines describe how to define and solve this

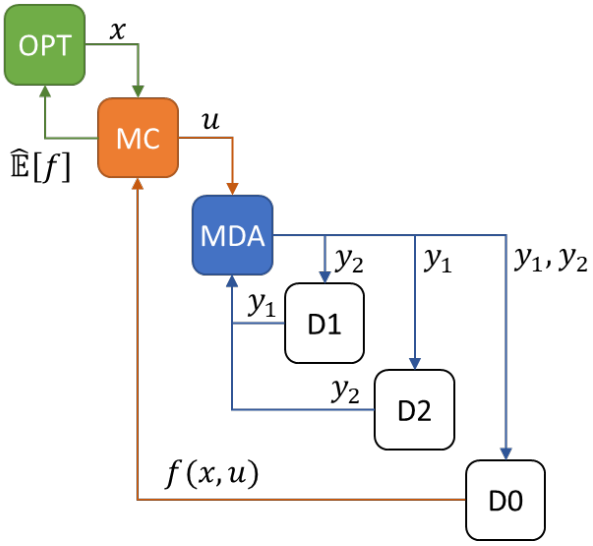


Figure 12: An optimization problem relying on a U-MDO formulation combining MC sampling and an MDF formulation, based on a Jacobi MDA algorithm, with $\widehat{\mathbb{E}}[f(x,U)] = \frac{1}{N}\sum_{i=1}^{N} f(x,u^{(i)})$.

U-MDO problem:

```
from gemseo import configure_logger
from gemseo.algos.parameter_space import ParameterSpace
from gemseo.problems.mdo.sobieski.core.problem import SobieskiProblem
from gemseo.problems.mdo.sobieski.disciplines import SobieskiAerodynamics
from gemseo.problems.mdo.sobieski.disciplines import SobieskiMission
from gemseo.problems.mdo.sobieski.disciplines import SobieskiPropulsion
from gemseo.problems.mdo.sobieski.disciplines import SobieskiStructure

from gemseo_umdo.scenarios.umdo_scenario import UMDOScenario

configure_logger()
```

Firstly, we instantiate the disciplines:

```
mission = SobieskiMission()
structure = SobieskiStructure()
propulsion = SobieskiPropulsion()
aerodynamics = SobieskiAerodynamics()
```

and the design space:

```
design_space = SobieskiProblem().design_space
```

Secondly, we define the uncertain space:

```
uncertain_space = ParameterSpace()
```

including the uncertain physical property C_4 which is distributed as the triangular distribution $\mathcal{T}(0.85c_4, c_4, 1.15c_4)$ where $c_4 = 0.01375$:

```
uncertain_space.add_random_variable(
    "c_4", "OTTriangularDistribution",
    minimum=0.01375*0.85, mode=0.01375, maximum=0.01375*1.15
)
```

and the uncertain design variable $X_{shared} = dv_{x_{shared}} + U_{x_{shared}}$ where $U_{x_{shared}}$ is distributed as the triangular distribution $\mathcal{T}(-0.15, u_{x_{shared}}, 0.15)$ where $u_{x_{shared}} = 0$:

```
uncertain_space.add_random_variable(
    "u_x_shared", "OTTriangularDistribution",
    minimum=-0.15, mode=0.0, maximum=0.15
)
```

Then, we build a `UMDOScenario` to maximize the expectation of the range $\mathbb{E}[y_4]$ estimated by 50 samples:

```
scenario = UMDOScenario(
    [mission, structure, propulsion, aerodynamics],
    "y_4",
    design_space,
    uncertain_space,
    "Mean",
    formulation_name="MDF",
    # By default, UMDOScenario estimates the statistics by sampling.
    statistic_estimation_parameters={"n_samples": 50},
    maximize_objective=True,
    # There is a convention to make a design variable uncertain.
    uncertain_design_variables={"x_shared": ("+", "u_x_shared")},
)
```

while satisfying margin constraints of the form $\mathbb{E}[g_i] + 2\mathbb{S}[g_i] \leq 0$:

```
scenario.add_constraint("g_1", "Margin")
scenario.add_constraint("g_2", "Margin")
scenario.add_constraint("g_3", "Margin")
```

and execute it with the gradient-based optimizer SLSQP [18]:

```
scenario.execute(algo_name="NLOPT_COBYLA", max_iter=100)
```

The solution is feasible, with an objective equal to 3890.33 nautical miles, which is more conservative than the solution without uncertainties (3963.88).

Lastly, we can plot the optimization history view (see Figure 13) to analyze the convergence of the algorithm:

```
scenario.post_process(post_name="OptHistoryView", save=False, show=True)
```

4.4 Statistics estimation techniques

`gemseo-umdo` can use any type of statistics estimation technique and already offers some standards, such as sampling, surrogate models and variance reduction methods. The choice of the estimation technique applies to all statistics $\mathbb{K}_f$, $\mathbb{K}_g$ and $\mathbb{K}_h$. In the rest of the section, we will limit the mathematical expressions to the function f with scalar uncertain variable U for the sake of clarity.

4.4.1 Sampling

The most naive approach consists of using MC estimators. It is also possible to replace crude MC by a quasi MC approach, such a low-discrepancy sequence or Latin hypercube sampling. `gemseo` offers different implementations of these DOE techniques based on SciPy [22], pyDOE3 and OpenTURNS. The U-MDO formulation *Sampling* can use any of them to generate the input samples $\{u^{(1)}, \ldots, u^{(N)}\}$. Then, the user can choose to compute all the output samples before estimating the statistics, or to estimate the statistics iteratively by using new OpenTURNS features.

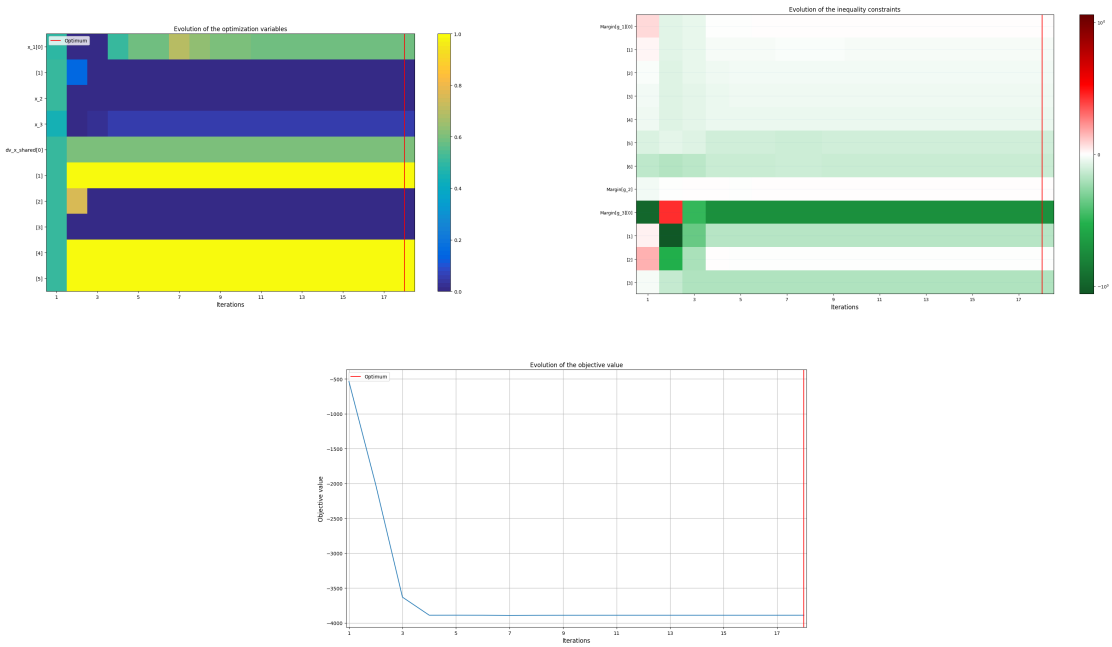


Figure 13: Optimization history view of the SSBJ's U-MDO problem.

These estimators are often unbiased, which is an interesting property. However, if some certain disciplines are costly to evaluate, the number of samples N will be limited and the variance of the estimators large. This is the main limitation of the sampling techniques.

4.4.2 Taylor polynomial

When evaluating the inner loop of the U-MDO formulation is costly, approximating $u \mapsto f(x,u)$ by a first-order Taylor polynomial at $\mu = \mathbb{E}[U]$ may be relevant, especially in the case of negligible uncertain sources U:

$$f(x,U) \approx \hat{f}(x,U) = f(x,\mu) + (U-\mu)\partial_u f(x,\mu).$$

Then, the expectation and variance reduce to $f(x,\mu)$ and $\mathbb{V}[U]\left(\partial_u f(x,\mu)\right)^2$ respectively, which requires only one evaluation and one differentiation of f at each iteration of the optimization process. If the partial derivative $\partial_u f$ is missing, it can be approximated by finite differences for example, which results in not 1 but $1+d$ evaluations of f per iteration, where d is the dimension of the design variable x. In low dimension, the U-MDO formulation *TaylorPolynomial* is still cheaper than sampling. The quality of these estimators could be improved if the second-order derivatives can be approximated. On the other hand, it can only be used for statistics combining expectations and variances.

4.4.3 Surrogate models

A Taylor polynomial can be considered as a linear surrogate model. If f is poorly linear with respect to u, the approximation can be quite poor and switching to a non-linear surrogate model can be a good option. `gemseo-umdo` offers the possibility to create a surrogate model of $x \mapsto f(x,u)$ over the uncertain space and estimate the statistics with intensive MC sampling at each iteration of the optimization loop. During the optimization process, information on the

quality of the surrogate models is logged, both in terms of learning and validation, to alert the user in the event of poor approximation.

The U-MDO formulation *Surrogate* allows the user to select any regression model interfaced to `gemseo`. The latter and its extension `gemseo-mlearning` provide access to several models from the libraries scikit-learn, OpenTURNS, SMT and SciPy, such as polynomial regression, Gaussian process regression, radial basis function regression and polynomial chaos expansion (PCE). In the special case of PCE, the U-MDO formulation *PCE* does the same thing, except that it estimates the statistics analytically from the PCE coefficients, which limits its use to combinations of expectations and variances (the expectation of a PCE is equal to its offset term it is equal to the sums of the squares of the other coefficients).

4.4.4 Control variates

Most of the time, the surrogate model $\hat{f}$ guarantees that $\frac{1}{N}\sum_{i=1}^{N}\hat{f}(x,U^{(i)})$ is an unbiased estimator of $\mathbb{E}[f(x,U)]$, while it is an interesting statistic estimator property. On the other hand, the estimator $E_N[\hat{f};x]$ may be strongly correlated to $E_N[f;x]$, with $E_N[f;x] = \sum_{i=1}^{N} f(x,U^{(i)})/N$, and thus could be a good control variate (CV) [7]. CV techniques are reduction variance methods to make an MC estimator more accurate. In the case of expectation, the surrogate-based CV estimator is:

$$E_{CV,N}[f;x] = E_N[f,x] - \alpha_N\left(E_N[f;x] - \mathbb{E}[f(x,U)]\right)$$

with $\alpha_N = \frac{C_N[f,\hat{f};x]}{V_N[f;x]}$, where C_N and V_N are the MC estimator of the covariance between $\hat{f}(x,U)$ and $f(x,U)$ and the MC estimator of the variance of $f(x,U)$ respectively. The U-MDO formulation *ControlVariate* can estimate the statistics using surrogate-based control variate techniques. By default, the surrogate model is a first-order Taylor polynomial.

4.4.5 Summary

Table 1 summarizes the different statistics estimation techniques. It is important to note that even though the sampling was done according to the uncertain space and the PCE was built on this same space, the statistics obtained with the U-MDO formulations *Sampling* and *PCE* are differentiable with respect to the design variables. This means that if the disciplines provide the derivatives and the formulation is differentiable, then you can use a gradient-based optimizer to solve the optimization problem without having to approximate the derivatives. This is particularly promising for using gradient algorithms to converge faster.

4.5 Benchmark problems

We have just seen that there are various statistical estimators, just as there are various optimizers, MDO formulations and MDA algorithms. All combined, it can be complicated to know which is the best combination for you. Faced with such a wealth of combinations, a question arises: which one to choose? Just as there is no free lunch for optimization algorithms, there is no a priori choice for the other components, and one way of doing this is to choose a combination by experience, or to select one based on a benchmark study. The literature on problems to benchmark optimization algorithms is much richer than the literature on problems to benchmark MDO algorithms. This difference is even greater in the case of MDO under uncertainties.

In the face of this reality, `gemseo` proposes different benchmark MDO problems that are scalable, in the sense that the user can chose the number of disciplines, the dimensions of the

	Sampling	Taylor	Surrogate	PCE	ControlVariate
Expectation	✓	✓	✓	✓	✓
Standard Deviation	✓	✓	✓	✓	✓
Variance	✓	✓	✓	✓	✓
Margin	✓	✓	✓	✓	✓
Probability	✓		✓		✓
Differentiable	✓			✓	
Cost	High	Small	Medium	Medium	Medium
Options	Iterative	Second-order	Model Settings Quality measure	Settings Quality measure	Model Settings

Table 1: Summary of the statistics estimation techniques to create a U-MDO formulation. The notion of evaluation cost is subjective and case-dependent. A greater number of U-MDO formulations will be differentiable in the future, especially *Surrogate* and *ControlVariate*.

design variables and the dimension of the coupling variables. In particular, it proposes an MDO problem in which the mathematical characteristics of disciplines reflect those of user disciplines, thanks to frugal sampling [29]. An extension of the scalable quadratic MDO problem proposed by [27] is also implemented, in which the user can define uncertainties in the coupling equations in particular, as well as the percentage of feasible design space [2]. In addition, classic MDO problems are implemented, such as Sellar's problem [25], supersonic business jet (SSBJ) problem [26] and propane combustion problem [27]. It is easy to extend them to the uncertain case, by replacing certain constants with random variables or considering some design variables as uncertain, as illustrated in Section 4.3.

Lastly, `gemseo` offers the `OptAsMDOScenario` class, which is an `MDOScenario` making a monodisciplinary optimization problem multi-disciplinary, by applying a method proposed by the authors of [2]. The original discipline must have as inputs at least three design variables and as outputs one or more objectives. It may also have as outputs one or more constraints. The MDO problem will include $2 + N$ disciplines, namely

- N strongly coupled disciplines computing the values of some coupling variables from the values of the other coupling variables and the values of the design variables,
- one weakly coupled discipline, called *link discipline*, computing the values of the design variables in the original optimization problem from the values of the design and coupling variables in the MDO problem,
- the original discipline.

By default, the strongly coupled disciplines and the link discipline are linear and the user can pass its own disciplines. For example, `gemseo-umdo` extends the `OptAsMDOScenario` class into `UOptAsUMDOScenario`, to make an optimization problem under uncertainty multi-disciplinary. This approach allowed us to rewrite the Rosenbrock optimization problem [21] in dimension 3 as a U-MDO problem with linear coupling equations:

```
discipline = AnalyticDiscipline(
    {"f": "100*(z_2-(u*z_1)**2)**2+(1-v*z_1)**2+100*(z_1-(u*z_0)**2)**2+(1-
   v*z_0)**2"},
    name="Rosenbrock",
```

```
)

design_space = DesignSpace()
design_space.add_variable("z_0", lower_bound=-1, upper_bound=1)
design_space.add_variable("z_1", lower_bound=-1, upper_bound=1)
design_space.add_variable("z_2", lower_bound=-1, upper_bound=1)

uncertain_space = DesignSpace()
uncertain_space.add_random_variable(
    "u", "OTNormalDistribution", mu=1., sigma=0.01
)
uncertain_space.add_random_variable(
    "v", "OTNormalDistribution", mu=1., sigma=0.01
)

scenario = UOptAsUMDOScenario(
    discipline,
    "f",
    design_space,
    uncertain_space,
    "Mean",
    formulation_name="MDF"
)
scenario.execute(algo_name="NLOPT_SLSQP", max_iter=100)
```

Figure 14 shows the discipline coupling graph associated with this problem, and the section ends with the start of the log associated with this resolution. You can notice that the optimization variables are renamed into x_0, x_1, x_2 because the variables z_0, z_1, z_2 are now coupling variables.

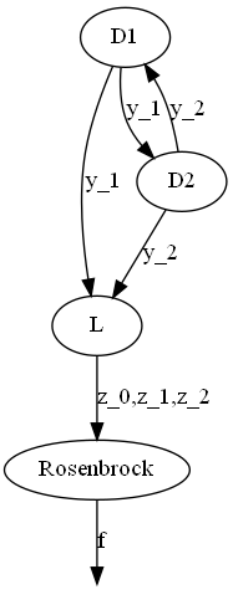


Figure 14: Discipline coupling graph in a multidisciplinary of the Rosenbrock problem under uncertainty, with the two strongly coupled discipline D_1 and D_2, the link discipline L and the standard Rosenbrock function.

```
INFO - 16:43:14: *** Start OptAsMDOScenario execution ***
INFO - 16:43:14: OptAsMDOScenario
```

```
INFO - 16:43:14:    Disciplines: D1 D2 L Rosenbrock
INFO - 16:43:14:    Formulation:
INFO - 16:43:14:       MDO formulation: MDF
INFO - 16:43:14:       Statistic estimation: Sampling
INFO - 16:43:14:    Uncertain space:
INFO - 16:43:14:       +------+---------------------------+
INFO - 16:43:14:       | Name |        Distribution       |
INFO - 16:43:14:       +------+---------------------------+
INFO - 16:43:14:       |  u   | Normal(mu=1.0, sigma=0.01) |
INFO - 16:43:14:       +------+---------------------------+
INFO - 16:43:14:       |  v   | Normal(mu=1.0, sigma=0.01) |
INFO - 16:43:14:       +------+---------------------------+
INFO - 16:43:14: Optimization problem:
INFO - 16:43:14:    minimize E[f(x_0, x_1, x_2)]
INFO - 16:43:14:    with respect to x_0, x_1, x_2
INFO - 16:43:14:    over the design space:
INFO - 16:43:14:       +------+-------------+-------+-------------+-------+
INFO - 16:43:14:       | Name | Lower bound | Value | Upper bound | Type  |
INFO - 16:43:14:       +------+-------------+-------+-------------+-------+
INFO - 16:43:14:       | x_0  |      -1     | -0.25 |      1      | float |
INFO - 16:43:14:       | x_1  |      -1     |  0.75 |      1      | float |
INFO - 16:43:14:       | x_2  |      -1     |  -0.9 |      1      | float |
INFO - 16:43:14:       +------+-------------+-------+-------------+-------+
INFO - 16:43:14: Solving optimization problem with algorithm NLOPT_SLSQP:
...
```

5 CONCLUSIONS

In this work, we have presented the extension of `gemseo` to UQ features, facilitated by the object-oriented paradigm intensively used.

First, we introduced the basic classes on which the UQ features rely. It was an opportunity to demonstrate the versatility of `gemseo`, capable to automate the creation of evaluation problems from user-friendly disciplines defined by data labels and an MDO formulation that defines the computational process.

Then, we introduced how to use `gemseo` and its extension `gemseo-umdo` to propagate uncertainties through a multi-disciplinary process and estimate the quantities of interest either empirically or parametrically. Graphical techniques were presented for uncertainty visualization in coupling graphs, in order to detect the paths of greatest variability.

We have also presented the sensitivity analysis techniques that separate the data generation process from the sensitivity indices estimation. This allows to save the samples and calculate the indices or perfect the graphs later, without having to resample the multi-disciplinary system. We have shown that it is possible to extend such techniques, e.g. the Morris method with new DOE algorithm to place the trajectories.

We also introduced the active learning capabilities available in `gemseo-mlearning`, with a focus on quantile estimation. This features can easily been extended to new machine learning libraries to create the surrogate models (typically Gaussian process regressors) [5] and new acquisition criteria.

Thirdly, we have tackled the extension of the MDO framework to account for uncertainties. To do this, we have shown how the new concepts are based on GEMSEO's basic classes, once again proving its genericity, and presented different estimation techniques for the statistics of objective and constraints. These features were illustrated on an academic use case from the

aeronautical litterature and the section concludes with benchmarks problems to compare U-MDO algorithms, combining both estimation techniques and MDO formulations.

In this work, we have also discussed the promising use of surrogate-based control variates to estimate the statistics more precisely, for whether Sobol' analysis or U-MDO problems, and highlighted that a surrogate can be a simple and cheap first-order Taylor polynomial.

In the future, we would like to illustrate the application of the U-MDO formulation engine with a wide selection of MDO formulations, including IDF and bi-level ones. In particular, we will show how `gemseo-umdo` can implement the bi-level U-MDO formulation proposed by [1] and inspired by the uncertainty-free bi-level formulation [9]. This U-MDO formulation returns both the optimum of the global design variables as fixed values and the optimum of the local variables as random variables. The latter depends on the sources of uncertainty that we hope will decrease during the design cycle and thus lead to better solutions. We can also propose a bi-level U-MDO formulation that seeks to optimize the local and global design variables at the same time. We also plan to increase the use of surrogate models in the UQ context, either trained offline or online for active learning purposes. In this context, new use of gradient-based techniques would find its place in U-MDO, to improve convergence and accuracy of optimization processes, e.g. gradient-enhanced and differentiable surrogate models. Bridging the gap between active learning techniques and U-MDO would also be a logical extension of the work done in GEMSEO. Lastly, there is an ongoing effort to improve interfaces and documentation, advise on the choice of U-MDO formulations depending on the computational budget and the properties of the problem, and thus democratize access to MDO techniques under uncertainty.

ACKNOWLEDGEMENTS

This study has been supported by the NEXTAIR project which has received funding from the European Union's Horizon Europe research and innovation program under grant agreement No 101056732. Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

REFERENCES

[1] Aziz-Alaoui, A., *Contributions to multidisciplinary design optimization under uncertainty, with applications to aircraft design*, Thèse de doctorat, Université de Toulouse, 2025.

[2] Aziz-Alaoui, A., Roustant, O. and De Lozzo, M. *A scalable problem to benchmark robust multi-disciplinary design optimization techniques*. Optim Eng 25, 941–958, 2024.

[3] Baudin, M., Dutfoy, A., Iooss, B., Popelin, AL, *OpenTURNS: An Industrial Software for Uncertainty Quantification in Simulation*. In: Ghanem, R., Higdon, D., Owhadi, H. (eds) Handbook of Uncertainty Quantification. Springer, Cham, 2015.

[4] Ben Salem, M., Roustant, O., Gamboa, F., and Tomaso, L., *Universal prediction distribution for surrogate models*. SIAM/ASA Journal on Uncertainty Quantification, 5:, 12 2015

[5] Bouhlel, M. A., and Hwang, J. T., and Bartoli, N., and Lafage, R., and Morlier, J., and Martins, J. R. R. A., *A Python surrogate modeling framework with derivatives*. Advances in Engineering Software, 2019.

[6] Brevault, L. and Balesdent, L. *Uncertainty-Based multi-disciplinary Design Optimization (UMDO)*. In: Aerospace System Analysis and Optimization in Uncertainty. Springer Optimization and Its Applications, **156**, Springer, Cham, 2020.

[7] El Amri, M. R., Mycek, P., Ricci, S., and De Lozzo, M., *Multilevel surrogate-based control variates*. arXiv preprint arXiv:2306.10800.

[8] Gallard, F., Vanaret, C., Guénot, D, et al., *GEMS: A Python Library for Automation of multi-disciplinary Design Optimization Process Generation*. In : 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 2018.

[9] Gazaix, A., et al, *Industrial Application of an Advanced Bi-level MDO Formulation to Aircraft Engine Pylon Optimization*. In : AIAA Aviation 2019 Forum, 2019.

[10] Hunter, J. D., *Matplotlib: A 2D Graphics Environment*. Computing in Science & Engineering, vol. 9, no. 3, pp. 90-95, 2007.

[11] Ishigami, T., and Homma, T, *An importance quantification technique in uncertainty analysis for computer models*. In First International Symposium on Uncertainty Modeling and Analysis, 1990.

[12] Jones, D., Schonlau, M., and Welch, W., *Efficient global optimization of expensive black-box functions*. Journal of Global optimization, 13:455–492, 1998.

[13] Laboulfie, C., De Lozzo, M., Grotto, F., Barrière, L., Navarro, J.-P., Bocquet, S., *Parallel Active Learning for Quantile Estimation from Composites Damage Models*.

[14] Martins, J. R. R. A., and Lambe, A. B., *multi-disciplinary Design Optimization: A Survey of Architectures*. AIAA Journal Vol. 51 (9) , 2049-2075, 2013.

[15] Morris, M.D., *Factorial Sampling Plans for Preliminary Computational Experiments*, Technometrics. 33 (2): 161–174, 1991.

[16] Muehlenstaedt, T., Roustant, R., Carraro, L., and Kuhnt, S., *Data-driven Kriging models based on FANOVA-decomposition*. Statistics and Computing, 22:723-738, 2012.

[17] Pedregosa et al., *Scikit-learn: Machine Learning in Python*, JMLR 12, pp. 2825-2830, 2011.

[18] Powell, M.J.D., *A direct search optimization method that models the objective and constraint functions by linear interpolation*. in Advances in Optimization and Numerical Analysis, eds. S. Gomez and J-P Hennart, Kluwer Academic (Dordrecht), pp. 51-67, 1994.

[19] Ranjan, P., Bingham, D., and Michailidis, G., *Sequential experiment design for contour estimation from complex computer codes*. Technometrics, 50(4):527–541, 2008.

[20] de Rocquigny, E., Devictor, N., and Tarantola, S. (Eds.), *Uncertainty in industrial practice: a guide to quantitative uncertainty management*. John Wiley & Sons, 2008.

[21] Rosenbrock, H.H., *An automatic method for finding the greatest or least value of a function*. The Computer Journal. 3 (3): 175–184, 1960.

[22] Roy et al, *Quasi-Monte Carlo Methods in Python*. Journal of Open Source Software, 8(84), 5309, 2023.

[23] Roy, S., and Notz, W., *Estimating percentiles in computer experiments: a comparison of sequential-adaptive designs and fixed designs*.Journal of Statistical Theory and Practice, Springer, 8(1), 2014

[24] Saltelli, A., Annoni, P., Azzini, I., Campolongo, F., Ratto, R. and Tarantola, S., *Variance based sensitivity analysis of model output. design and estimator for the total sensitivity index*. Computer physics communications, 181(2):259–270, 2010.

[25] Sellar, R., Batill, S., and Renaud, J., *Response surface based, concurrent subspace optimization for multi-disciplinary system design*. 34th Aerospace Sciences Meeting and Exhibit, 1996.

[26] Sobieszczanski-Sobieski, J., Agte, J. S., and Sandusky, R. R. S., Jr., *Bi-Level Integrated System Synthesis (BLISS)*. Tech. Rep. TM-1998-208715, NASA, Langley Research Center, 1998.

[27] Tedford, N., and Martins, J. R. R. A., *Benchmarking multi-disciplinary design optimization algorithms*. Optimization and Engineering 11, 159–183, 2010.

[28] The pandas development team, 10.5281/zenodo.3509134, 2020.

[29] Vanaret, C., Gallard, F., and Martins, J. R. R. A., *On the Consequences of the "No Free Lunch" Theorem for Optimization on the Choice of an Appropriate MDO Architecture*. 18th AIAA/ISSMO multi-disciplinary Analysis and Optimization Conference, 2017.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

A MACHINE LEARNING APPROACH FOR A 2-POINT STATISTICAL HOMOGENIZATION METHOD

Luzie Schmollack[1] **and Sandra Klinge**[1]

[1]Technical University Berlin
Strasse des 17. Juni 135, 10623 Berlin
e-mail: {luzie.schmollack, sandra.klinge}@tu-berlin.de

Abstract. *Statistical homogenization methods are widely used to predict the response of a heterogeneous material to loading. A key challenge in applying these methods is the choice of a suitable probability function that accurately captures the material's spatial correlations. However, limitations in both experimental and theoretical tools often lead to difficulties in approximating the statistical descriptor effectively. High computational demands make modelling these complex structures even more complex. To address these challenges, we propose a machine learning based approach to identify a correct correlational descriptor for a class of 2D two-phase-microstructures. A combination of neural networks is used to extract the probability functions from the microstructure image focusing purely on the geometry of the given structure. By coupling a fully connected neural network (FCNN) and a convolutional neural network (CNN), both trained simultaneously, our approach significantly reduces the data requirements, allowing for effective training with a relatively small data set.*

Keywords: Machine Learning, Correlation Functions, Material Modelling, Microstructure Analysis, Data-Driven Neural Network.

1 INTRODUCTION

In the field of mechanical engineering, there is a wide variety of methods predicting effective material properties under different loading cases such as the multiscale FEM approach [7]. Here, selecting an appropriate representative volume element (RVE) is an essential step, as it provides information necessary for simulating the behavior of a macroscopic body. The connection between length scales is based on the principle of volume averaging and the Hill-Mandel macrohomogeneity condition. The latter ensures the derivation of various boundary conditions for the RVE, thereby establishing a well-posed problem at this level. A closely related problem is the inverse analysis identifying material parameters of nonlinear composites based on the experimental data [6]. Crucial issues here are the investigation of problems involving a large number of unknown material parameters leading to the non-uniqueness of the solution as well as addressing the ill-posedness and determination of the global minimum. In recent years, Machine Learning (ML) techniques have proven to be highly effective in this field. A variety of approaches are available, including data-driven and physics-informed neural networks [5, 3], convolutional neural networks (CNNs) [9], and even neural operators [4]. For instance, in [5], an extendable, efficient, and explainable ML approach is proposed to model cyclic plasticity and replace conventional material models. This approach combines data-driven and physics-informed network architectures. For statistical homogenization, an explainable statistical represantation on micrstructures using CNNs is presented in [2] . In the present work, we follow a statistical homogenization approach to describe spatial correlation in 2D two-phase microstructures, describing the overall appraoch in section 2. Data and a classical statistical method for correlation function computation are described in section 3, while the Neural Network architecture to replace the process is introduced in section 4, followed by a conclusion and outlook.

2 STATISTICAL HOMOGENIZATION

In materials science, statistical homogenization is a powerful method for predicting the macroscopic response of heterogeneous materials to loading. This approach is particularly useful in cases where materials exhibit complex microstructures, as it enables the derivation of effective properties by considering statistical characteristics of the underlying phases. A crucial aspect of statistical homogenization is the computation of n-point correlation functions. They serve as spatial descriptors of different phases within the material. For two-phase materials, the volume fractions of each phase can be described using 1-point correlation functions.

In this work, we present a neural network architecture designed to predict 2-point correlation functions from microstructure images, addressing a key challenge for statistical homogenization methods in computational materials science. Statistical methods are widely employed to predict material behavior because of their ability to capture the underlying patterns within heterogeneous materials [1, 8]. In their undeformed state, most heterogeneous materials can be effectively described using 2-point correlation functions [10]. For elastic materials, particularly those exhibiting linear elastic behavior, 2-point correlation functions are generally sufficient, as the small deformations involved do not significantly alter the material structure. Usually, we are interested in the effective elasticity tensor $\mathbf{C}$, which, in the statistically homogenized version of the local elasticity tensor $\mathbf{c}$, can be written as follows:

$$\mathbf{C} = \langle \mathbf{c} \rangle - \mathbf{G} * \mathbf{F}, \tag{1}$$

with $\mathbf{G}$ denoting the Green's function tensor and tensor $\mathbf{F}$ having components of the form

$$f(\mathrm{r}' \in h_i | \mathrm{r} \in h_j), \tag{2}$$

denoting the probability of r' being in phase h_i under the condition of r being a point within phase h_j. The functions in 2 are called 2-point correlation functions.

3 DATA AVAILABILITY

In data-driven machine learning, the quality and quantity of training and test data sets affect the accuracy and stability of ML-method. Since data availability is scarce, the usage of synthetic data sets is a popular option. Trained on a high quality, big enough dataset, Neural Networks (NN) have high perfomance rates and accuracy. Here, the input data are images of isotropic two-phase microstructure and the outputs are corresponding 2-point correlation functions, which serve as spatial descriptors, quantifying the probability of finding material points in a specific phase. In 2-isotropic-phase composites, there are four 2-point functions describing the spatial correlation of material phases. A standard way to extract a correlation function descriptor from a microstructure image is the usage of the Monte Carlo method. Since the microstructures under consideration are isotropic, the correlation functions depend only on the distance between two points. Therefore, for a given distance r, one can randomly select two distinct points within the microstructure and count the occurrences of each of the four possible cases, yielding the probabilities $P_{ij}(\mathrm{r})$. In statistically homogeneous media, 2-point correlation functions depend only on the relative position of the two points, making them translationally invariant. To compute these functions, we randomly select 400 points within the microstructure and, for each, evaluate 200 evenly distributed distances in a counterclockwise direction. At each step, we determine whether the sampled point belongs to phase 1 or phase 2 and update the corresponding counters accordingly. This stochastic approach provides an estimation of the desired 2-point probability functions. This generates a statistical Monte Carlo process. For the case of 2- and 3-point probability functions, the process is illustrated in figure 3.

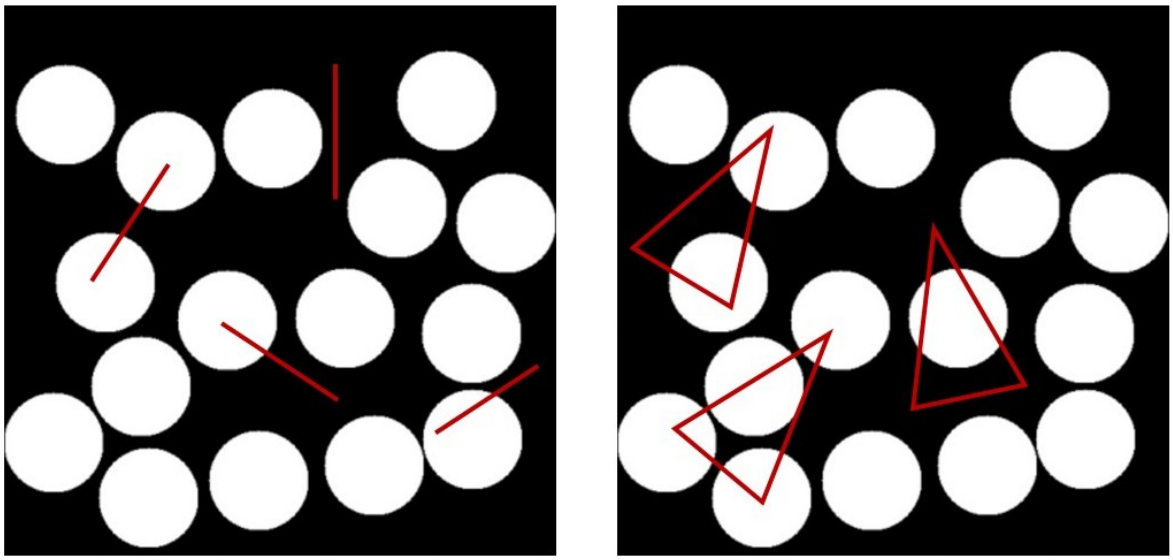


Figure 1: Illustration of the statistical process extracting 2- and 3-point correlation functions, respectively.

For 2-point statistics, the left hand side image shows the randomly placed distance troughout the structure in as red lines, whereas the equivalent process for 3-point statistics requires the traction of points which can be realized by considering triangular shapes and counting the occurence of points in phases at corners of the triangles. In order to generate enough data for the training of a network, we abstract the overall properties of 2D isotropic two-phase microstructures resulting in sqaured black and white images with one phase being black and the other phase being white. The desired material behavior is exhibited by circular inclusions, randomly distributed throughout the material matrix, making this a reasonable choice for synthetic data in ML tasks.

4 ML SURROGATE OF CORRELATION FUNCTIONS

The usage of NNs as a substitute of analytic correlation functions in statistical approaches is a promising research field since ML techniques proved to be computationally efficient for many engineering tasks. The four 2-point functions describing the spatial correlation of material phases arising in isotropic two-phase materials are usually extracted by a statistical process like the Monte Carlo method described above. However, this approach is computationally expensive. For this reason, we suggest a solution coupling Convolution Neural Networks (CNN) and Fully Connected Neural Networks (FCNN). Convolutional Neural Networks (CNNs) have emerged as one of the most powerful tools for processing grid-structured data, such as microstructure images. CNNs are a reliable tool in a wide range of applications, including image classification, object detection, style transfer (e.g., learning the artistic style of a painter and transforming another image to match that style), image reconstruction, and more. A CNN works by applying a variety of filters to the image and carrying out multiple convolution operations. Mathematically, the convolution operation applied to a 2D $N \times N$-image can be expressed as follows:

$$y(i,j) = f(\sum_{k,l} \mathbf{x}(k,l) \cdot \mathbf{w}(i-k, j-l) + b), \tag{3}$$

with $\mathbf{x}$ the input image, $\mathbf{y}$ the output feature map, $\mathbf{w}$ the filter or kernel, and, as above, b the bias and f the activation function. By performing these operations repeatedly, the network condenses the pixel-level details into a set of higher-level, abstracted representations. This not only reduces data complexity but also highlights the most critical features necessary for subsequent analysis or prediction tasks. Fully connected neural networks (FCNNs) form another fundamental component of deep learning. Once trained on a sufficiently large and diverse dataset, an FCNN becomes a highly efficient and data-driven predictor, capable of rapidly estimating the desired properties of previously unseen data. We suggest a neural network architecture designed to take a microstructure image as input and produce the corresponding 2-point correlation function as output as shown in figure 2. This architecture integrates convolutional and fully connected layers, effectively addressing the unique characteristics of image data and functional data.

Here, unlike traditional tasks where a single function is learned, we deal with an entire class of functions, each exhibiting unique characteristics. Some of these functions are strictly monotonic, while others display non-monotonic behavior with local minima and maxima, which is a key challenge in the determination of an appropriate surrogate. We overcome this problem by combining a CNN and FCNN to simultaneously process both image and functional data. Finally, once both networks have processed their respective inputs, the outputs are merged to create a unified representation. Employing this multi-path approach, the network is capable of

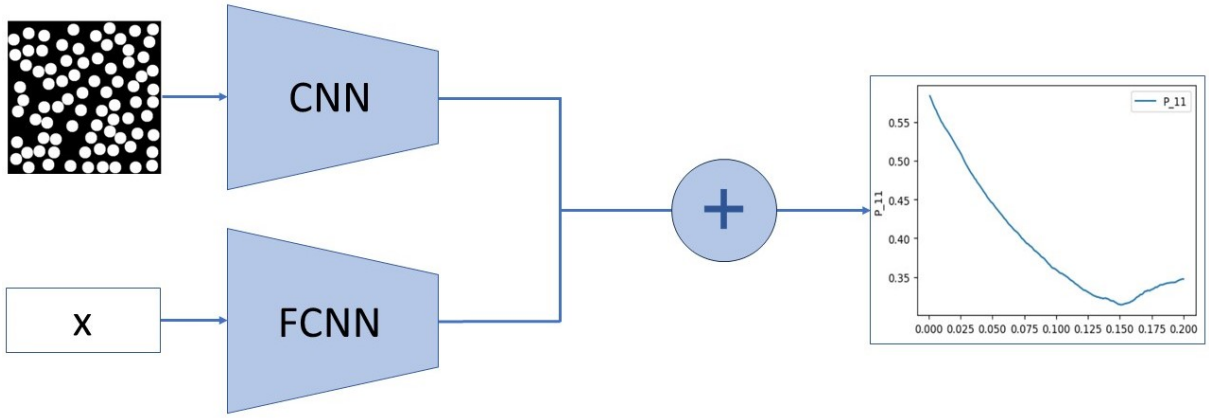


Figure 2: Neural network architecture.

effectively learning a diverse class of functions while maintaining computational efficiency and required accuracy.

The present study considers a range of composites with the inclusion volume fractions of 35-45%. However, the approach is easily adaptable to different ranges of inclusions via extensions in the training data set. Illustrative examples dealing with different microstructures and corresponding correlation functions are shown in figures 3 and 4. In figure 3, the left microstructure has 60% circular inclusions smaller in size compared to the 57% circular inclusions in the right hand side microstructure image. Due to the bigger size, there are fewer inclusions in the second one. The effect, inclusion sizes have on the monotonocity of the corresponding correlation function, is shown in figure 4. Smaller more scattered second phase inclusion result in a prominent minimum, followd by a local maximum, before the funtion converges to a limit. Our most recent investigation makes use of neural operators to take into account the infinite dimensional nature of the output space as an alternative of CNN-FCNN combination.

5 CONCLUSION AND OUTLOOK

In this work, we successfully developed a neural network architecture capable of predicting 2-point correlation functions for isotropic two-phase materials, specifically for cases with larger inclusion sizes. We achieved accurate predictions even with a relatively small training dataset, demonstrating the network's ability to generalize effectively. Despite these promising results, certain challenges remain. One limitation remaining is the network's performance on micrographs containing smaller inclusions, where the model may struggle to capture finer structural details. Additionally, expanding the variety of microstrcutres, e. g. in terms of the volume fraction range, the network can accurately predict is planned for future work.

To address these challenges, we follow multiple strategies. One approach involves diversifying the training dataset by incorporating a broader range of inclusion sizes and volume fractions, ensuring that the model is exposed to a more comprehensive set of material structures. Another aspect of current work is refining the network architecture itself, by introducing modifications that enhance its ability to capture finer details or improve its robustness across different material configurations. Both of these strategies are currently worked on by the authors.

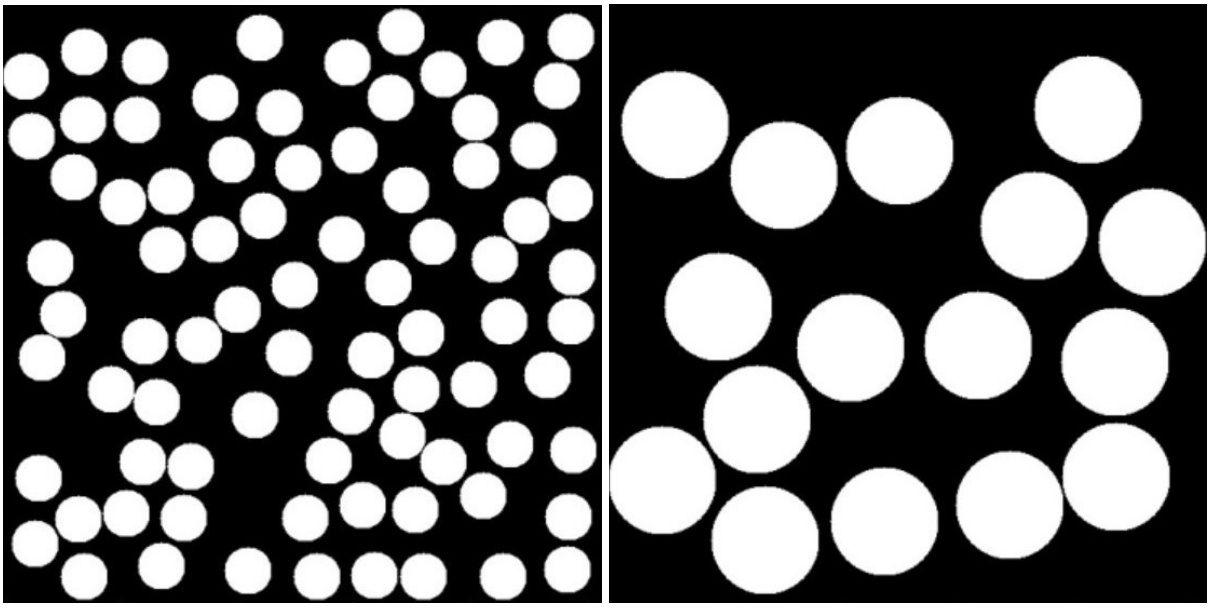


Figure 3: Microstructure image examples.

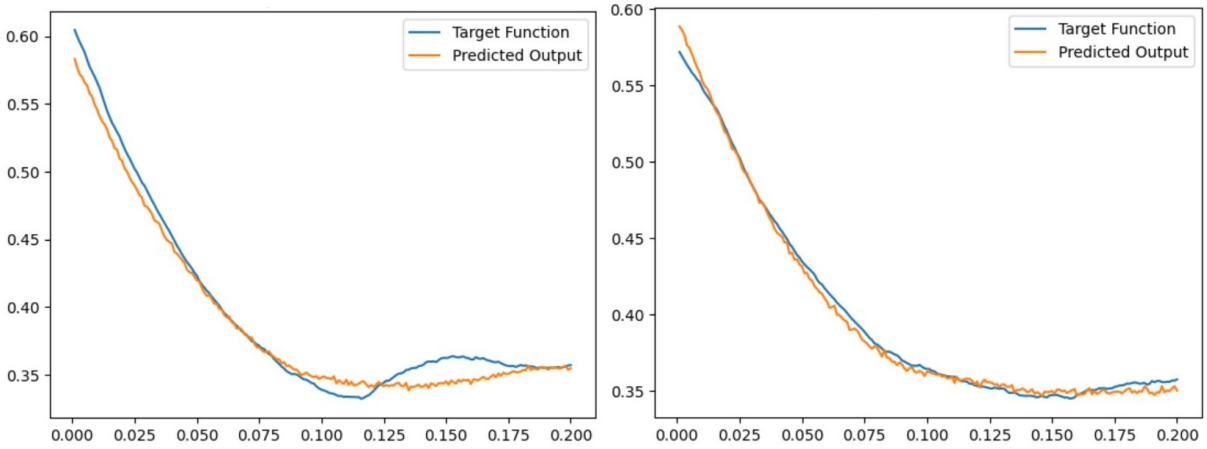


Figure 4: 2-point correlation functions corresponding to the microstructures shown in figure 3.

REFERENCES

[1] P. Corson, Correlation functions for predicting properties of heterogeneous materials. II. Empirical construction of spatial correlation functions for two-phase solids. *Journal of Applied Physics*, **45(7)**, 3165–3170, 1974.

[2] P. Gavallas, G. Stefanou, D. Savvas, C. Mattrand, J-M.Bourinet, CNN-based prediction of microstructure-derived random property fields of composite materials. *Computer Methods in Applied Mechanics and Engineering*, **490**, 117–207, 2024.

[3] S. Hildebrand, Beissel J. G., M. Mohammadkhah, and S. Klinge, Coupled CANN-DEM Simulation in Solid Mechanics. *Machine Learning: Science and Technology*, accepted for the publication, 2024.

[4] S. Hildebrand, S. Klinge, Comparison of Neural FEM and Neural Operator Methods for applications in Solid Mechanics. *Neural Computing and Applications*, **36**, 16657–16682, 2024.

[5] S. Hildebrand, S. Klinge, Hybrid data-driven and physics-informed regularized learning of cyclic plasticity with neural networks. *Machine Learning: Science and Technology*, **5(4)**, 045–058, 2024.

[6] S. Klinge, Inverse Analysis for Multiphase Nonlinear Composites with Random Microstructure. *Int. J. Multiscale Comp. Eng.*, **10(4)**, 361–373, 2012.

[7] S. Klinge, K. Hackl, Application of the Multiscale FEM to the Modeling of Nonlinear Composites with a Random Microstructure. *Int. J. Multiscale Comp. Eng.*, **10(3)**, 213–227, 2012.

[8] S. Lin, H. Garmestani, Statistical continuum mechanics analysis of an elastic two-isotropic-phase composite material. *Composites Part B: Engineering*, **31(1)**, 39–46, 2000.

[9] J. Pejic, M. Petkovic, S. Klinge, Exploring spatial reasoning performances of CNN on linear layout dataset. *Machine Learning: Science and Technology*, **5(4)**, 045–056, 2024.

[10] S. Torquato, Effective stiffness tensor of composite media—I. Exact series expansions. *Journal of the Mechanics and Physics of Solids*, **45**, 1421–1448, 1997.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

STOCHASTIC FINITE ELEMENT ANALYSIS OF CONCRETE BEAMS

Leontios Kotropoulos*, George Stefanou

Department of Civil Engineering, Aristotle University of Thessaloniki
University Campus, 54124 Thessaloniki, Greece
leoskotrop@gmail.com, gstefanou@civil.auth.gr

Abstract

In recent years, the rapid advancement of computers and their capabilities has enabled the solution of complex realistic problems using numerical methods, such as the finite element method (FEM). The classical deterministic FEM ignores the inherent uncertainty in the problem's parameters. This unutilized information is taken into account in the stochastic finite element method (SFEM), which is an extension of the classical method, incorporating stochastic analysis to handle uncertainties. In the framework of SFEM, the finite elements are characterized by random properties described by random variables. In this way, a more refined and accurate model of the problem is created, although this leads to an increase in computational cost.

The aim of the present work is the application of SFEM for the analysis of two concrete structures and the extraction of useful conclusions that can lead to safer and more cost-effective designs, as well as to the improvement of safety factors in construction codes. To this end, the mesoscale random fields of the elasticity tensor of concrete are determined from the reconstructed microstructure of a cubic specimen. Additionally, their statistical characteristics are computed and sample functions are generated using the spectral representation method and translation field theory, in order to examine the response variability of the above structures and to understand the effect of the random microstructure on their macroscopic behavior.

Keywords: Concrete, Mesoscale random fields, Response variability.

1 INTRODUCTION

Concrete is by far the dominant material in construction. It is estimated that the current global production of concrete amounts to approximately 11 billion tons per year, making it the most widely produced material in the world today. No other material, apart from water, is consumed by humans in such astonishing quantities. Additionally, concrete is the primary material that determines the economic development of a region, as most infrastructure projects are built with it. However, its highly heterogeneous and complex microstructure makes it very difficult to reliably predict its behavior, since theoretical models and analytical methods in mechanics of materials are insufficient when it comes to concrete structures [1].

In this work, the response variability of two concrete structures is computed using stochastic finite elements based on mesoscale random fields of the elasticity tensor, which are derived from the reconstructed microstructure of a cubic specimen [2]. To achieve this, the statistical characteristics of these fields are evaluated [3] and sample functions are generated using the spectral representation method in conjunction with translation field theory [4,5]. The objective is to assess the effect of the random microstructure on the macroscopic behavior of the above structures, with the ultimate goal of extracting useful conclusions that can lead to safer and more cost-effective designs, as well as to the improvement of safety factors in construction codes.

The paper is organized as follows: In Section 2, the methodology followed for determining the mesoscale random fields of the elasticity tensor of concrete from the reconstructed microstructure of a cubic specimen is analyzed. Section 3 focuses on the generation of sample functions and the examination of the response variability of two concrete structures. The key conclusions derived are presented in Section 4.

2 RANDOM FIELDS OF THE ELASTIC PROPERTIES OF CONCRETE

In this section, the methodology followed in the recent publication by Stefanou et al. [2] for determining the spatial variation of the elastic properties of concrete at various scales is presented, taking into account its highly heterogeneous and complex microstructure. From the computed tomography of a cubic concrete specimen, its microstructure is reconstructed and the mesoscale random fields of the elasticity tensor are computed by applying the moving window method in conjunction with the computational homogenization process. The analysis of the statistical characteristics of the mesoscale random fields enables the investigation of the effect of the random microstructure on the macroscopic behavior of concrete.

2.1 Computed tomography

Computed tomography is an imaging method for visualizing the internal structure of an object using X-rays. More specifically, a radiation source rotates around the object and its attenuation is measured by a series of detectors positioned diametrically opposite, as shown in Figure 1. Upon completing a full rotation, automated algorithms generate a two-dimensional cross-sectional image of the object, known as a tomographic image. When a sufficient number of consecutive tomographic images of the object are collected, they can be combined to reconstruct its three-dimensional internal structure.

Each pixel of the tomographic image represents the measured attenuation value of the radiation, which is converted through a linear transformation into the corresponding HU value on the Hounsfield scale. The Hounsfield scale is a system for quantifying the density of an object, where distilled water under standard pressure and temperature corresponds to $0\ \mathrm{HU}$, while air under the same conditions corresponds to $-1000\ \mathrm{HU}$.

Although computed tomography was initially developed and applied for medical purposes, the gradual increase in its availability and reduction in cost today have expanded its use to other fields, such as the study of the microstructure of heterogeneous materials.

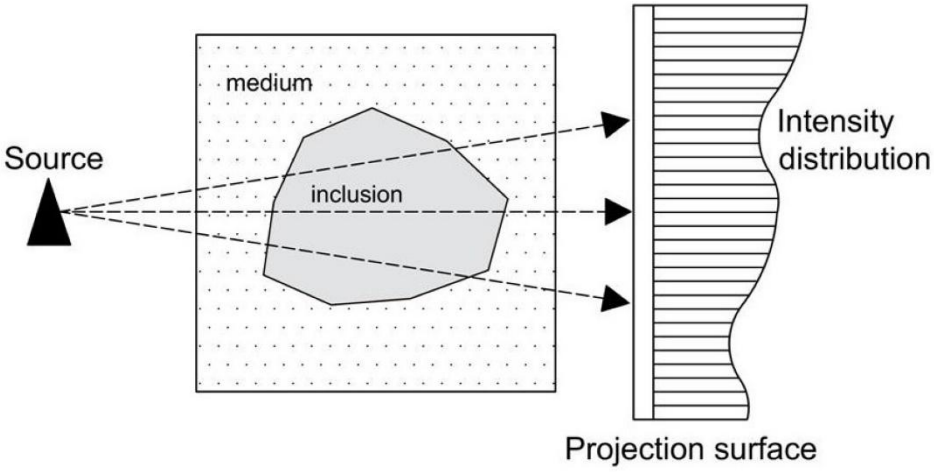


Figure 1: Source-detector pair and radiation attenuation.

2.2 Computational homogenization

Computational homogenization is a process used to replace a heterogeneous material with an equivalent homogeneous one by determining its effective properties. This approach offers an efficient alternative for analyzing heterogeneous materials, as the computational cost is significantly reduced compared to directly simulating the microstructure at the macroscale. The effective properties describe the macroscopic behavior of the heterogeneous material and are linked to its microstructural characteristics through Hill's energy condition, which is also referred to as Hill's macro-homogeneity condition by some researchers.

2.2.1 Representative and statistical volume element

In the context of computational homogenization, a representative volume element (RVE) is considered, which sufficiently captures the microstructure of a heterogeneous material. The RVE must be large enough to incorporate all essential microstructural characteristics, while remaining small enough to allow efficient computational handling.

According to Hashin [6], the determination of the RVE is based on the assumption of statistical homogeneity, which states that the statistical characteristics of a heterogeneous material are similar throughout its extent. In the case of heterogeneous materials with a periodic or nearly periodic microstructure, the RVE can be explicitly defined, whereas for heterogeneous materials with a random microstructure, the RVE is determined approximately.

According to Hill [7], the existence of the RVE requires a separation of scales in the form:

$$d \ll L \ll L_{macro} \tag{1}$$

where d the characteristic dimension of the microstructure, such as the average size of the aggregates, L the characteristic dimension of the RVE at the mesoscale and L_{macro} the characteristic dimension of the problem at the macroscale. Figure 2 depicts an illustration of scale separation.

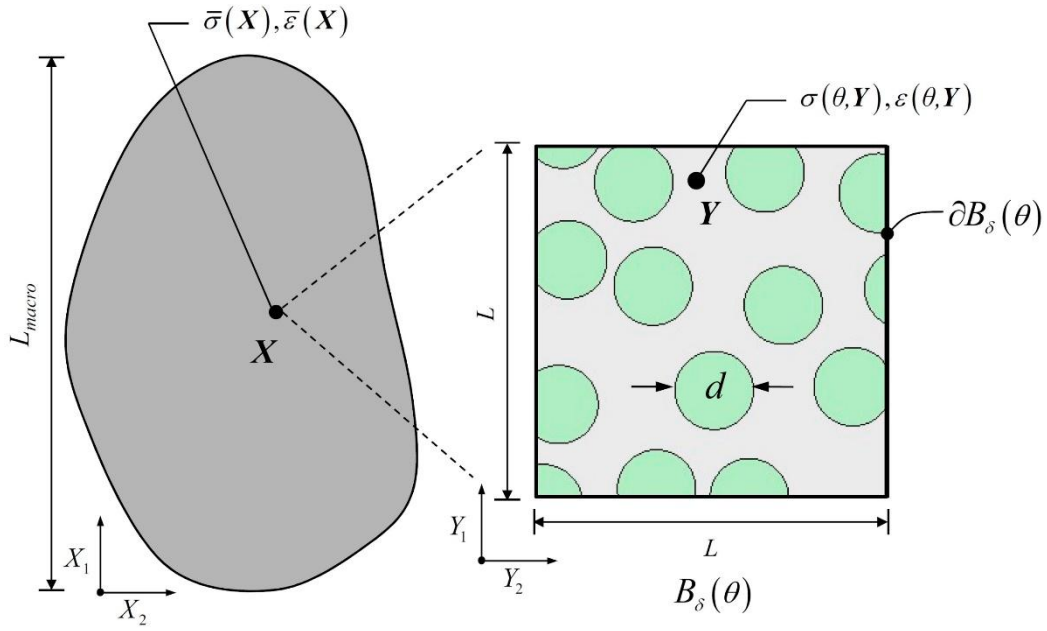


Figure 2: Illustration of scale separation.

The RVE is characterized by constant and deterministic properties. To account for the spatial variation of properties at the mesoscale, the concept of the statistical volume element (SVE) is introduced [8,9]. The SVE is always smaller than the corresponding RVE and is described by the non-dimensional parameter $\delta = L/d$ with $\delta \in [1,\infty]$, which is called the scale factor. The properties derived from an SVE are referred to as apparent properties. When $\delta \to \infty$, the SVE approaches the corresponding RVE and the apparent properties converge to the effective properties.

2.2.2 Computation of the apparent elasticity tensor

Let a random mesoscale realization of the microstructure of a heterogeneous material $B_\delta(\theta)$, which is defined in a coordinate system Y. The stress and strain fields, expressed as a superposition of means ($\bar{\sigma}$ and $\bar{\varepsilon}$) and zero-mean fluctuations (σ' and ε'), are written as:

$$\sigma(\theta,Y) = \bar{\sigma} + \sigma'(\theta,Y) \qquad \varepsilon(\theta,Y) = \bar{\varepsilon} + \varepsilon'(\theta,Y) \tag{2}$$

The means of stress and strain tensors at some point X of the macroscale are computed as:

$$\bar{\sigma}(X) = \frac{1}{V_\delta}\int_{B_\delta(\theta)} \sigma(\theta,Y)\,dV_\delta \qquad \bar{\varepsilon}(X) = \frac{1}{V_\delta}\int_{B_\delta(\theta)} \varepsilon(\theta,Y)\,dV_\delta \tag{3}$$

The volume average of the strain energy is given by:

$$\bar{U} = \frac{1}{2V_\delta}\int_{B_\delta(\theta)} \sigma(\theta,Y):\varepsilon(\theta,Y)\,dV_\delta = \frac{1}{2}\overline{\sigma:\varepsilon} = \frac{1}{2}\bar{\sigma}:\bar{\varepsilon} + \frac{1}{2}\overline{\sigma':\varepsilon'} \tag{4}$$

where the notation $\sigma:\varepsilon$ represents $\sigma_{ij}\varepsilon_{ij}$. For $\delta \to \infty$ the term $\overline{\sigma':\varepsilon'}$ becomes negligible and it follows that:

$$\overline{\sigma:\varepsilon} = \bar{\sigma}:\bar{\varepsilon} \tag{5}$$

Equation (5) is known as Hill's energy condition and is also valid when the following constraint is satisfied:

$$\int_{\partial B_\delta} \left(t-\bar{\sigma}n\right)\left(u-\bar{\varepsilon}Y\right)dS = 0 \tag{6}$$

where t the traction vector and u the displacement vector. Equation (6) is a priori satisfied by the following boundary conditions:

- Uniform strains (otherwise known as kinematic, essential or Dirichlet)
- Uniform stresses (otherwise known as static, natural or Neumann)
- Uniform orthogonal-mixed
- Periodic

The elasticity tensor, being a 4th order tensor, contains $3^4 = 81$ components, which are reduced to 21 independent components due to symmetry. The matrix form of the elasticity tensor is given by:

$$C = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} & C_{36} \\ C_{41} & C_{42} & C_{43} & C_{44} & C_{45} & C_{46} \\ C_{51} & C_{52} & C_{53} & C_{54} & C_{55} & C_{56} \\ C_{61} & C_{62} & C_{63} & C_{64} & C_{65} & C_{66} \end{bmatrix} \tag{7}$$

In order to determine the apparent elasticity tensor, it is necessary to impose a prescribed uniform strain tensor $\bar{\varepsilon} = \begin{bmatrix} \bar{\varepsilon}_1 & \bar{\varepsilon}_2 & \bar{\varepsilon}_3 & \bar{\varepsilon}_4 & \bar{\varepsilon}_5 & \bar{\varepsilon}_6 \end{bmatrix}^T$ on the boundary nodes of the discretized microstructure model through the following uniform displacements:

$$u_b = D_b^T \bar{\varepsilon} \tag{8}$$

where D_b a geometric matrix which depends on the coordinates Y_1, Y_2, Y_3 of the boundary node b and is defined as:

$$D_b = \frac{1}{2}\begin{bmatrix} 2Y_1 & 0 & 0 \\ 0 & 2Y_2 & 0 \\ 0 & 0 & 2Y_3 \\ Y_2 & Y_1 & 0 \\ 0 & Y_3 & Y_2 \\ Y_3 & 0 & Y_1 \end{bmatrix} \tag{9}$$

The apparent elasticity tensor is now computed as:

$$C_\delta\left(\theta\right) = \frac{1}{V_\delta} D\tilde{K}_{bb}D^T \tag{10}$$

where $\tilde{K}_{bb} = K_{bb} - K_{bi}K_{ii}^{-1}K_{ib}$ the condensed stiffness matrix resulting from the separation of the internal nodes i and the boundary nodes b and $D = \begin{bmatrix} D_1 & D_2 & ... \end{bmatrix}$ a matrix containing the geometric matrices D_b of the boundary nodes b [10].

2.3 Determination of mesoscale random fields

The determination of the mesoscale random fields of the elasticity tensor of concrete is achieved through the application of the moving window method to a cubic specimen, whose microstructure has been reconstructed based on a set of tomographic images.

2.3.1 Reconstruction of concrete microstructure

The cubic concrete specimen used had dimensions of approximately $136.724 \times 136.724 \times 131.25\ \text{mm}^3$. From its computed tomography with a constant slice thickness of $0.625\ \text{mm}$, 211 tomographic images were obtained with a resolution of 281×281 pixels, meaning a pixel spacing of $0.4883 \times 0.4883\ \text{mm}^2$. Figure 3 depicts a tomographic image of the cubic concrete specimen.

These data are utilized for the reconstruction of its microstructure, which is divided into a set of SVEs, as shown in Figure 4, by applying the moving window method. Each SVE is discretized with hexahedral elements corresponding to the voxels of the reconstructed microstructure. In order to reduce the computational cost, the hexahedral elements consist of a grid of $2 \times 2 \times 2$ voxels. At the integration points of the hexahedral elements, the appropriate material is assigned depending on the HU values of these points, which are computed based on the corresponding nodal quantities through the shape functions.

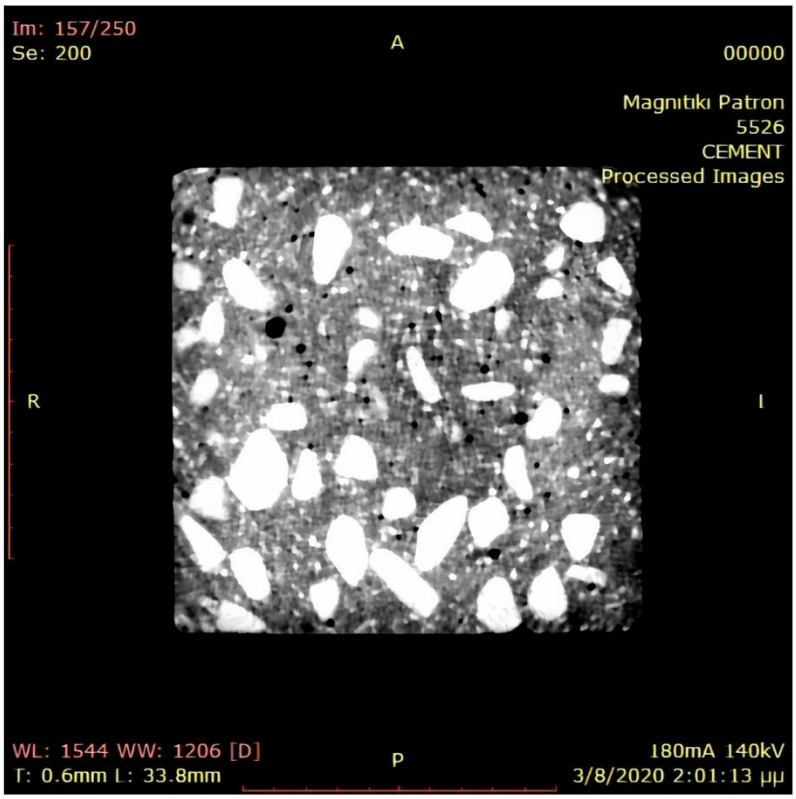


Figure 3: Tomographic image of the cubic concrete specimen.

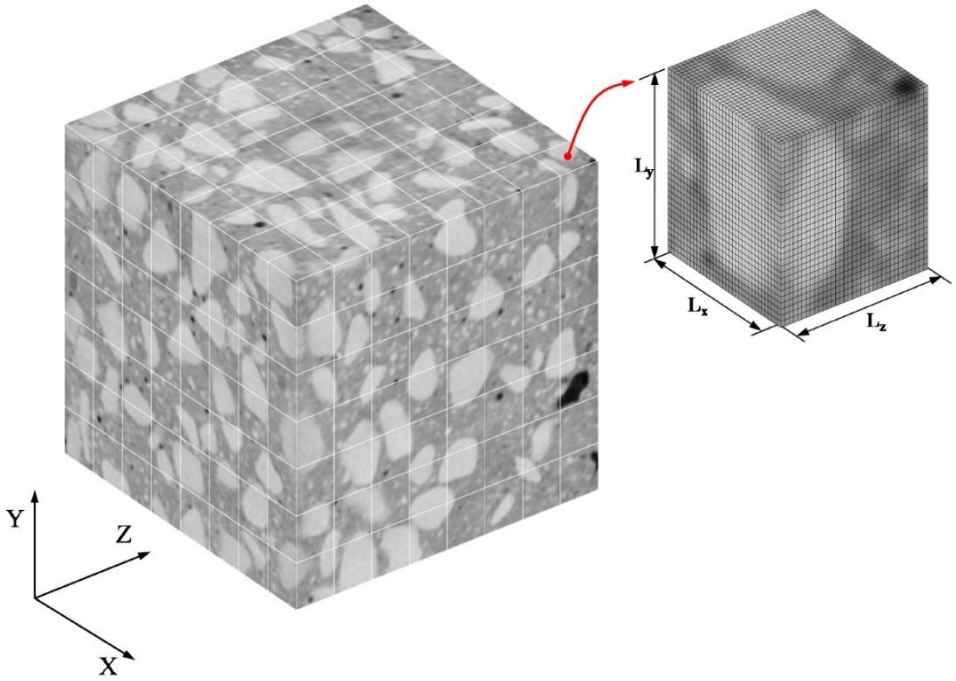


Figure 4: Reconstructed microstructure of the cubic concrete specimen, which has been divided into a set of SVEs.

The distinction of concrete constituents lies in defining representative HU value ranges for each of them. More specifically, for the pores, cement and aggregates the ranges $[-960, 800]$, $[800, 2000]$ and $[2000, 2974]$ are selected respectively, as shown in Figure 5. Additionally, for the cement the modulus of elasticity and Poisson's ratio are considered to be 20 GPa and 0.2 respectively, while for the aggregates the same properties are considered to be 100 GPa and 0.2 respectively. The pores are modeled as an almost incompressible material with a modulus of elasticity equal to 1% of that of cement, meaning a modulus of elasticity and Poisson's ratio of 0.2 GPa and 0.45 respectively.

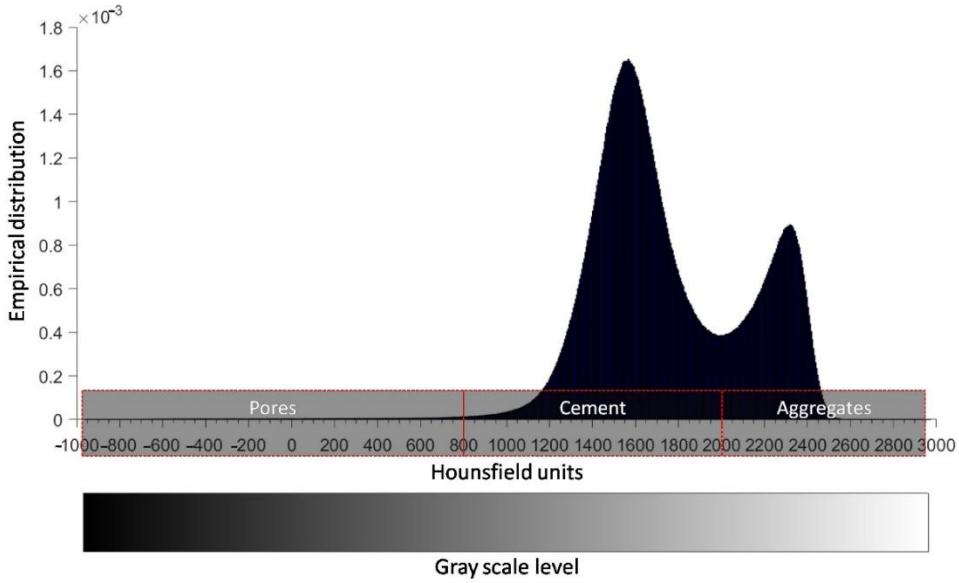


Figure 5: Empirical distribution of HU values.

2.3.2 Moving window method

In the moving window method, an initial window of dimensions $L_x \times L_y \times L_z$ is shifted with a fixed step $\Delta\xi$ in all three directions, fully covering the microstructure. In this way, a set of windows is extracted, each containing a small portion of the microstructure. For every window, the apparent properties are computed through the procedure described in Section 2.2.2. These properties are then combined with the positions of the window centers to form the corresponding random fields. Figure 6 depicts an illustration of the moving window method.

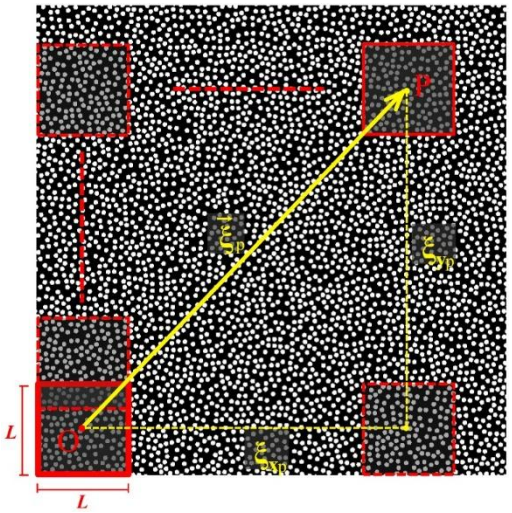


Figure 6: Illustration of the moving window method.

From the random fields, histograms of their values can be obtained and their spatial correlation functions can be determined for every lag $\left(\xi_x, \xi_y, \xi_z\right)$ as:

$$\rho_A^B\left(\xi_x,\xi_y,\xi_z\right)=\frac{1}{n_{wx}n_{wy}n_{wz}-1}\sum_{i=0}^{n_{wx}}\sum_{j=0}^{n_{wy}}\sum_{k=0}^{n_{wz}} \left(\frac{A\left(x_i,y_j,z_k\right)-\mu_A}{\sigma_A}\right)\left(\frac{B\left(x_i+\xi_x,y_j+\xi_y,z_k+\xi_z\right)-\mu_B}{\sigma_B}\right) \tag{11}$$

$$-n_{wx}\Delta\xi \le \xi_x \le n_{wx}\Delta\xi, -n_{wy}\Delta\xi \le \xi_y \le n_{wy}\Delta\xi, -n_{wz}\Delta\xi \le \xi_z \le n_{wz}\Delta\xi$$

where μ_A, μ_B the mean values and σ_A, σ_B the standard deviations of quantities A, B respectively. When $A \equiv B$, ρ_A^B denotes auto-correlations, otherwise cross-correlations are defined.

It is noted that the moving window step was selected equal to its size to avoid extracting overlapping windows. The examined mesoscale sizes, the dimensions of the moving window and the total number of SVEs are provided in Table 1.

δ_i	L_x x L_y x L_z (mm^3)	No. SVEs
δ_1	4.88 x 4.88 x 6.25	16464
δ_2	9.77 x 9.77 x 8.75	2940
δ_3	19.53 x 19.53 x 18.75	343
δ_4	27.34 x 27.34 x 26.25	125

Table 1: Examined mesoscale sizes, dimensions of the moving window and total number of SVEs.

2.3.3 Results

In this work, the mean xy plane of the three-dimensional concrete specimen is considered and the two-dimensional random fields of the elasticity components C_{11} and C_{44} are computed for all examined mesoscale sizes. These fields are depicted in Figures 7 and 8. The elasticity components C_{11} and C_{44} represent the axial and shear stiffness respectively and are measured in GPa. Lengths are measured in mm.

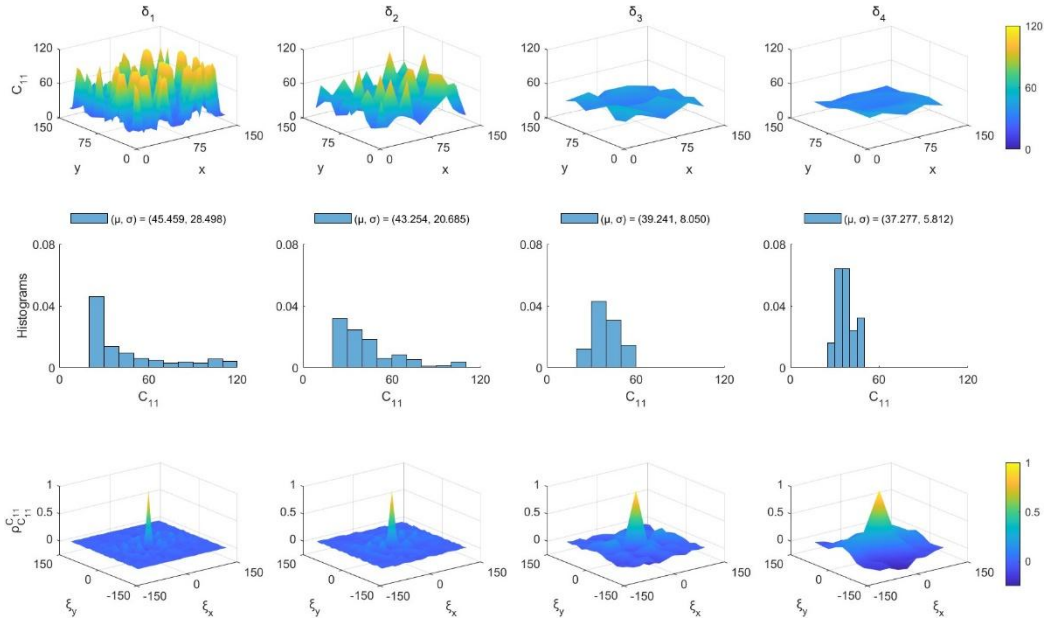


Figure 7: Random fields of the elasticity component C_{11}.

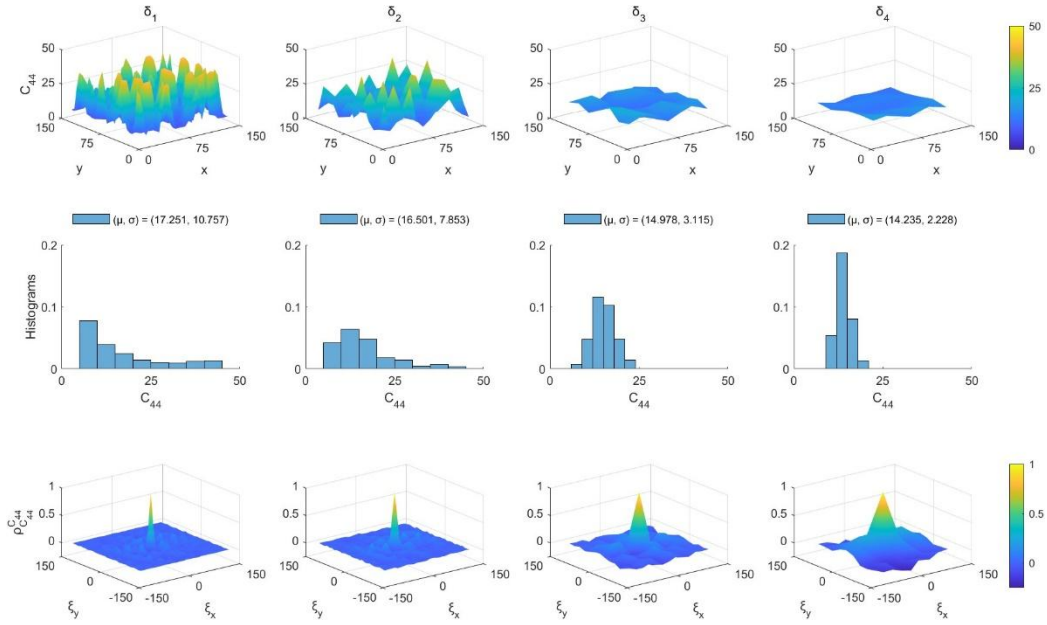


Figure 8: Random fields of the elasticity component C_{44}.

Figure 9 depicts the cross-correlations between the elasticity components C_{11} and C_{44} for lag values along x (ξ_x, $\xi_y = 0$) and y ($\xi_x = 0$, ξ_y), while Figure 10 depicts their scatter plots.

From Figure 10, it follows that the elasticity components C_{11} and C_{44} exhibit a strong correlation with each other, as expected.

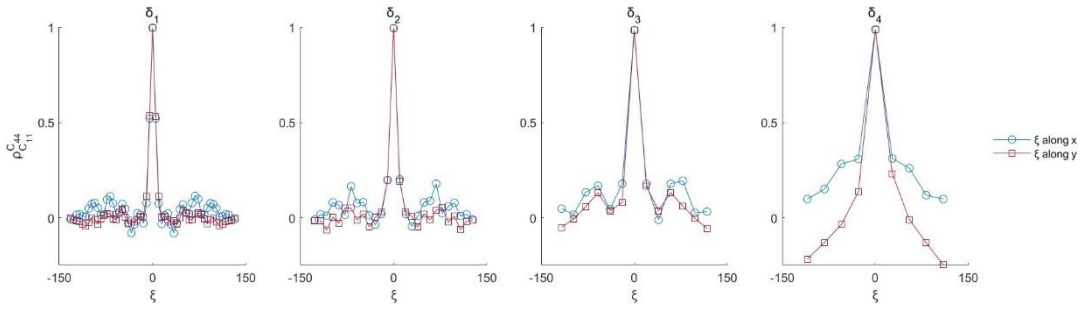


Figure 9: Cross-correlations between the elasticity components C_{11} and C_{44} for lag values along x (ξ_x, $\xi_y = 0$) and y ($\xi_x = 0$, ξ_y).

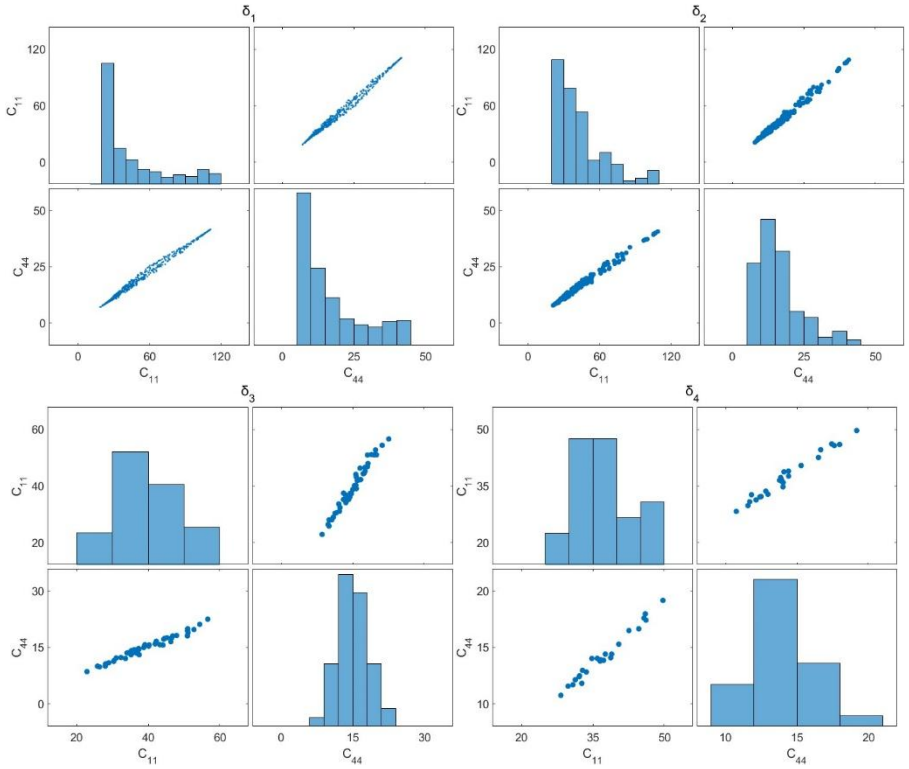


Figure 10: Scatter plots of the elasticity components C_{11} and C_{44}.

The correlation lengths of the random fields of the elasticity components C_{11} and C_{44} are obtained by fitting a predefined correlation model to their auto-correlations, as shown in Figures 11 and 12. In this work, the squared exponential model is used, as it represents a realistic choice for concrete and is given by:

$$R_{ff}\left(\xi_x, \xi_y\right) = \sigma^2 \exp\left[-\left(\frac{\xi_x}{b_x}\right)^2 - \left(\frac{\xi_y}{b_y}\right)^2\right] \tag{12}$$

where b_x, b_y the parameters which are proportional to the correlation lengths of the random fields along the x, y axes respectively.

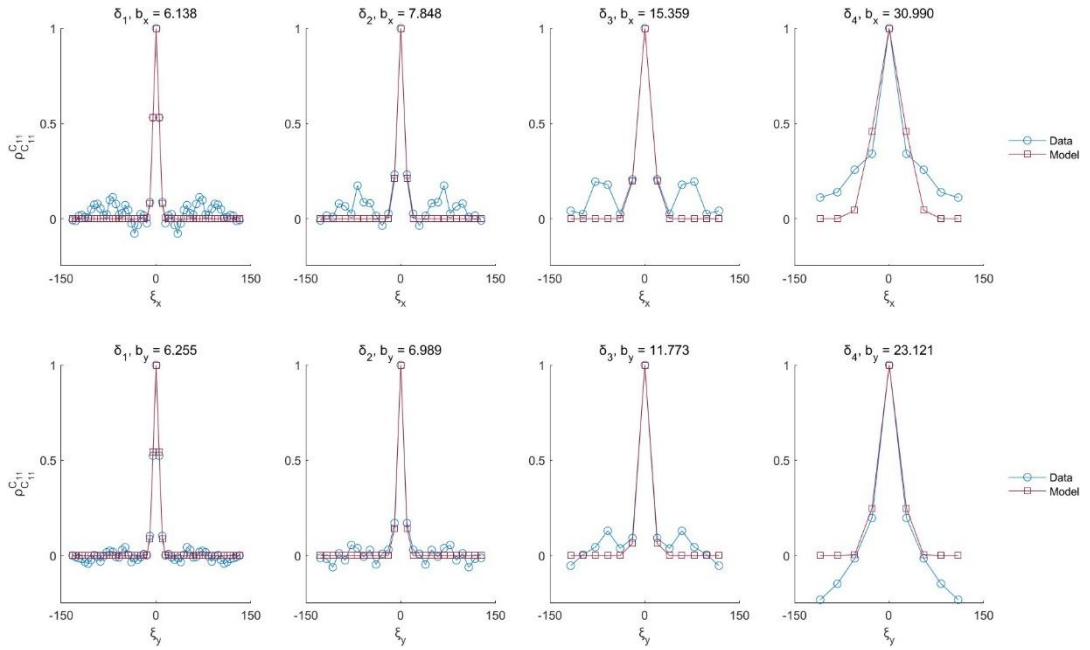


Figure 11: Fitting of the squared exponential model to the auto-correlations of the elasticity component C_{11}.

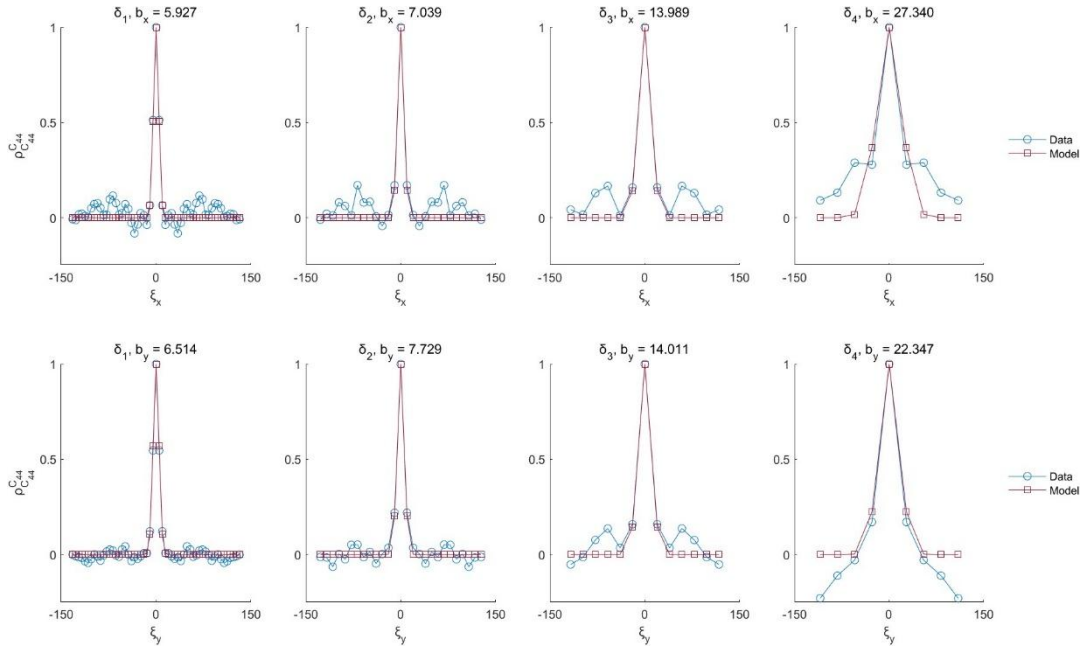


Figure 12: Fitting of the squared exponential model to the auto-correlations of the elasticity component C_{44}.

Additionally, the fitting of both the normal and the lognormal distribution to the histograms of the elasticity components C_{11} and C_{44} is examined, as shown in Figures 13 and 14. It is observed that the lognormal distribution provides a satisfactory approximation of these histograms and therefore non-Gaussian fields are required for the simulation of the random fields of the elasticity components C_{11} and C_{44}, particularly at smaller scales. Another reason for preferring the lognormal distribution is that it eliminates the possibility of negative values for the elasticity components C_{11} and C_{44}.

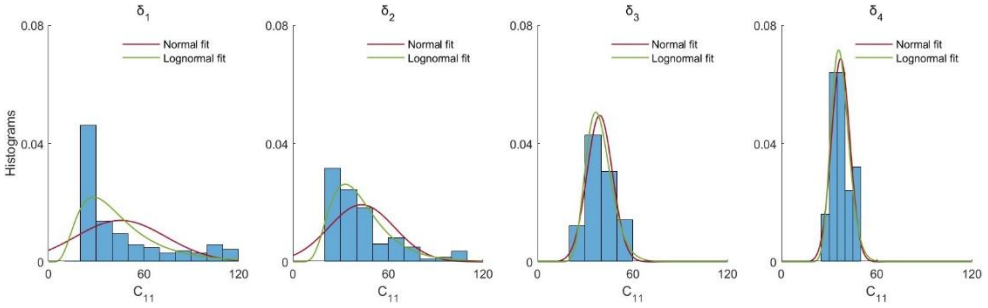


Figure 13: Fitting of both the normal and the lognormal distribution to the histograms of the elasticity component C_{11}.

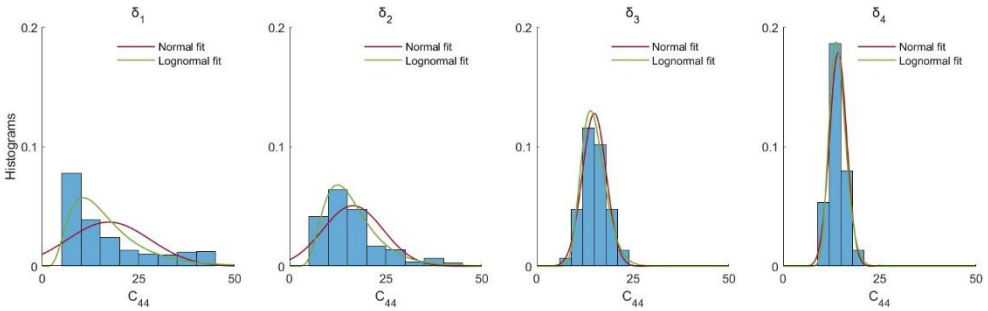


Figure 14: Fitting of both the normal and the lognormal distribution to the histograms of the elasticity component C_{44}.

3 RESPONSE VARIABILITY OF CONCRETE STRUCTURES

This section focuses on generating sample functions of the random fields of the elasticity tensor of concrete by applying the spectral representation method in conjunction with the mathematical theory of translation fields. This method is well suited in the context of Monte Carlo simulation, which is used for computing the response variability of two concrete structures [11]. Useful conclusions are derived regarding the effect of the random microstructure on the macroscopic behavior of concrete.

3.1 Spectral representation method

The i-th sample function of a two-dimensional homogeneous Gaussian field is given by [4]:

$$f^{(i)}(x,y)=\sqrt{2}\sum_{n_x=0}^{N_x-1}\sum_{n_y=0}^{N_y-1}\Big[A^{(1)}_{n_xn_y}\cos\left(\kappa_{xn_x}x+\kappa_{yn_y}y+\varphi^{(1)(i)}_{n_xn_y}\right)+A^{(2)}_{n_xn_y}\cos\left(\kappa_{xn_x}x-\kappa_{yn_y}y+\varphi^{(2)(i)}_{n_xn_y}\right)\Big] \tag{13}$$

where $\varphi^{(1)(i)}_{n_xn_y},\varphi^{(2)(i)}_{n_xn_y}$ the i-th sample of the independent random phase angles uniformly distributed in the range $[0,2\pi]$. The coefficients $A^{(1)}_{n_xn_y},A^{(2)}_{n_xn_y}$ are written as:

$$A^{(1)}_{n_xn_y}=\sqrt{2S_{ff}\left(\kappa_{xn_x},\kappa_{yn_y}\right)\Delta\kappa_x\Delta\kappa_y} \tag{14}$$

$$A^{(2)}_{n_xn_y}=\sqrt{2S_{ff}\left(\kappa_{xn_x},-\kappa_{yn_y}\right)\Delta\kappa_x\Delta\kappa_y} \tag{15}$$

where

$$\kappa_{xn_x} = n_x \Delta\kappa_x \qquad \kappa_{yn_y} = n_y \Delta\kappa_y \tag{16}$$

$$\Delta\kappa_x = \frac{\kappa_{xu}}{N_x} \qquad \Delta\kappa_y = \frac{\kappa_{yu}}{N_y} \tag{17}$$

$$n_x = 0,1,...,N_x - 1 \qquad n_y = 0,1,...,N_y - 1 \tag{18}$$

and

$$A_{0n_y}^{(1)} = A_{n_x 0}^{(1)} = 0 \tag{19}$$

$$A_{0n_y}^{(2)} = A_{n_x 0}^{(2)} = 0 \tag{20}$$

or

$$S_{ff}\left(0,\kappa_y\right) = S_{ff}\left(\kappa_x,0\right) = 0 \tag{21}$$

The terms N_x, N_y represent the number of intervals in which the wave number axes are subdivided, while the terms κ_{xu}, κ_{yu} represent the cut-off wave numbers beyond which the power spectral density function of the random field is assumed to be zero. The cut-off wave numbers are computed as:

$$\int_0^{\kappa_{xu}} \int_{-\kappa_{yu}}^{\kappa_{yu}} S_{ff}\left(\kappa_x,\kappa_y\right) d\kappa_x d\kappa_y = \left(1-\varepsilon\right) \int_0^{\infty} \int_{-\infty}^{\infty} S_{ff}\left(\kappa_x,\kappa_y\right) d\kappa_x d\kappa_y \tag{22}$$

where ε an arbitrarily small positive number, such as $\varepsilon = 0.001$.

The fulfillment of Equations (19)-(21) is necessary to ensure that the mean value and auto-correlations of any sample function are identical to those of the original field. Additionally, to avoid the aliasing effect, the space increments Δx and Δy, separating the generated values of any sample function along the x, y axes respectively, must satisfy the conditions:

$$\Delta x \le \frac{\pi}{\kappa_{xu}} \qquad \Delta y \le \frac{\pi}{\kappa_{yu}} \tag{23}$$

The power spectral density function corresponding to the squared exponential model is given by:

$$S_{ff}\left(\kappa_x,\kappa_y\right) = \sigma^2 \frac{b_x b_y}{4\pi} \exp\left[-\left(\frac{b_x \kappa_x}{2} \right)^2 - \left(\frac{b_y \kappa_y}{2} \right)^2 \right] \tag{24}$$

It is noted that the same set of phase angles was used for generating the sample functions of the elasticity components C_{11} and C_{44}. This is in accordance with the observation that the elasticity components C_{11} and C_{44} exhibit a strong correlation with each other. The terms N_x, N_y were selected equal to 20.

3.2 Translation fields

The generation of sample functions of a non-Gaussian field is based on a simple transformation of some underlying Gaussian field. If $g(x,y)$ a homogeneous zero-mean Gaussian field with unit variance, then a homogeneous non-Gaussian field $f(x,y)$ is defined as:

$$f(x,y) = F^{-1}\Phi\left[g(x,y)\right] \tag{25}$$

where Φ the standard Gaussian cumulative distribution function and F the non-Gaussian marginal cumulative distribution function of $f(x,y)$. The transformation $F^{-1}\Phi$ is a memory-less translation, since the value of $f(x,y)$ at an arbitrary point (x,y) depends on the value of $g(x,y)$ at the same point only. The resulting non-Gaussian field is called a translation field [5].

In the case of lognormal translation fields, Equation (25) becomes:

$$f(x,y) = \exp\left[\mu + \sigma g(x,y)\right] \tag{26}$$

where μ and σ the auxiliary parameters of the transformation.

3.3 Monte Carlo simulation

In the Monte Carlo simulation method, *NSIM* samples of a deterministic problem are generated and the problem is then solved as many times as the number of samples. The response variability of the problem is determined using well-known relationships of statistics. For example, the mean value and variance of the displacement u_i at degree of freedom i are computed as:

$$E(u_i) = \frac{1}{NSIM}\sum_{j=1}^{NSIM} u_i(j) \tag{27}$$

$$\sigma^2(u_i) = \frac{1}{NSIM-1}\left[\sum_{j=1}^{NSIM} u_i^2(j) - NSIM\, E^2(u_i)\right] \tag{28}$$

It is obvious that the accuracy of the estimation depends on the number of samples, which was selected equal to 1000.

3.4 Numerical examples

In this section, the generated sample functions of the elasticity components C_{11} and C_{44} are compared with the original fields and the response variability of two concrete structures is examined. More specifically, the analyzed structures include a simply supported and a cantilever beam subject to a uniform and a concentrated load respectively.

3.4.1 Generated sample functions

Figures 15a and 16a depict the generated sample functions of the elasticity components C_{11} and C_{44} in the same domain as the original fields, while Figures 15b and 16b depict their auto-correlations for lag values along x (ξ_x, $\xi_y = 0$) and y ($\xi_x = 0$, ξ_y). It is observed that the first two statistical moments and the auto-correlations of the sample functions of the elasticity components C_{11} and C_{44} are almost identical to those of the original fields.

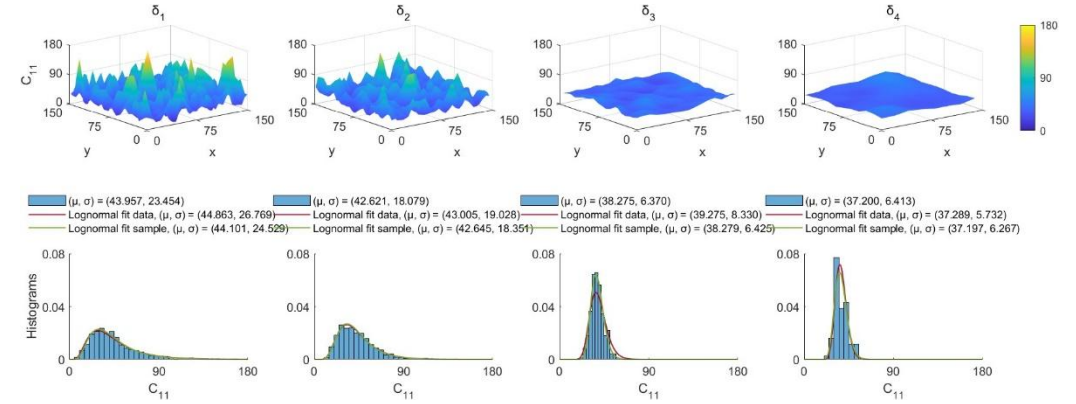


Figure 15a: Generated sample functions of the elasticity component C_{11}.

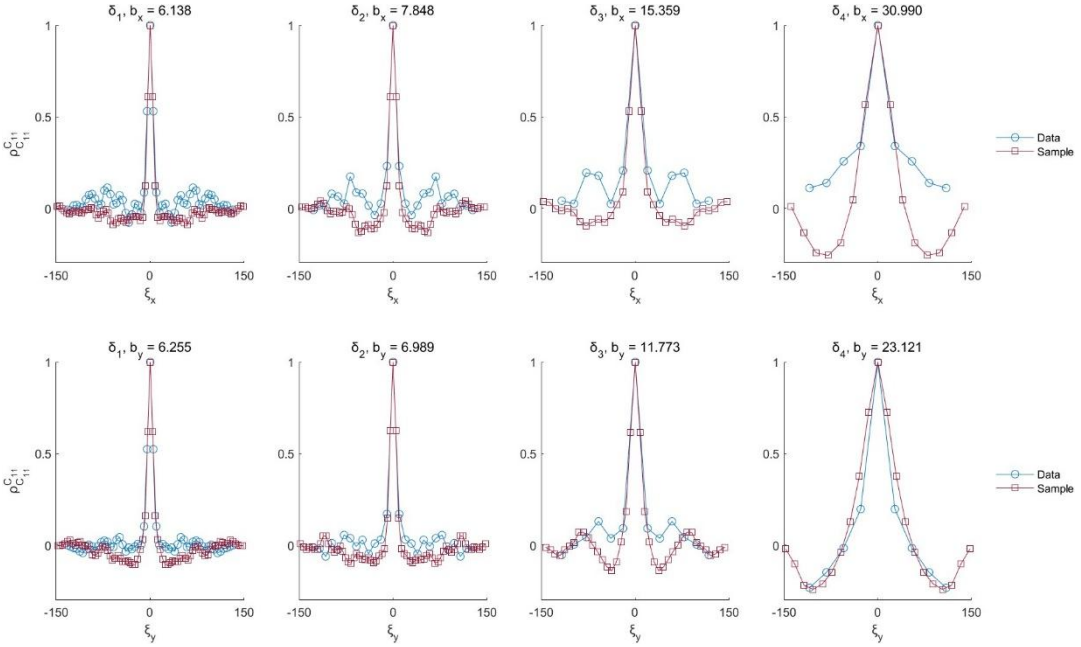


Figure 15b: Auto-correlations of the elasticity component C_{11} for lag values along x (ξ_x , $\xi_y = 0$) and y ($\xi_x = 0$, ξ_y).

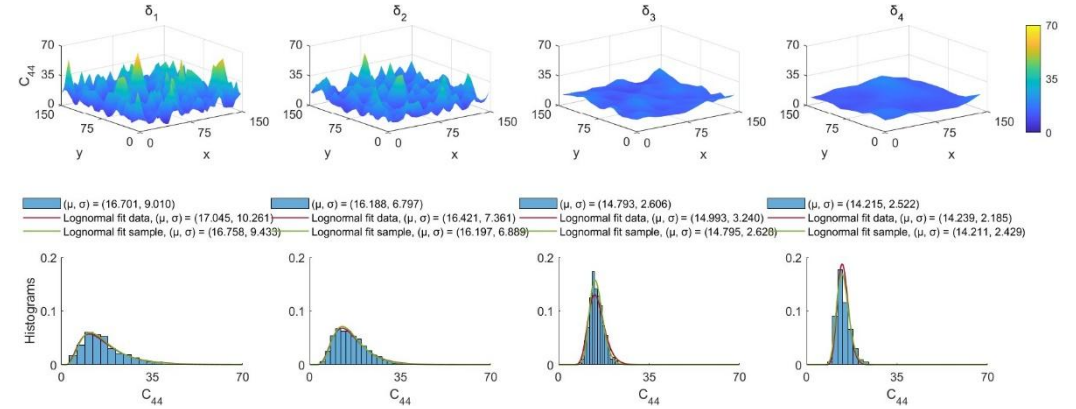


Figure 16a: Generated sample functions of the elasticity component C_{44}.

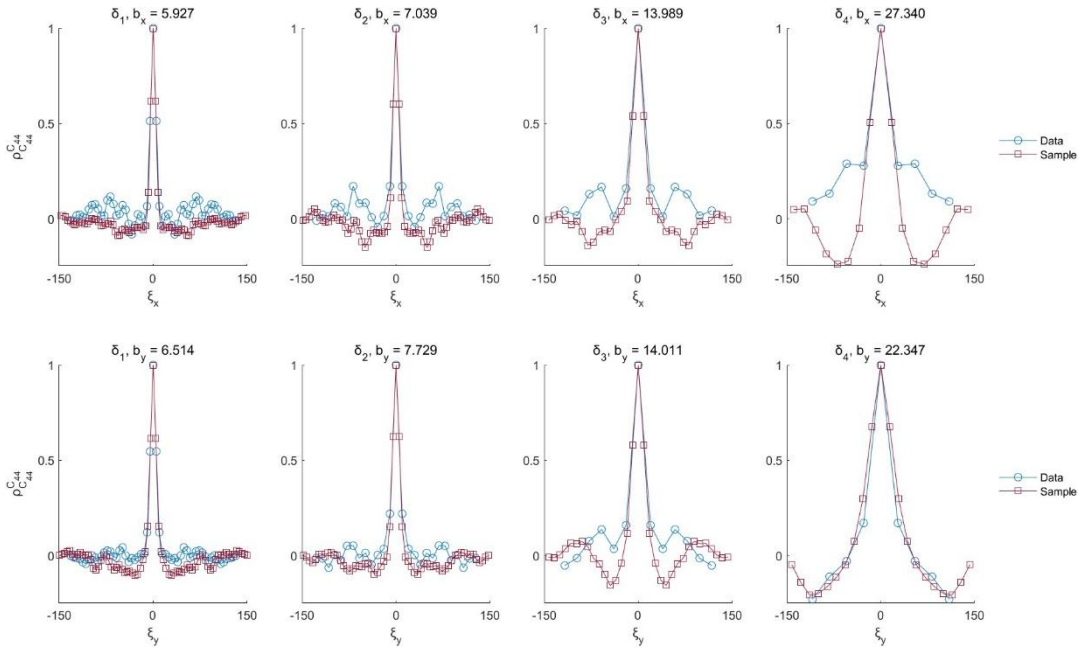


Figure 16b: Auto-correlations of the elasticity component C_{44} for lag values along x (ξ_x , $\xi_y = 0$) and y ($\xi_x = 0$, ξ_y).

3.4.2 Simply supported beam

The simply supported beam analyzed is depicted in Figure 17. The beam has a unit thickness and is modeled with four-node quadrilateral isoparametric elements under plane stress conditions. The element size is chosen between $b/4$ and $b/2$ with $b = \min\left(b_x, b_y\right)$, to sufficiently capture the variation of the random fields of the elasticity components C_{11} and C_{44}. The generated values of the sample functions correspond to the centroids of the elements. It is noted that the monitored response quantity is the vertical displacement u_y at the midpoint of the bottom edge of the beam.

The mean and COV of the displacement u_y are plotted in Figure 18 with respect to the number of Monte Carlo simulations, along with the response histogram. Statistical convergence of the results is achieved and it is observed that the structural response is influenced by the statistical characteristics of the random fields of the elasticity components C_{11} and C_{44} and particularly by the examined mesoscale sizes.

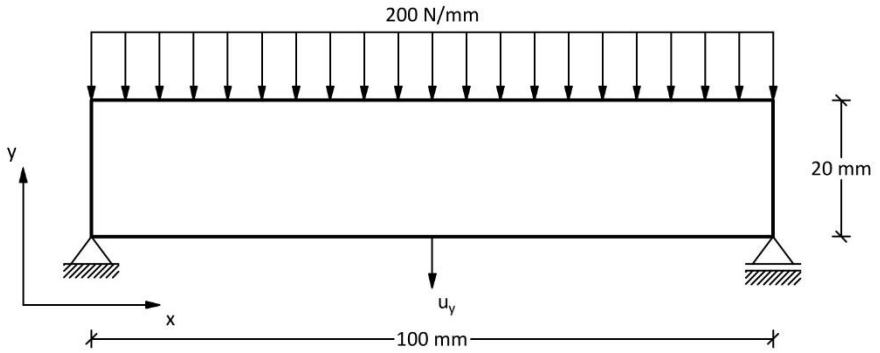


Figure 17: Simply supported beam subject to a uniform load.

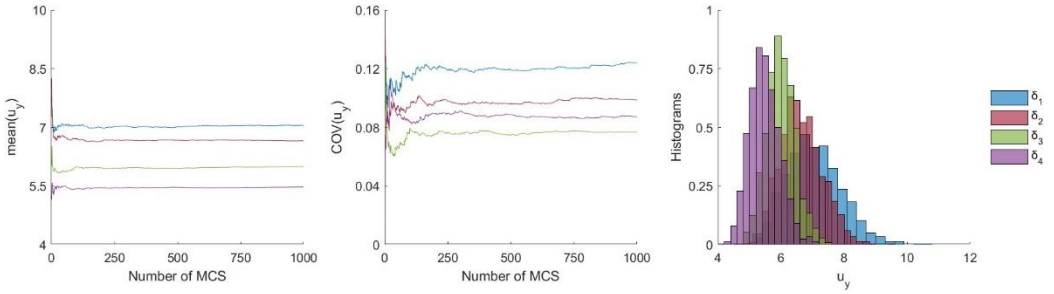


Figure 18: Mean and COV of the displacement u_y with respect to the number of Monte Carlo simulations, along with the response histogram.

3.4.3 Cantilever beam

The cantilever beam analyzed is depicted in Figure 19. The beam has a unit thickness and is modeled with four-node quadrilateral isoparametric elements under plane stress conditions. As in the previous example, the element size is chosen between $b/4$ and $b/2$ with $b = \min\left(b_x, b_y\right)$, to sufficiently capture the variation of the random fields of the elasticity components C_{11} and C_{44}. The generated values of the sample functions correspond to the centroids of the elements. It is noted that the monitored response quantity is the vertical displacement u_y at the free end of the bottom edge of the beam.

The mean and COV of the displacement u_y are plotted in Figure 20 with respect to the number of Monte Carlo simulations, along with the response histogram. Statistical convergence of the results is achieved and it is again observed that the structural response is influenced by the statistical characteristics of the random fields of the elasticity components C_{11} and C_{44} and particularly by the examined mesoscale sizes.

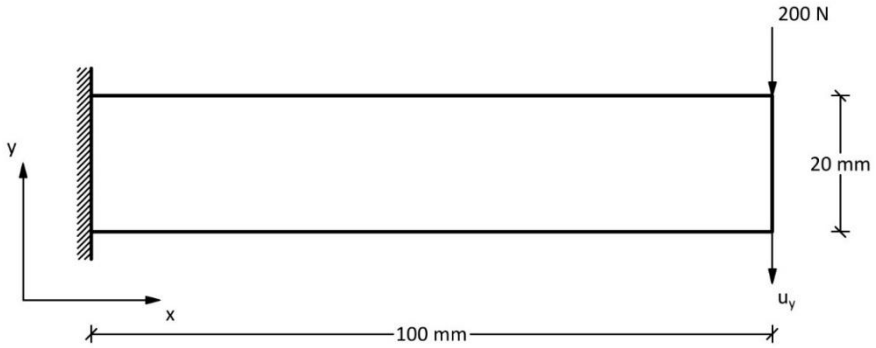


Figure 19: Cantilever beam subjected to a concentrated load.

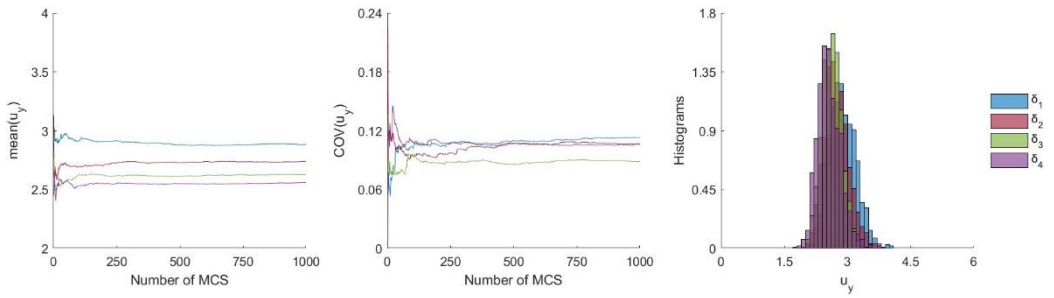


Figure 20: Mean and COV of the displacement u_y with respect to the number of Monte Carlo simulations, along with the response histogram.

4 CONCLUSIONS

In this work, the mesoscale random fields of the elasticity tensor of concrete were determined by applying the moving window method to the reconstructed microstructure of a cubic specimen. Additionally, their statistical characteristics were computed and the methodology for generating sample functions was described, in order to examine the response variability of two concrete structures and to understand the effect of the random microstructure on their macroscopic behavior. The key conclusions derived are as follows:

- The random fields of the elasticity components C_{11} and C_{44} become less variable and histograms narrower, as the scale factor increases. More specifically, the COV reduces from 63% to 16% for both elasticity components C_{11} and C_{44}, as the scale factor increases.
- The elasticity components C_{11} and C_{44} exhibit a strong correlation with each other, as expected.
- The correlation lengths of the random fields of the elasticity components C_{11} and C_{44} increase, as the scale factor increases. In other words, the random fields tend to a random variable, as the scale factor increases.
- The random fields of the elasticity components C_{11} and C_{44} are statistically isotropic at smaller scales, whereas a slight deviation from statistical isotropy is observed at larger scales.
- The elasticity components C_{11} and C_{44} are well described by the lognormal distribution.
- The first two statistical moments and the auto-correlations of the sample functions of the elasticity components C_{11} and C_{44} are almost identical to those of the original fields.
- The selected number of samples is sufficient to achieve statistical convergence of the results.
- The structural response is influenced by the statistical characteristics of the random fields of the elasticity components C_{11} and C_{44} and particularly by the examined mesoscale sizes.
- The response COV tends approximately to the COV of both elasticity components C_{11} and C_{44} at larger scales [3], while a reduction with respect to the input uncertainty is observed at smaller scales.

The combination of the above observations can lead to safer and more cost-effective design of concrete structures, as well as to the improvement of safety factors in construction codes. The consideration of the interfacial transition zone for determining the spatial variation of the elastic properties of concrete and the conduction of experiments to select the appropriate scale factor [12] are subjects of further research that will contribute even more in this direction.

ACKNOWLEDGEMENTS

The financial support provided by the John S. Latsis Public Benefit Foundation scholarship for Leontios Kotropoulos' postgraduate studies during the academic year 2024-2025 is gratefully acknowledged.

REFERENCES

[1] P. Wriggers & S. O. Moftah, "Mesoscale models for concrete: Homogenisation and damage behaviour", *Finite Elements in Analysis and Design*, vol. 42, no. 7, pp. 623-636, 2006. https://doi.org/10.1016/j.finel.2005.11.008

[2] G. Stefanou, D. Savvas & P. Metsis, "Random Material Property Fields of 3D Concrete Microstructures Based on CT Image Reconstruction", *Materials*, vol. 14, no. 6, p. 1423, 2021. https://doi.org/10.3390/ma14061423

[3] G. Stefanou, D. Savvas & M. Papadrakakis, "Stochastic finite element analysis of composite structures based on mesoscale random fields of material properties", *Computer Methods in Applied Mechanics and Engineering*, vol. 326, pp. 319-337, 2017. https://doi.org/10.1016/j.cma.2017.08.002

[4] M. Shinozuka & G. Deodatis, "Simulation of Multi-Dimensional Gaussian Stochastic Fields by Spectral Representation", *Applied Mechanics Reviews*, vol. 49, no. 1, pp. 29-53, 1996. https://doi.org/10.1115/1.3101883

[5] M. Grigoriu, "Simulation of Stationary Non-Gaussian Translation Processes", *Journal of Engineering Mechanics*, vol. 124, no. 2, 1998. https://doi.org/10.1061/(ASCE)0733-9399(1998)124:2(121)

[6] Z. Hashin, "Analysis of Composite Materials – A Survey", *Journal of Applied Mechanics*, vol. 50, no. 3, pp. 481-505, 1983. https://doi.org/10.1115/1.3167081

[7] R. Hill, "Elastic properties of reinforced solids: Some theoretical principles", *Journal of the Mechanics and Physics of Solids*, vol. 11, no. 5, pp. 357-372, 1963. https://doi.org/10.1016/0022-5096(63)90036-X

[8] D. Savvas, G. Stefanou & M. Papadrakakis, "Determination of RVE size for random composites with local volume fraction variation", *Computer Methods in Applied Mechanics and Engineering*, vol. 305, pp. 340-358, 2016. https://doi.org/10.1016/j.cma.2016.03.002

[9] M. Ostoja-Starzewski, S. Kale, P. Karimi, A. Malyarenko, B. Raghavan, S. I. Ranganathan & J. Zhang, "Chapter Two – Scaling to RVE in Random Media", *Advances in Applied Mechanics*, vol. 49, pp. 111-211, 2016. https://doi.org/10.1016/bs.aams.2016.07.001

[10] C. Miehe & A. Koch, "Computational micro-to-macro transitions of discretized microstructures undergoing small strains", *Archive of Applied Mechanics*, vol. 72, pp. 300-317, 2002. https://doi.org/10.1007/s00419-002-0212-2

[11] G. Stefanou, "The stochastic finite element method: Past, present and future", *Computer Methods in Applied Mechanics and Engineering*, vol. 198, no. 9-12, pp. 1031-1051, 2009. https://doi.org/10.1016/j.cma.2008.11.007

[12] G. Stefanou, D. Savvas, P. Gavallas & I. Papaioannou, "The effect of random field parameter uncertainty on the response variability of composite structures", *Composites Part C: Open Access*, vol. 9, p. 100324, 2022. https://doi.org/10.1016/j.jcomc.2022.100324

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

MULTISCALE STRUCTURAL OPTIMIZATION WITH EMBEDDED HIERARCHICAL MATERIAL ARCHITECTURES

Andrea Chiozzi[1] **and Andrea Nale**[2]

[1]University of Ferrara, Department of Environmental and Prevention Sciences
Via Ercole I d'Este 32, 44121 Ferrara, Italy
e-mail: andrea.chiozzi@unife.it

[2] University of Ferrara, Department of Architecture
Via della Ghiara 36, 44121 Ferrara, Italy
e-mail: andrea.nale@unife.it

Abstract. *Traditional topology optimization techniques enable the development of structurally efficient designs subject to specific constraints, loads, and boundary conditions. These methods typically involve continuous modifications of the topology throughout the optimization process. In this work, we introduce an innovative multiscale structural optimization framework, based on the minimization of compliance, that decouples design across scales: the microscale topology is optimized using a multi-material strategy, while the macroscale geometry of the domain remains fixed. This formulation takes advantage of embedding optimized hierarchical material architectures at the microscale, thereby enabling exploration across a broad range of structural configurations. Meaningful numerical examples are provided to demonstrate the capabilities and relevance of the proposed method.*

Keywords: Multiscale structural optimization, topology optimization, architected materials, homogenization, additive manufacturing

1 INTRODUCTION

Topology optimization is a computational design methodology used to determine the most efficient material layout within a defined domain, based on specific boundary conditions and loading scenarios, while optimizing a particular objective under given constraints. Historically, research in this area focused largely on theoretical developments due to the difficulty in manufacturing the complex geometries produced by these methods. However, with the advent of advanced manufacturing technologies, particularly 3D printing, many of these limitations have been mitigated. This progress has enabled the application of topology optimization across various engineering disciplines, including automotive, aerospace, and biomedical sectors.

The shift from traditional manufacturing to additive manufacturing (AM) has also opened new opportunities in industry. The motivation for this transition is driven by the need to reduce the environmental footprint and lower production costs. Integrating topology optimization with AM is therefore seen as a significant advancement, as it promotes energy efficiency and reduces carbon emissions. Topology optimization helps achieve material efficiency by designing structures that use the minimum necessary material, following the principle of "limits of economy" [1]. Concurrently, AM minimizes waste and supports the use of sustainable and recycled materials, such as natural aggregates combined with environmentally friendly binders.

The synergy between AM and topology optimization has encouraged exploration into structural design across multiple scales. Control at the microscale has enabled the development of lightweight cellular materials with tailored properties at the macroscale. Nevertheless, common periodic microstructures, such as truss or plate-based designs, often suffer from stress concentrations at junctions, affecting their strength and reliability.

An effective alternative is found in triply periodic minimal surfaces (TPMS) [2], which avoid stress concentrations through smooth geometries and offer superior scalability thanks to their doubly curved surfaces. More recently, attention has shifted to non-periodic stochastic structures inspired by spinodal decomposition processes during phase separation. These spinodal architectures also exhibit smooth, non-intersecting interfaces between material phases, akin to TPMS [3].

In this study, we introduce a volume-constrained, multiscale, multi-material topology optimization framework aimed at maximizing structural stiffness by optimally distributing various types of material architectures with different densities and orientations [4], such as TPMS [5] or spinodal architectures [6]. This framework can be integrated with large-scale additive manufacturing using a voxel-based post-processing approach [6].

This paper is structured as follows: Section 2 presents the optimization formulation, the characteristics of non-periodic architected materials, and the applied additive manufacturing techniques. Section 3 discusses numerical results, and Section 4 concludes with final thoughts and directions for future research.

2 MATERIALS AND METHODS

The presented topology optimization framework adopts a multi-material strategy grounded in homogenization, enabling the integration and manipulation of various microstructures throughout the iterative design process. As defined in Eq. 1, the objective is to minimize structural compliance, J, while satisfying a set of volume constraints, g_k.

$$
\begin{aligned}
\min_{\mathbf{Z},\rho,\alpha,\beta,\gamma} \quad & J = \mathbf{F}^T\,\mathbf{U}(\mathbf{Z},\rho,\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}),\\
\text{s.t.} \quad & g_k = \frac{\sum_{m\in\mathcal{G}_k}\sum_{\ell\in\mathcal{E}_j} A_\ell m_v(y_{\ell m})}{\sum_{\ell\in\mathcal{E}_k} A_\ell} - \overline{v}_k \le 0,\ k = 1,...,N^{\mathrm{c}},\\
& Z_{\ell m} \in [0,1],\ \ell = 1,...,N^{\mathrm{el}},\ m = 1,...,N^{\mathrm{mat}},\\
& \rho_\ell \in [\underline{\rho},\overline{\rho}],\{\alpha_{\ell m},\beta_{\ell m},\gamma_{\ell m}\} \in [-\pi,\pi],\\
\text{with} \quad & \mathbf{K}(\mathbf{Z},\rho,\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma})\,\mathbf{U}(\mathbf{Z},\rho,\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}) = \mathbf{F},
\end{aligned}
\tag{1}
$$

The volume constraint is defined with flexibility, allowing for either global or localized material limitations. Local constraints are implemented by restricting material choices within specific subdomains, identified through lists of element indices. Specifically, each material index list, $\mathcal{G}k$, specifies the materials permitted during optimization, and is associated with an element index list, $\mathcal{E}k$. For global constraints, $\mathcal{E}_k$ refers to the entire domain, whereas for local constraints, it targets defined subdomains. Additionally, passive regions can be designated within the domain—these can either be voids or regions composed of a specific material, referred to respectively as passive-void and passive-solid zones.

The design variable space includes material presence $\mathbf{Z}$, spinodal density $\boldsymbol{\rho}$—bounded between $\underline{\rho}$ and $\overline{\rho}$ to ensure manufacturability—and spinodal orientation angles $\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}$. The material distribution field $\mathbf{Z}$ is regularized using a linear filter [7], which applies a specified radius to ensure mesh independence and well-posedness of the optimization problem. To promote distinct material–void transitions, this filter is combined with a Heaviside projection [8]. The resulting filtered density field $\tilde{Y}$ is further processed using a SIMP (Solid Isotropic Material with Penalization) interpolation scheme [9] to discourage intermediate density values.

To progressively refine the solution, continuation methods are applied to gradually increase both the material penalization factor p and the material mixing parameter τ until target values are reached, either upon convergence or after a specified number of iterations. Once the optimal material layout is obtained, the solid boundaries are smoothed appropriately. A post-processing step is then necessary to both visualize and incorporate the designed microstructures.

The computational domain is discretized using finite elements, with each element linked to a corresponding set of design variables. The displacement field, $\mathbf{U}$, is calculated by solving the linear elastic equilibrium equation $\mathbf{K}(\mathbf{Z},\boldsymbol{\rho},\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}),\mathbf{U}(\mathbf{Z},\boldsymbol{\rho},\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}) = \mathbf{F}$, where $\mathbf{F}$ is a prescribed, design-independent vector of nodal forces. The global stiffness matrix $\mathbf{K}$ is assembled from the elemental stiffness contributions [10] as described below:

$$
\mathbf{K}_\ell = \int_{\Omega_\ell} \mathbf{B}^T\mathbf{D}_\ell(\mathbf{Z}_\ell,\rho_\ell,\boldsymbol{\alpha}_\ell,\boldsymbol{\beta}_\ell,\boldsymbol{\gamma}_\ell)\mathbf{B}d\mathbf{x},, \tag{2}
$$

The element-strain displacement matrix is $\mathbf{B}$ and the material matrix, $\mathbf{D}$, is calculated by multi-material interpolation of the m-candidate materials material matrix, D_m^H, according to:

$$
\mathbf{D}_\ell = \sum_{m=1}^{N^m} w_{\ell m} \prod_{\substack{q=1\\ q\neq m}}^{N^m} (1-\tau w_{\ell q})\mathbf{M}(\alpha_{\ell m},\beta_{\ell m},\gamma_{\ell m})\mathbf{D}_m^H(\rho_\ell)\mathbf{M}^T(\alpha_{\ell m},\beta_{\ell m},\gamma_{\ell m}). \tag{3}
$$

The optimal orientation of spinodal architectures reference frame is obtained by tensor transformation laws, $\mathbf{M}(\alpha_{\ell m},\beta_{\ell m},\gamma_{\ell m})$, for stress and strains. The angles $(\alpha_{\ell m},\beta_{\ell m},\gamma_{\ell m})$ are updated by Eq. 6. The m-candidate materials are included by a homogenized material elasticity tensor,

D_m^H, calculated by classical homogenization theory [11] for assigned densities and topologies. The stochastic and non-periodic nature of spinodal topology is considered in the implementation by calculating the elastic tensor by means of 15 spinodal architectures. Each topology is calculated by applying level-set function to Eq. 11 and the computational homogenization is performed by discretizing the topology in 150x150x150 elements with Young's modulus, E = 1, and Poisson's ratio, $\nu = 0.3$ for the bulk material. The problem in Eq. (1) is solved by a gradient-based approach. The optimal solution is ruled by the sensitivities of objective and constraint functions with respect to design variables. The augmented lagrangian method [12] is adopted with the Steepest Descent Method (SDM) [13] to iteratively guide the design toward an optimal solution, a local minimum. The augmented lagrangian function is minimized in each inner iteration t as the sum of objective function and the penalty term and reads:

$$\mathcal{L}(\mathbf{x})^{(\mathbf{t})} = J(\mathbf{x}) + \sum_{j=1}^{K}\left[\lambda_j^{(t)} \max\left(g_j(x), -\frac{\lambda_j^{(t)}}{\mu^{(t)}}\right) + \frac{\mu^{(t)}}{2} max\left(g_j(x), -\frac{\lambda_j^{(t)}}{\mu^{(t)}}\right)^2\right], \tag{4}$$

The penalization parameters, $\lambda_j^{(k)}$ and $\mu^{(k)}$, are updated every 5 outer iterations, k, as:

$$\lambda_j^{(k+1)} = \lambda_j^{(k)} + \mu^{(k)} \max\left(g_j^{(k)}(\mathbf{x}), -\frac{\lambda_j^{(t)}}{\mu^{(k)}}\right) \; and \; \mu^{(k+1)} = 1.25\mu^{(k)}. \tag{5}$$

At each inner optimization iteration, t, the variables $\mathbf{x}$ are updates following the rule:

$$\mathbf{x}^{(t+1)} = \max\left[\min\left(\mathbf{x}^{(t)} - \tau^{(t)}\frac{\partial \mathcal{L}^{(t)}}{\partial \mathbf{x}}, \mathbf{x}^{(t)} + move\right), \mathbf{x}^{(t)} - move)\right], \tag{6}$$

where $move$ is move limit and $\tau^{(t+1)} = \max(0.99\tau^{(t)}, 0.01)$ is the step size with $\tau^{(0)} = 1$.
The sensitivities involved in the update scheme are the objective and penalty function and they are computed as follows:

$$\begin{aligned}
&\frac{\partial J}{\partial \mathbf{z}_m} = \frac{\partial y_{\ell m}}{\partial z_{\ell m}}\frac{\partial \tilde{y}_{\ell m}}{\partial y_{\ell m}}\frac{\partial J}{\partial w_{\ell m}}, \; \frac{\partial J}{\partial \boldsymbol{\alpha}_m} = \frac{\partial J}{\partial \alpha_{\ell m}}, \frac{\partial J}{\partial \rho} = \frac{\partial J}{\partial \rho_\ell} \\
&\frac{\partial g_j}{\partial \mathbf{z}_m} = \frac{\partial y_{\ell m}}{\partial z_{\ell m}}\frac{\partial \tilde{y}_{\ell m}}{\partial y_{\ell m}}\frac{\partial v_{\ell m}}{\partial \tilde{y}_{\ell m}}\frac{\partial g_j}{\partial v_{\ell m}}, \; \frac{\partial g_j}{\partial \boldsymbol{\rho}} = \frac{\partial v_{\ell m}}{\partial \rho_\ell}\frac{\partial g_j}{\partial v_{\ell m}}, \\
&m = 1, ..., N^m, \; j = 1, ..., N^c
\end{aligned} \tag{7}$$

where:

$$\begin{aligned}
&\frac{\partial y_m}{\partial Z_m} = \mathbf{P}^T, \frac{\partial w}{\partial \tilde{y}_{\ell m}} = p y_{\ell m}^{p-1}, \frac{\partial \tilde{y}}{\partial y_{\ell m}} = \frac{\xi(1 - \tanh^2(\xi(y_{\ell m} - \eta)))}{\tanh(\xi\eta) + \tanh(\xi(1-\eta))}, \\
&\frac{\partial v_{\ell m}}{\partial \tilde{y}} = \rho_\ell, \frac{\partial v_{\ell m}}{\partial \rho_k} = \tilde{y}_{\ell m}, \frac{\partial g_j}{\partial v_{\ell m}} = \frac{V_\ell}{\sum_{\ell \in E_j} V_\ell}
\end{aligned} \tag{8}$$

The components of each objective and constraint sensitivities are defined as:

$$\frac{\partial J}{\partial w_{\ell m}} = -\mathbf{U}^T\frac{\partial \mathbf{K}}{\partial w_{\ell m}}\mathbf{U}, \; \frac{\partial J}{\partial \rho_\ell} = -\mathbf{U}^T\frac{\partial \mathbf{K}}{\partial \rho_\ell}\mathbf{U}, \frac{\partial J}{\partial \alpha_{\ell m}} = -\mathbf{U}^T\frac{\partial \mathbf{K}}{\partial \alpha_{\ell m}}\mathbf{U} \tag{9}$$

where the stiffness sensitivities are computed from the material elasticity tensor, with respect to the design variable, $\mathbf{w}$, the spinodal density, ρ, and the orientation angle (α):

$$\frac{\partial \mathbf{D}_\ell}{\partial w_{\ell m}} = \prod_{\substack{q=1 \\ q\neq m}}^{N^m} (1-\tau w_{\ell q}) \mathbf{M}_{\ell m} \mathbf{D}_m^H(\rho_\ell) \mathbf{M}_{\ell m}^T - \sum_{\substack{q=1 \\ q\neq m}}^{N^m} \gamma w_{\ell q} \prod_{\substack{r=1 \\ r\neq q \\ r\neq m}}^{N^m} (\tau_{\ell r}) \mathbf{M}_{\ell m} \mathbf{D}_m^H(\rho_\ell) \mathbf{M}_{\ell m}^T,$$

$$\frac{\partial \mathbf{D}_\ell}{\partial \rho_\ell} = \sum_{m=1}^{N^m} w_{\ell m} \prod_{\substack{q=1 \\ q\neq m}}^{m} (1-\tau w_{\ell q}) \mathbf{M}_{\ell m} \frac{\partial \mathbf{D}_m^H(\rho_k)}{\partial \rho_\ell} \mathbf{M}_{\ell m}^T, \tag{10}$$

$$\frac{\partial \mathbf{D}_\ell}{\partial \alpha_{\ell m}} = \sum_{m=1}^{N^m} w_{\ell m} \prod_{\substack{q=1 \\ q\neq m}}^{N^m} (1-\tau w_{\ell q}) \left[\frac{\partial \mathbf{M}_{\ell m}}{\partial \alpha_{\ell m}} \mathbf{D}_m^H \mathbf{M}_{\ell m}^T + \mathbf{M}_k \mathbf{D}_m^H \frac{\partial \mathbf{M}_{\ell m}^T}{\partial \alpha_{\ell m}} \right]$$

with $\partial \mathbf{D}/\partial \beta_{\ell m}$ and $\partial \mathbf{D}/\partial \gamma_{\ell m}$ and of similar form.

3 NUMERICAL RESULTS

The optimization formulation described in Eq. 1 is coupled with a homogenization strategy that facilitates the incorporation of microstructural features into the design of multiscale, hierarchical structures. In this work, we adopt spinodal architected materials as the microstructural design basis. These materials derive their topology from the thermodynamic phenomenon of spinodal decomposition [16], wherein a homogeneous phase at peak free energy undergoes spontaneous separation into two distinct phases. This separation is initiated by small fluctuations in concentration, which lower the system's free energy.

Spinodal decomposition was originally modeled by the Cahn–Hilliard equation [17], which, while accurate, is computationally intensive and offers limited control over the resulting microstructure. To address this, a more practical phase-field approximation based on Gaussian random fields (GRFs) has been proposed [18].

The introduction of a GRF, ω, sets anisotropic features by manipulating the stochastic distribution of a set of wave vectors, $\mathbf{n}_i, i = 1, ...N^w$, on a unit sphere $\mathbf{n}_i \in \mathcal{U}[\mathbb{S}^2]$. The GRF in the Cartesian space $\mathbb{R}^3$ with basis $\{\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2, \hat{\mathbf{e}}_3\}$ is computed as follows:

$$\omega(\mathbf{x}) = \sqrt{\frac{2}{N^{\mathrm{w}}}} \sum_{j=1}^{N^w} \cos \beta \mathbf{n}_j \cdot \mathbf{x} + \zeta_j, \tag{11}$$

with amplitude, N^w, wavelength, β and phase shift $\zeta_j \in \mathcal{U}[0, 2\pi)$ of the j-th wave vector sampled as:

$$\mathbf{n}_j \in \mathcal{U}[\{\mathbf{m} \in \mathbb{S}^2 : (|\mathbf{m} \cdot \hat{\mathbf{e}}_1| > \cos\theta_1) \oplus (|\mathbf{m} \cdot \hat{\mathbf{e}}_2| > \cos\theta_2) \oplus (|\mathbf{m} \cdot \hat{\mathbf{e}}_3| > \cos\theta_3)\}], \tag{12}$$

where the angle set $\{\theta_1, \theta_2, \theta_3\} \in [0, \pi/2]$ restricts the wave vectors 12 on an unit sphere [18] providing the desirable anisotropic features.

The numerical model and the physical sample include the lamellar spinodal architecture. The lamellar class is generated by restricting wave vectors through cone angles $\{\theta_1, \theta_2, \theta_3\} = \{0, 30, 0\}$. The homogenized properties are included in the optimization formulation by the elastic matrix D^H.

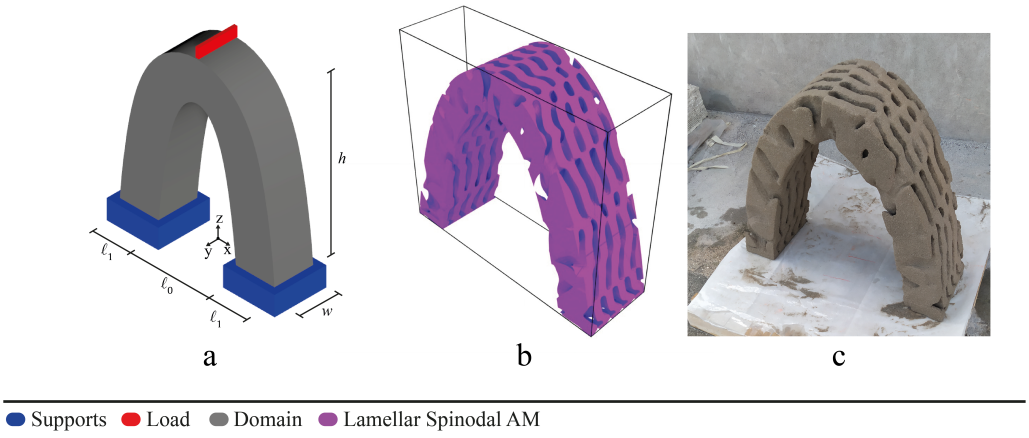


Figure 1 Lamellar Catenary Arch: a) domain, b) numerical model, c) manufactured sample.

The proposed topology optimization framework has been here applied using a single material class; specifically, the lamellar spinodal architecture—subject to a global volume constraint $\overline{v}$. The optimization process focuses on determining the optimal spatial orientation of the microstructure within a 3D domain. The design space is defined as a catenary arch, with its geometry and boundary conditions illustrated in Fig. 1. The central axis of the structure follows the inverse of the standard catenary equation: $z = a\cosh(x/a) = a(e^{x/a}+e^{-x/a})/2$. To generate the solid arch, the parameter $a = 1/6$ sets the vertical distance from the lowest point of the arch to the x-axis, while a constant depth of $\ell_1 = 15$ cm is maintained along its axis. The structure spans a length $\ell_0 = 65$ cm, yielding a total length of $\ell = 95$ cm, a height of $h = 83$ cm, and a fixed width of $w = 30$ cm.

The optimization is initialized by assigning values to the design variables: the material volume is defined as $v_{lm} = \overline{v}/\rho_\ell$ for all elements ℓ, with the initial density set to the midpoint value $\rho_\ell = (\underline{\rho} + \overline{\rho})/2$, and all orientation angles initialized as $\alpha_\ell = \beta_\ell = \gamma_\ell = 0$. A linear filter with radius $R = 0.4$ is applied to the density field to ensure smooth transitions. The design variables are then iteratively updated according to Eq. 6, with maximum allowable changes of $move_{\mathrm{z}} = 0.1$ for density and $move_{\mathrm{ang}} = 0.25$ for orientation angles. Convergence is assessed by monitoring the variation in the density field $\mathbf{Z}$ until it falls below a specified tolerance tol, with each penalization phase limited to $\max^{\mathrm{iter}} = 150$ iterations.

The search for an optimal solution is conducted through a continuation scheme involving five stages of penalization, using material penalty values $p = [1, 2, 2, 3, 4]$. Since the formulation involves only one material class, the multi-material interpolation factor τ is not needed. The iterative optimization routine refines the material layout, porosity distribution, and microstructural orientation throughout the domain.

This methodology is specifically adapted for water jetting powder-bed additive manufacturing, which is well-suited for producing self-supporting geometries. Due to the inherent lack of tensile strength in the printed material, the approach favors compression-dominant structural forms. Both simulation results and the fabricated prototype confirm the feasibility of using this AM process to construct large-scale components from non-periodic architected materials.

4 CONCLUSIONS

This work demonstrates the scalability of spinodal architectures within a large-scale additive manufacturing context, exemplified by the fabrication of an optimally oriented lamellar cate-

nary arch. Given the brittle, stone-like nature of the printed materials used—characterized by negligible tensile strength—along with the characteristics of the adopted optimization method, a self-supporting structure was chosen as the design input. Such structures are predominantly subjected to compressive stress, aligning with the material's mechanical behavior. The adopted topology optimization approach does not incorporate any failure criteria. Although stress-based formulations—such as the one proposed in [19]—might be more suitable for practical applications, their integration within multiscale systems remains a significant challenge. Looking ahead, future research will explore the incorporation of failure-based criteria, such as stress-constrained formulations, and the use of many different classes of materials architectures at the microscale.

5 ACKNOWLEDGEMENTS

Andrea Chiozzi gratefully acknowledges the financial support of European Union - Next Generation EU, Italian Ministry of University and Research (MUR) under the program "Progetti di Ricerca di Rilevante Interesse Nazionale (PRIN), Bando 2022", D.D. n. 104/2022, project "COMˆ3D-CREATE", grant No. 2022SMSZZ9.

REFERENCES

[1] Michell, A. G. M. (1904). The limits of economy of material in frame-structures. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 8(47), 589-597*

[2] Feng, J., Fu, J., Yao, X., & He, Y. (2022). Triply periodic minimal surface (TPMS) porous structures: from multi-scale design, precise additive manufacturing to multidisciplinary applications. *International Journal of Extreme Manufacturing, 4(2), 022001*

[3] Vidyasagar, A., Krödel, S., Kochmann, D. (2018). Microstructural patterns with tunable mechanical anisotropy obtained by simulating anisotropic spinodal decomposition. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 474, 20180535*

[4] Sanders, E. D., Pereira, A., Aguiló, M. A., & Paulino, G. H. (2018). PolyMat: an efficient Matlab code for multi-material topology optimization. *Structural and Multidisciplinary Optimization, 58, 2727-2759*

[5] Nale, A., Chiozzi, A. (2025). Multiscale topology optimization with embedded TPMS Architected Materials. *International Journal of Mechanical Sciences, (in press).*

[6] Nale, A., Chiozzi, A., Senhora, F.V., and Paulino, G.H. (2025). Large-scale additive manufacturing of optimally-embedded spinodal material architectures. *Additive Manufacturing, 101, 104700*

[7] Bourdin, B. (2001). Filters in topology optimization. *International journal for numerical methods in engineering, 50(9), 2143-2158*

[8] Guest, J. K., Prévost, J. H., & Belytschko, T. (2004). Achieving minimum length scale in topology optimization using nodal design variables and projection functions. *International journal for numerical methods in engineering, 61(2), 238-254*

[9] Bendsøe, M. P. (1989). Optimal shape design as a material distribution problem. *Structural optimization, 1, 193-202*

[10] Bathe, K. J. (1996). Finite element procedures. *Klaus-Jurgen Bathe*

[11] Hassani, B., & Hinton, E. (2012). Homogenization and structural topology optimization: theory, practice and software. *Springer Science & Business Media*

[12] Nocedal, J., & Wright, S. J. (Eds.). (1999). Numerical optimization. *Springer New York*

[13] Bertsekas, D. P. (1997). Nonlinear programming. *Journal of the Operational Research Society, 48(3), 334-334*

[14] Senhora, F. V., Sanders, E. D., & Paulino, G. H. (2022). Optimally-tailored spinodal architected materials for multiscale design and manufacturing. *Advanced Materials, 34(26), 2109304*

[15] Desamanera. La stampa 3D dei Minerali per le grandi dimensioni. *www.desamanera.com*

[16] Binder, K. (1987). Theory of first-order phase transitions. *Reports on progress in physics, 50(7), 783*

[17] Cahn, J. W., & Hilliard, J. E. (1958). Free energy of a nonuniform system. I. Interfacial free energy. *The Journal of chemical physics, 28(2), 258-267*

[18] Portela, C. M. (2019). Fabrication, Mechanical Characterization, and Modeling of 3D Architected Materials Upon Static and Dynamic Loading. *California Institute of Technology*

[19] Senhora, F. V., Giraldo-Londono, O., Menezes, I. F., & Paulino, G. H. (2020). Topology optimization with local stress constraints: a stress aggregation-free approach. *Structural and Multidisciplinary Optimization, 62, 1639-1668*

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

ACOUSTIC WAVE TRANSMISSION OF MICROPOLAR PLATES

A. Pau[1], C. Meso[2], Z.E.A. Fellah[2], and E. Ogam[2]

[1]Department of Structural and Geotechnical Engineering, Sapienza University of Rome,
via Eudossiana 18, 00184 Rome, Italy
e-mail: annamaria.pau@uniroma1.it

[2] Aix Marseille Univ, CNRS, Centrale Med, LMA, Marseille, France
e-mail: {meso,fellah,ogam}@lma.cnrs-mrs.fr

Abstract. *This study investigates the interaction of sound waves with expanded (EPS) polystyrene plates, which possess a microstructure composed of interacting grains. The Cosserat or micropolar theory of elasticity is recognized for its ability to account for the internal length scales inherent to materials with microstructure. This theory introduces additional independent rotational degrees of freedom, along with associated curvature strain and couple stress descriptors, enabling the continuum to support a diverse range of bulk waves. Therefore, we apply the Cosserat theory to model through-thickness vibrations of linear elastic EPS plates, aiming to reproduce the experimental response. Experiments were carried out to measure the transmission coefficient of such low density plates. With a comprehensive experimental setup including a loudspeaker, a microphone, and a data acquisition system, through-thickness frequencies were observed within the frequency range between 2 kHz and 70 kHz. The comparison between theoretical and experimental results allows for the identification and calibration of material parameters.*

Keywords: Transmission Coefficients, Micropolar Elasticity, Cosserat Continuum, Through-Thickness Frequency.

1 INTRODUCTION

Following Voigt's foundational work [1], significant discrepancies emerged between experimentally observed dynamics and classical elasticity predictions, particularly in polymers, granular, and porous solids. These deviations stem from microstructural effects, absent in classical theories [2]. In granular and multimolecular materials, these effects become pronounced, giving rise to novel wave types unexplained by classical elasticity.

Expanding on the work of the Cosserat brothers [3], the theory of asymmetric elasticity introduces a framework where each material point in a continuum is associated with a rigid *triad* that can rotate independently of its translational displacement. This formulation established the balance equations for momentum in dynamic cases. A rigorous and complete linear theory of micropolar elasticity, incorporating microinertia effects, was later developed by Eringen [4, 5].

This study focuses exclusively on *micropolar linear elastic media*, hereafter referred to as the *Cosserat continuum*. This framework preserves the reversibility of deformations under mechanical stress while explicitly accounting for microstructural effects by introducing additional rotational degrees of freedom at each material point. Most research on micropolar elastodynamics has remained largely theoretical [6]. However, recent experimental studies have examined polystyrene plates, both expanded and extruded [7, 8].

Beyond fundamental insights, this study conducts a through-thickness analysis of vibrations, establishing the dependence of longitudinal resonances on the Cosserat coupling modulus κ—an effect absent in classical elasticity. Experimental validation over the 2 – 70 kHz range confirms predicted resonances on a polystyrene plate sample, enabling parameter identification. Sections 2-4 shortly present the theoretical model and the analysis carried out, leading to numerical values of through-thickness frequencies. Section 5 details the experimental setup and results, while Section 6 concludes with key findings.

2 DESCRIPTION OF THE PROBLEM

A sketch of the experiment is shown in Figure 1a. Sound waves generated by a loudspeaker propagate through the air and impinge on a plate. Within the plate, wave propagation occurs in the form of elastic waves, which reflect and transmit through its thickness. The oscillation of the plate's surface generates transmitted sound waves in the surrounding medium, which are recorded by a microphone placed beyond the obstacle. In our model, we focus solely on the solid plate and assume that it is immersed in vacuum, neglecting any fluid-structure interaction. Our objective is to investigate whether and how the microstructure of the plate, described by a micropolar model, influences wave propagation within the solid.

3 EQUATIONS OF MOTION

The model of the micropolar continuum is based on Eringen's theory of micropolar elasticity [5]. The set of coupled differential equations that govern the field motion of a micropolar continuum are the following:

$$\begin{aligned}(\mu+\alpha)\Delta\mathbf{u}+(\lambda+\mu)\nabla(\operatorname{div}\mathbf{u})+\kappa\operatorname{curl}\boldsymbol{\theta} &= \rho\ddot{\mathbf{u}}\\ \gamma\Delta\boldsymbol{\theta}+(\alpha+\beta)\nabla(\operatorname{div}\boldsymbol{\theta})+\kappa\operatorname{curl}\mathbf{u}-2\kappa\boldsymbol{\theta} &= \rho J\ddot{\boldsymbol{\theta}}\end{aligned} \tag{1}$$

where $\mathbf{u}$ and $\boldsymbol{\theta}$ are, respectively, the unknown displacement and microrotation fieds that are a function of time t and of the position vector $\mathbf{x} = \{x_1, x_2, x_3\}^T$ (Figure 1 b). The double dots indicate the second derivative with respect to time t. Moreover, $\lambda, \mu, \alpha, \beta, \gamma$ and κ are material

moduli, ρ is the density of the material, and J is the microinertia. The differential operators, when applied to a vector field $\mathbf{v} = \{v_1, v_2, v_3\}^T$ provide the following results: $\Delta \mathbf{v}_i = v_{i,i} + v_{i,j} + v_{i,k}$, div $\mathbf{v} = v_{1,1} + v_{2,2} + v_{3,3}$, curl $\mathbf{v}_i = (-1)^{(i-1)}(v_{k,j} - v_{j,k})$ where the comma subscript indicates the derivation with respect to the spatial coordinates and $i = 1, 2, 3, j = 1, 2, 3, k = 1, 2, 3$. The gradient ∇ applied to a scalar field ϕ gives the vector $\nabla\phi = \{\phi_{,1}\ \phi_{,2}\ \phi_{,3}\}^T$.

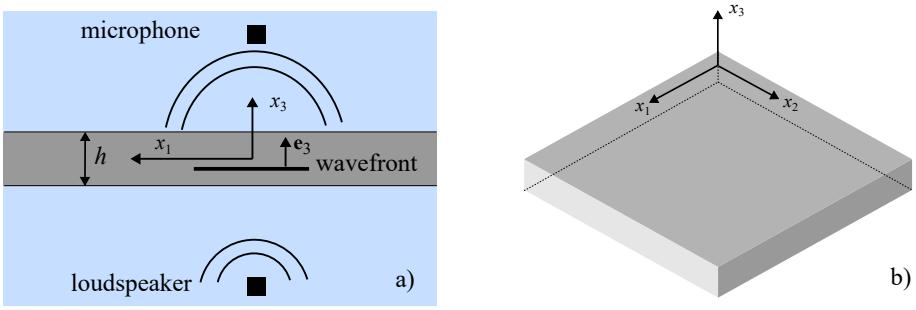


Figure 1: Sketch of the experiment and cross-section of the plate (a) and system of coordinates (b).

4 THROUGH-THICKNESS FREQUENCIES OF A MICROPOLAR PLATE

A cross section of the plate under study is reported in Figure 1a. We want to describe the vibrations of the plate when a plane wave with wavefront parallel to the upper and lower free-stress surfaces propagates within the plate thickness. Such kind of harmonic progressive waves with frequency ω (in rad/s) advancing in the positive direction of the vector $\mathbf{e}_3$, which is normal to the wavefront, are expressed as:

$$\mathbf{u}(x_3, t) = \mathbf{a}e^{I(k\,x_3 - \omega t)} \quad , \quad \boldsymbol{\theta}(x_3, t) = \mathbf{b}e^{I(k\,x_3 - \omega t)} \tag{2}$$

where k is the wavenumber, $\mathbf{a}$ and $\mathbf{b}$ are the amplitude vectors, and I is the imaginary unit.

Substitution of the tentative solution (2) into the field equations (1), leads to a first decoupled eigenvalue problem where in-plane and out-of-plane vibrations behave independently. In this eigenvalue problem, which has quadratic structure, the eigenvalue is the wavenumber k and the eigenvectors are the amplitude vectors, which are both functions of ω. In-plane and out-of-plane vibrations are associated with displacement components $\{u_1, u_3, \theta_2\}$ and $\{u_2, \theta_1, \theta_3\}$, respectively. We restrict our interest to in-plane vibrations. The six related eigenvalues k_i with $i = 1..6$ are associated with eigenvectors involving displacements along u_3 only, which are longitudinal waveforms, or displacements along u_1 and microrotations θ_2, which will be called shear/micropolar waveforms. The in-plane longitudinal and shear/micropolar waves are also decoupled. Therefore, the displacement fields will be expressed as the following modal superposition:

$$u_3 = a_1 e^{I(k_1\,x_3 - \omega t)} + a_2 e^{I(k_2\,x_3 - \omega t)} \tag{3}$$

for longitudinal waves, and

$$u_1 = a_3 e^{I(k_3\, x_3 - \omega t)} + a_4 e^{I(k_4\, x_3 - \omega t)} + a_5 e^{I(k_5\, x_3 - \omega t)} + a_6 e^{I(k_6\, x_6 - \omega t)}$$
$$\theta_2 = R(k_3) a_3 e^{I(k_3\, x_3 - \omega t)} + R(k_4) a_4 e^{I(k_4\, x_3 - \omega t)} + R(k_5) a_5 e^{I(k_5\, x_3 - \omega t)} + R(k_6) a_6 e^{I(k_6\, x_3 - \omega t)} \tag{4}$$

for shear micropolar waves, where a_i are the modal amplitudes and $R(k_i)$ are the ratios between the modal amplitudes b_i/a_i, derived from the solution of the first eigenvalue problem. It should be noted that the in-plane eigenvalue problem, due to the constitutive relationships ([5]) does not depend on the constitutive constants α and β.

Among the solutions (3) and (4), we want to select those that satisfy the free stress conditions on the two upper and lower surfaces of the plate, that is:

$$\sigma_{33} = 0 \quad \sigma_{13} = 0 \quad m_{23} = 0 \quad \text{on} \quad x_3 = \pm h/2 \tag{5}$$

where σ_{33}, σ_{13} are, respectively, the normal and tangent stress components acting on the surface of the plate, and m_{23} is the microcouple inducing rotation about the axis x_2, normal to the plane of the plate; h is the height of the plate. By substituting equations (3) and (4) into equations (5), a second problem of eigenvalues is obtained, whose eigenvalues are the square of the frequencies ω^2 at which oscillations such as those expressed by the tentative solution (2) can occur. These frequencies are called through-thickness frequencies in classical Cauchy continua, and we will extend this nomenclature to the Cosserat media.

The second eigenvalue problem associated with Equations (5) also decouples into longitudinal and shear/microrotation waves. For longitudinal waves, it provides infinite discrete analytical solutions that are:

$$\omega = \frac{\sqrt{\lambda + 2\mu + \kappa}}{h\sqrt{\rho}}\, n\pi \tag{6}$$

where n is an integer number representing the mode order. Unlike Cauchy continua, the through-thickness frequency of longitudinal waveforms also depends on the constitutive coupling parameter κ. The eigenvalues of the shear microrotation waves are obtained numerically. The through thickness frequencies in the range of 0-100 kHz for a sample plate are reported on the vertical axis of Figure 2 as a function of the order of the mode. The red points refer to longitudinal modes, while the blue and cyan points, respectively, refer to shear and micropolar modes. The classification is carried out by observing the components u_1 and θ_2. In shear modes, we have $u_1 = 1, \theta_2 = 0$, while in micropolar modes $u_1 = 0, \theta_2 = 1$. Figure 2 presents a rich spectrum, with several shear/micropolar modes in-between two subsequent longitudinal through-thickness modes.

In the experimental response, we expect that the longitudinal thickness frequencies will dominate over the shear/micropolar thickness frequencies, since the shear vibration will transmit in the air beyond the obstacle only because of viscosity. Moreover, micropolar vibration modes could possibly transmit beyond the obstacle only due to surface irregularities. Based on that, the relationship (6) can be used to determine the material parameters from the observation of the peaks in the transmitted response spectrum. In particular, when longitudinal thickness frequencies are known together with the density and thickness of the plate, by inverting (6), it is possible to determine the combination of material parameters $\lambda + 2\mu + \kappa$.

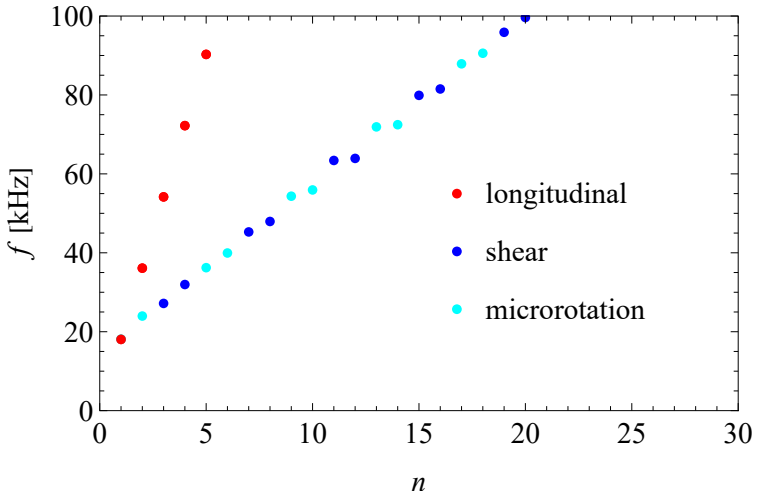


Figure 2: Through-thickness frequencies as a function of the mode order of a sample micropolar plate with the following mechanical and geometrical parameters: $h = 20$ mm, $\lambda = 1.3\,10^7$ N/m^2, $\mu = 2.7\,10^6$N/m^2, $\kappa = 3.0\,10^6$ N/m^2, $\gamma = 1.7\,10^6$N, $\rho = 32$ kg/m^3, $J = 4.010^{-1}$m^2.

5 EXPERIMENTAL METHOD

5.1 Sample characteristics and experimental setup

A plate-shaped expanded polystyrene (EPS) sample was used for acoustic transmission analysis (Fig. 3, left). EPS is widely used for its thermal and acoustic insulation properties in building applications. The sample specifications are summarized in Table 1.

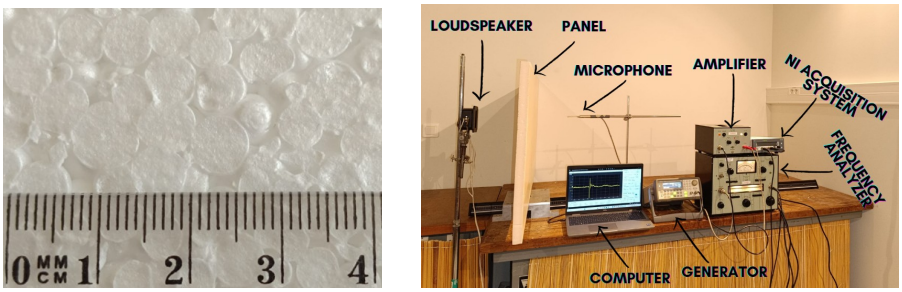


Figure 3: Expanded polystyrene sample (left) and experimental setup (right).

Our ref.	Material	Thickness	Manufacturing process
Plate G	Expanded Polystyrene (EPS)	20 mm	Thermal Expansion

Table 1: Descriptive characteristics of the sample used in this study.

The experimental configuration (Fig. 3, right) ensures normal-incidence plane wave excitation ($\theta = 0°$). A 15 μs pulsed signal, generated by an Agilent 33220A function generator, excites a Visaton MHT 12 loudspeaker. The transmitted wave is detected by a Brüel & Kjær 4138-C-006 microphone, relayed to a National Instruments PXIe-1090 acquisition system. A

Brüel & Kjær 2706 power amplifier stabilizes excitation, while a 2121 frequency analyzer enhances signal fidelity. Temporal acquisitions undergo FFT processing for spectral characterization.

5.2 Experimental results

Fig. 4 illustrates both incident and transmitted signals in the time domain (left), along with their corresponding frequency spectra (right). Significant attenuation of the transmitted signal amplitude is observed, highlighting the sound-absorbing characteristics of the sample. Furthermore, the transmitted signal appears earlier in the time domain compared to the incident wave (as shown in the zoomed-in region of Fig. 4, left). This phenomenon can be attributed to the interaction between the airborne longitudinal wave and the elastic structure of the plate. Upon encountering the sample, the acoustic wave induces vibrations within the material, facilitating the generation and propagation of additional waves within the plate, thereby modifying the effective wave velocity. The spectral content of the incident signal reveals a broadband energy distribution ranging from 2 kHz to approximately 70 kHz, which is particularly suitable for analyzing the fundamental propagation modes.

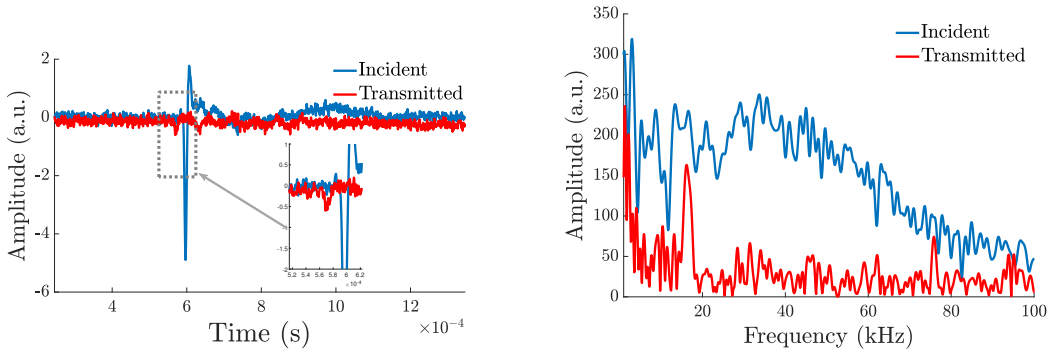


Figure 4: Time-domain representation of the incident and transmitted signals (left) and their corresponding frequency spectra (right) for the 20 mm thick EPS plate.

The magnitude of the transmission coefficient T, shown in Fig. 5, was computed using Matlab®'s `tfestimate` function, applied to the acquired incident and transmitted signals. To mitigate spectral leakage and improve the robustness of the frequency domain estimation, appropriate windowing techniques were implemented during preprocessing [9]. From Fig. 5, it is clear that several peaks that are integer multiples of the first (16.35 kHz) arise, as predicted by Equation 6. In-between adjacent couples of these integer multiples, a rich spectrum appears, which could be attributed to other thickness resonances, as shown in Figure 3. Then, on knowing the density of the plate and its thickness, based on the first peak of the transmission spectrum, inversion of Equation 6 provides $\lambda + 2\mu + \kappa = 1.3\,10^7$ N/m^2.

6 CONCLUSIONS

This study has explored the transmission of acoustic waves through plates made of polystyrene, employing the Cosserat theory of elasticity. By incorporating microstructural effects, the micropolar model enables the description of additional wave modes that cannot be captured by classical elasticity theories. The theoretical formulation led to a system of coupled differential equations that govern both displacement and microrotation fields. Analysis of through-

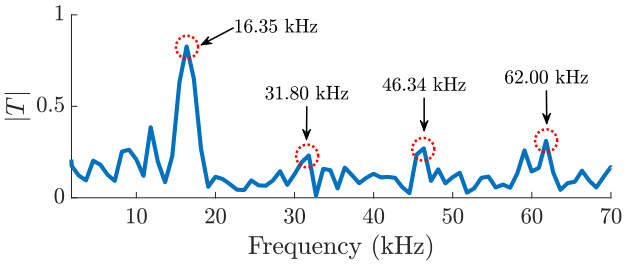


Figure 5: Experimental magnitude of the transmission coefficient for the 20 mm thick EPS plate.

thickness frequencies demonstrated that, unlike classical elasticity, longitudinal resonance frequencies depend on the micropolar coupling parameter κ. In addition, the study identified shear and microrotation modes, highlighting the role of microinertia in wave propagation. The experimental measurements, conducted over a broad frequency range (2 kHz – 70 kHz), confirmed the presence of predicted resonance peaks. The comparison between theoretical and experimental results allowed the identification of a combination of material parameters, specifically $\lambda + 2\mu + \kappa$, demonstrating the impact of micropolar elasticity in low-density polymeric materials.

7 ACKNOWLEDGMENTS

Annamaria Pau acknowledges the funding contribution from Sapienza grant No. 6023149 and from the National Center on HPC, Big Data and Quantum Computing CN00000013. This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

References

[1] W. Voigt. Theoretische studien über die elastizitätsverhältnisse. *Abhandlungen der Akademie der Wissenschaften in Göttingen*, 34:48–55, 1887.

[2] P. Trovalusci, D. Capecchi, and G. Ruta. Genesis of the multiscale approach for materials with microstructure. *Archive of Applied Mechanics*, 79:981–997, 2009.

[3] E. Cosserat and F. Cosserat. *Théorie des corps déformables*. A. Hermann et fils, 1909.

[4] A. C. Eringen. Linear theory of micropolar elasticity. *Journal of Mathematics and Mechanics*, pages 909–923, 1966.

[5] V. R. Parfitt and A. Cemal Eringen. Reflection of plane waves from the flat boundary of a micropolar elastic half-space. *Journal of the Acoustical Society of America*, 45:1258–1272, 1969. URL `https://api.semanticscholar.org/CorpusID:119509404`.

[6] A. Pau and P. Trovalusci. A multifield continuum model for the description of the response of microporous/microcracked composite materials. *Mechanics of Materials*, 160, 2021.

[7] E. Ogam, Z.E.A. Fellah, M. Fellah, and C. Depollier. Theoretical and experimental study of micropolar elastic materials using acoustic waves in air. *Journal of Sound and Vibration*, 510:116298, 2021. URL `https://api.semanticscholar.org/CorpusID:237666206`.

[8] C. Meso, H. Franklin, E. Ogam, and Z.E.A. Fellah. Observation of micropolar modes in the transmission of acoustic waves at oblique incidence by polystyrene plates. *The Journal of the Acoustical Society of America*, 156 4:2606–2618, 2024. URL https://api.semanticscholar.org/CorpusID:273402925.

[9] The MathWorks. *MATLAB version: 9.13.0 (R2022b)*. The MathWorks Inc., Natick, Massachusetts, United States, 2022. URL https://www.mathworks.com.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

SECTIONAL HOMOGENIZATION WITH A GENERAL NONLINEAR CONSTITUTIVE LAW FOR CORRUGATED BOARD ANALYSIS

Tomasz Garbowski[1], **Anna Szymczak-Graczyk**[2], **and Aram Cornaggia**[3]

[1] Department of Biosystems Engineering, Poznan University of Life Sciences
Wojska Polskiego 50, 60-627 Poznań, Poland
e-mail: tomasz.garbowski@up.poznan.pl

[2] Department of Construction and Geoengineering, Poznan University of Life Sciences
Piątkowska 94E, 60-649 Poznań, Poland
anna.szymczak-graczyk@up.poznan.pl

[3] Department of Engineering and Applied Sciences, Università degli studi di Bergamo
viale G. Marconi 5, 24044 Dalmine (BG), Italy
aram.cornaggia@unibg.it

Abstract

Corrugated board is widely used in packaging applications due to its favorable mechanical properties and cost-effectiveness. This study proposes a two-step, multiscale, sectional homogenization approach to determine the effective elastic parameters and predict the mechanical behavior of cardboard under various loading conditions. In the first step, a representative 3D periodic model of corrugated board is developed to extract its effective elastic properties. This homogenization process reduces the complex 3D structure to a simplified 2D shell model. The second step focuses on computing deformations, strains, and stresses in cardboard structures. Instead of using a conventional plastic constitutive model, the 3D Representative Volume Element (RVE) model is employed. Strains from the 2D model are applied as boundary conditions to the 3D RVE, where corresponding stresses are determined based on pre-calibrated strain-stress relationships obtained from uniaxial tests. Stiffness degradation is captured, from a computational viewpoint, at Gauss points in the 2D model. The results demonstrate that the proposed approach accurately reproduces the real mechanical behavior of cardboard while utilizing a simplified 2D model.

Keywords: Corrugated Board, Sectional Homogenization, Nonlinear Material Modelling, Inverse Representative Volume Element Evaluation, Composite Materials and Structures.

1 INTRODUCTION

Corrugated cardboard, invented in the late 19th century, has become one of the most widely used structural materials in the packaging industry [1, 2]. Its success lies in the unique combination of low weight, high stiffness, cost-efficiency, and recyclability. Composed of one or more corrugated core layers (fluting), sandwiched between flat linerboards, such structure provides resistance to bending, compression, and impact, while using minimal raw material. These characteristics have made corrugated board indispensable, not only in transport packaging, but also in furniture design, temporary structures, and architectural prototypes (see, e.g., [3, 4, 5]).

Despite its widespread use, an accurate modeling of the mechanical behavior of corrugated board remains a challenging task. Its geometry is highly heterogeneous and periodic, exhibiting nonlinear responses under realistic loading conditions, including localized buckling, delamination, and stiffness degradation. Standard computational methods often fail to efficiently capture these effects.

Homogenization techniques have long been used to approximate the macroscopic behavior of materials with periodic microstructures. Classic methods, such as asymptotic homogenization, periodic cell analysis, and numerical Representative Volume Element (RVE) approaches, allow the reduction of complex 3D problems to simplified 2D models [6, 7]. These methods are especially effective when analyzing large-scale structures, where solving the full geometry would be computationally infeasible.

In the case of corrugated cardboard, numerical homogenization has been successfully applied to estimate effective elastic parameters, including in-plane moduli and transverse shear stiffness [8]. Previous research by Garbowski and Gajewski [9] proposed a Finite Element-based homogenization approach suitable for extracting the transverse shear stiffness of sandwich panels with corrugated cores. However, such methods are typically limited to the elastic regime and do not account for local instability phenomena, such as localized buckling of liners in unconstrained regions [10].

In order to address these limitations, in the present research contribution a novel sectional, two-step multiscale homogenization method has been developed. This approach divides the full 3D periodic unit into distinct segments (sections) based on their mechanical role and local behavior. Each section is then independently homogenized, allowing for more accurate modeling of nonlinear effects, such as, e.g., local buckling of unsupported liner segments.

The proposed method also incorporates a general nonlinear constitutive law, calibrated from uniaxial tests, which is used in the second step of the procedure. Consistently, strain values obtained from a simplified 2D shell model are applied as boundary conditions to the local 3D RVE sections. The stress response, including stiffness degradation, is then assessed and mapped back into the global model. This workflow enables realistic simulations of mechanical behavior, even in the post-elastic range, while maintaining computational efficiency [11].

This work builds upon the broader field of multiscale modeling, where local RVEs are embedded within global continuum models, often in the form of shells or plates [12, 13]. Such methods have been proven effective in composite structures and advanced sandwich panels, and their extension to the modeling of corrugated board opens new possibilities for structural optimization, durability analysis, and virtual prototyping, among the aims of the present contribution.

Following the current introduction and brief literature survey, the article is organized in three main sections. Section 2 is devoted to the development of the two-step computational method, both regarding constitutive modelling and computational features. Consequently, Section 3 presents a numerical validation example, also providing relevant discussion about

the results and the devised methodology in Section 4. Closing remarks are gathered in an additional section, to highlight the main outcome and the innovative key aspects.

2 MATERIALS AND METHOD

2.1 Sectional two-step homogenization

The proposed method is based on the concept of sectional two-step homogenization. In the first step, effective elastic properties of the corrugated board are determined using a Representative Volume Element modeled in 3D. Unlike classic approaches that consider a single periodic unit, here, the periodic unit is divided into mechanically distinct sections, reflecting different local behaviors under load. These sections may include:

- compressed linerboard segments that are either supported or unsupported by the fluting;
- regions transmitting shear between liners and the corrugated core;
- zones prone to local or global buckling.

Such segmentation allows the method to capture nonlinear effects that are typically averaged out, in traditional homogenization.

Each defined section is separately homogenized, and the resulting properties are assigned to Gauss points in the simplified 2D shell model. This 2D model reproduces the overall geometry of the real structure (e.g., a packaging wall), assuming that each point exhibits different effective stiffness properties depending on the underlying section of the corrugated geometry.

The general structure of the method is illustrated in Figure 1, which outlines the complete workflow, from the construction of the 3D geometry and sectional RVE identification, through homogenization, to the strain-to-stress mapping in the simplified model.

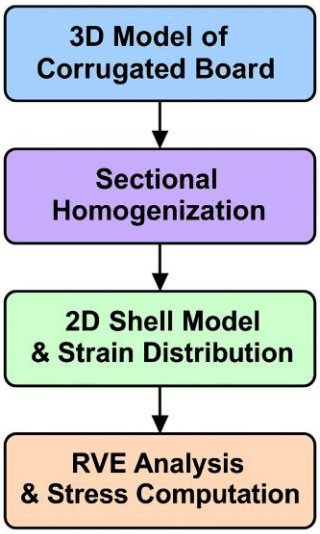


Figure 1. Flowchart illustrating the proposed two-step sectional homogenization methodology. The workflow includes 3D geometry modeling, sectional homogenization, 2D shell modeling, and reverse mapping through pre-calibrated RVE responses.

2.2 Inverse constitutive calibration

In traditional finite element simulations, especially those involving shell or plate elements, nonlinear behavior is typically introduced through built-in constitutive models, such as elasto-

plasticity or damage mechanics. However, these models operate at the macroscopic generalized level and require global parameters that are often difficult to calibrate for complex heterogeneous, possibly anisotropic, materials like corrugated cardboard. Moreover, when homogenization is applied in the first step, information about the material microstructure is typically lost, and, with it, its local instability mechanisms.

To overcome this limitation, the second step of the current devised approach uses a non-standard constitutive procedure based on inverse RVE evaluation. Instead of assuming a predefined constitutive model for the 2D shell elements, a reverse projection technique is employed: local strain tensors computed at Gauss integration points, in the 2D model, are imposed as boundary conditions on the corresponding 3D RVE sections.

To this aim, it is worth to note that, each RVE has been previously calibrated using either:

- laboratory uniaxial tests (e.g., compressive or tensile tests in Machine Direction/Cross-machine Direction), or
- detailed numerical simulations for each RVE section under representative loading conditions.

From these, nonlinear strain-stress curves are obtained and stored as lookup tables or interpolation functions. These curves represent how each section responds to deformation, also accounting for effects such as: (a) stiffness softening due to local damage or buckling; (b) nonlinear stress redistribution in post-elastic ranges; (c) anisotropic behavior, if present in liners or fluting.

In the simulation workflow, for each Gauss point: (i) the local strain tensor ($\boldsymbol{\varepsilon}$) is computed in the 2D shell element; (ii) the point is associated with a specific RVE section (e.g., compressed unsupported liner); (iii) the corresponding RVE model is loaded, and $\boldsymbol{\varepsilon}$ is applied as a boundary condition; (iv) the stress tensor ($\boldsymbol{\sigma}$) is computed based on pre-calibrated nonlinear response curves.

The stress is then transferred back to the 2D shell model, replacing what would otherwise be computed via a simple constitutive law. This decouples the geometry from the constitutive behavior while retaining local mechanical realism. It also allows for heterogeneous nonlinear responses within a single shell element, which is especially valuable for structures exhibiting localized phenomena such as:

- folding or crushing in unsupported liner zones,
- tension stiffening in stretched regions,
- shear-lag effects between layers.

Importantly, this procedure preserves computational efficiency: the 3D RVEs are not solved in real time, but their responses are interpolated from precomputed data, making the approach suitable for large-scale simulations. Moreover, it can be extended to time-dependent behavior (e.g., creep or cyclic fatigue) by enriching the database of RVE responses.

3 NUMERICAL EXAMPLE

3.1 Three-point bending test

To evaluate the accuracy and predictive capability of the proposed sectional homogenization method, we performed a numerical simulation of a classic three-point bending test, on a specimen made of single-wall corrugated cardboard. This test setup is commonly used in experimental studies to evaluate flexural stiffness and to observe failure mechanisms such as local buckling, delamination, or crushing. A schematic representation of the experimental configuration is presented in Figure 2.

The tested specimen had the following dimensions: length 200 mm, width 50 mm, and total thickness 4 mm (including two liners and a sinusoidal fluting). The sample was simply supported at both ends with a hinge and a roller boundary condition, respectively. A concentrated vertical force was applied at midspan using a rigid cylindrical indenter. The loading process was displacement-controlled, to facilitate detailed observation of post-buckling behavior.

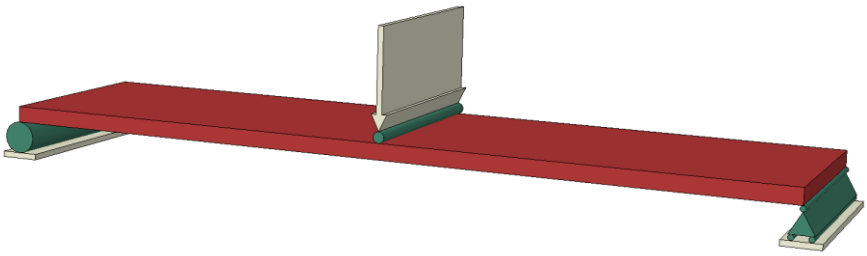
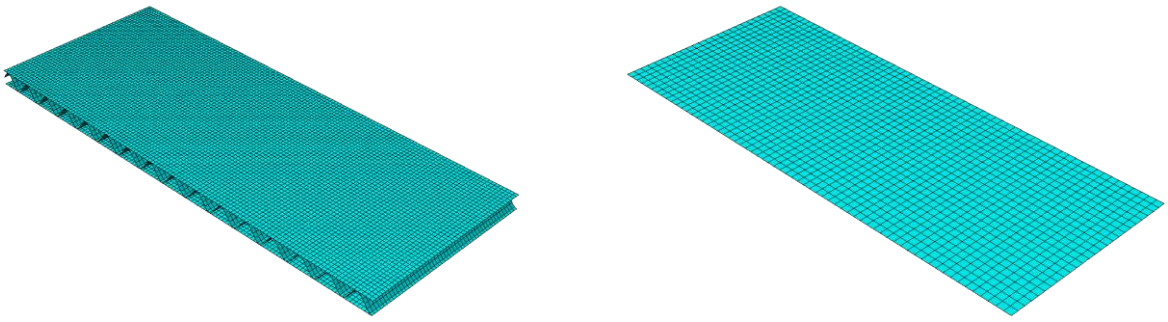


Figure 2. Experimental layout of the three-point bending test: a corrugated board sample is placed on two supports with a concentrated load applied in the center.

To investigate the efficiency and fidelity of the proposed method, two numerical models were constructed:

Model A – Full 3D geometry (Figure 3a). This model explicitly reproduces the detailed geometry of the corrugated core and linerboards. The fluting was modeled using four-node fully integrated shell elements, discretized to follow the actual sinusoidal profile. Both linerboards were represented by layered shell elements, co-meshed with the fluting at contact nodes. Material orthotropy was defined in accordance with Machine Direction (MD) and Cross-machine Direction (CD), and nonlinear behavior, including local buckling, was captured through geometric nonlinearity and material softening laws.

Model B – 2D shell with sectional RVEs (Figure 3b). In the simplified model, the entire structure was represented using a single-layer shell mesh with effective elastic properties derived from the first step of the sectional homogenization procedure. However, the key innovation lies in how material behavior is assigned: each Gauss point in the shell mesh is linked to a pre-defined sectional RVE, representing specific geometric and mechanical conditions (e.g., compressed liner above a flute, or in unsupported regions). During simulation, local strains at each Gauss point are used to retrieve nonlinear stresses via calibrated strain-stress curves specific to that RVE. This enables the model to replicate local stiffness degradation and post-buckling behavior, while maintaining a significantly reduced computational cost.



(a) (b)

Figure 3. Numerical models used in the analysis: (a) half of the full 3D model (with 22,050 shell 4-node elements) representing the actual geometry of the corrugated structure with shell elements for fluting and liners; (b) half of the simplified 2D shell model (with 1 575 shell 4-node elements) using sectional RVEs to represent local nonlinear behavior.

This comparison aimed to verify whether the simplified 2D model could reproduce the complex mechanical response observed in the 3D simulation, especially: (a) stiffness evolution in the elastic and nonlinear ranges; (b) initiation and development of local buckling in compressed linerboard zones; (c) global deformation patterns, including deflection shape and symmetry.

4 RESULTS AND DISCUSSION

4.1 Observations and phenomena

In the full 3D simulation (Model A), local buckling of the bottom linerboard was observed shortly after the linear range, particularly: (i) directly under the point of load application and (ii) near the supports, where the liner was not adequately supported by the fluting.

This is a well-known weakness in corrugated structures and typically leads to nonlinear softening, stress redistribution, and in some cases, delamination or folding.

The 2D shell model (Model B), despite lacking explicit geometric detail, was able to faithfully reproduce the onset of buckling and subsequent stiffness reduction due to the presence of sectional RVEs. This confirms that the method captures not only average stiffness, but also localized failure mechanisms.

4.2 Force-displacement comparison

Figure 4 presents the force-displacement curves obtained from both numerical models, the full 3D model and the simplified 2D shell model with sectional RVE evaluation. The horizontal axis represents the vertical displacement at the midspan of the sample, while the vertical axis corresponds to the loading reaction force. Both curves exhibit similar initial stiffness, indicating that the homogenized elastic parameters used in the 2D model were correctly extracted. A slight divergence appears after the initiation of local buckling, as the 3D model begins to experience geometric softening.

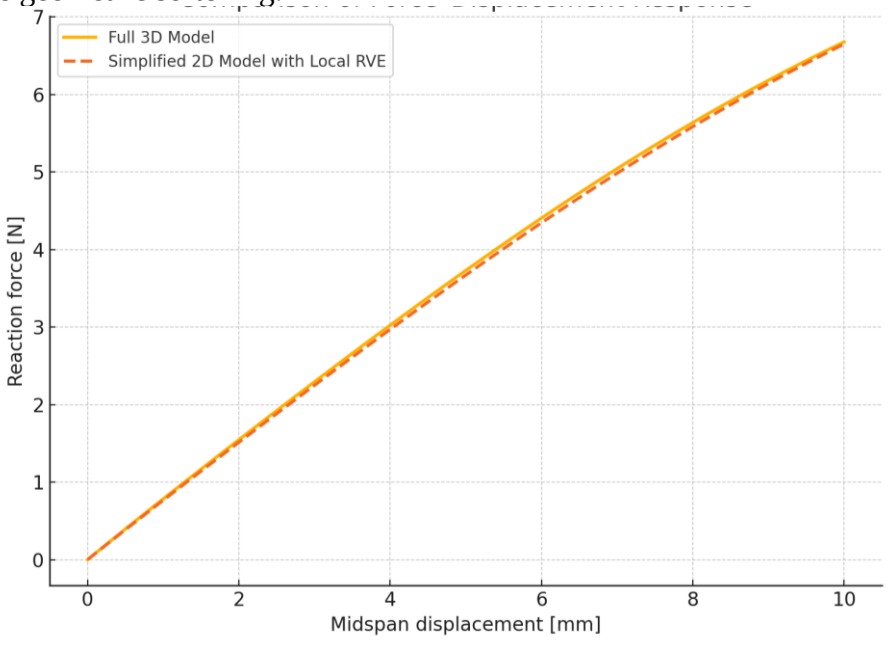


Figure 4. Force-displacement response comparison for the full 3D model and the simplified 2D shell model with local RVEs under three-point bending.

Nevertheless, the 2D model closely reproduces the nonlinear response, including the post-buckling stiffness degradation. The ability of the simplified model to capture this behavior confirms that the calibrated sectional RVEs effectively encode the complex local instabilities observed in the real geometry.

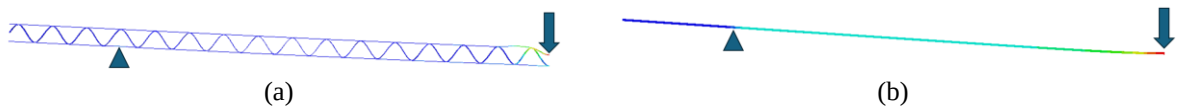
4.3 Stress distribution and local effects

To better understand how each model captures local failure mechanisms, Figure 5 presents the stress field distribution (component σ_{xx}) in the bottom liner at selected load levels in both the 3D and 2D simulations.

In the 3D model (Figure 5a), clear stress concentrations are visible: (i) beneath the point load, where the liner is unsupported by the fluting, (2) and near the supports, due to bending-induced compression.

These zones coincide with regions of local buckling, observable as out-of-plane deformations in the compressed linerboard. The material, in these areas, exhibits a sudden drop in stiffness, initiating nonlinear behavior and global softening.

In the 2D shell model (Figure 5b), although the geometry does not explicitly model the fluting and liner layers, the same critical zones show reduced stress-carrying capacity due to the RVE-based stiffness degradation. This confirms that the proposed method not only reproduces global response, but also identifies local failure-prone regions, even with significantly reduced model complexity.



(a) (b)

Figure 5. Stress distribution (σ_{xx}) and local effects at supports and midspan: (a) half of the 3D model showing concentration zones and buckling in unsupported liners, (b) half of the 2D model showing analogous softening via calibrated RVE behavior.

5 CONCLUSIONS

This study introduced a novel, two-step, sectional homogenization framework for modeling the nonlinear mechanical behavior of corrugated cardboard structures. The method combines the efficiency of simplified 2D shell models with the accuracy and realism of local 3D behavior, as captured through sectional Representative Volume Elements. Its foundation lies in dividing the geometrically periodic unit cell into mechanically distinct regions, each of which undergoes separate homogenization and nonlinear calibration. This allows for accurate modeling of local phenomena, such as liner buckling, stiffness degradation, and stress redistribution, phenomena often neglected or overly smoothed in classic homogenization approaches.

The main conclusions of this work are as follows.

1. The proposed method successfully integrates detailed local behavior into global simulations, allowing simplified shell models to capture highly nonlinear and localized effects typical of corrugated cardboard under compressive and bending loads.
2. Sectional homogenization offers a mechanically meaningful partitioning of the Representative Volume Element, enabling finer resolution of structurally critical areas, such as unsupported liner segments or regions near concentrated loads and supports.
3. The inverse use of RVEs, where strain fields from the 2D model are imposed on pre-calibrated 3D sections, allows recovery of accurate stress fields without relying

on conventional constitutive laws at the macroscale. This increases the predictive power of the simulation, particularly in the post-elastic regime.

4. The method was validated through a three-point bending test simulation. The 2D model, although lacking explicit geometry, replicated the behavior of the full 3D model with impressive accuracy. Both global and local responses, including force-displacement curves, buckling zones, and stress concentrations, were effectively reproduced.
5. The approach is computationally efficient, enabling large-scale structural simulations with realistic post-elastic effects, while drastically reducing model complexity, computational burden and solving time.

The methodology presented here lays a robust foundation for further extensions in both research and industrial practice. Ongoing and future developments will focus on:

- automatic generation and classification of sectional RVEs for arbitrary geometries and boundary conditions, based on machine learning or pattern recognition approaches (see, e.g., [14]);
- application to multi-layered corrugated systems, such as double-wall and triple-wall boards, and to more complex load states, including shear-dominated and biaxial scenarios;
- extension of the RVE calibration to time-dependent effects, including creep, fatigue, and environmental degradation due to relative humidity and temperature fluctuations (see, e.g., [15]);
- integration with design and optimization frameworks, enabling shape and thickness optimization, material selection, and durability-driven packaging assessment and design.

The devised framework also gains the potential to be generalized beyond corrugated cardboard, to other engineered materials with heterogeneous, layered, or architected microstructures (see, e.g., [16, 17, 18])—where mechanical behavior is driven by complex local interactions that cannot be captured by homogenized continuum models alone—for an effective assessment and modeling methodology.

REFERENCES

[1] J.W. Simon, A review of recent trends and challenges in computational modeling of paper and paperboard at different scales. *Archives of Computational Methods in Engineering*, **28**, 2409-2428, 2020.

[2] D.W. Coffin, Historical perspectives of corrugated box testing. *TAPPI journal*, **19**, 161-173, 2020.

[3] B. Suarez, M.L.M. Muneta, J.D. Sanz-Bobi, G. Romero, Application of homogenization approaches to the numerical analysis of seating made of multi-wall corrugated cardboard. *Composite Structures*, **262**, 113642, 2021.

[4] F.M. Di Russo, M.P. Desole, A. Gisario, A. Genovesi., M. Barletta, Study and Design of Corrugated Cardboard Trays With Microwaves by Experimental Analysis (EA) and Finite Element Methods (FEM). *Packaging Technology and Science*, **37**, 857-874, 2024.

[5] A. Cornaggia, D. Mrówczyński, T. Gajewski, A. Knitter-Piątkowska, T. Garbowski, Advanced Numerical Analysis of Transport Packaging. *Applied Sciences*, **14**, 11932, 2024.

[6] E. Sanchez-Palencia, *Non-Homogeneous Media and Vibration Theory*. Springer-Verlag, 1980.

[7] M.G.D. Geers, V.G. Kouznetsova, W.A.M. Brekelmans, Multi-scale computational homogenization: Trends and challenges. *Journal of Computational and Applied Mathematics*, **234**, 2175-2182, 2010.

[8] Y.K. Suh, H.G. Park, Homogenization and analysis of corrugated board structures using numerical RVE approach. *Composite Structures*, **66**, 423-432, 2004.

[9] T. Garbowski, T. Gajewski, Determination of transverse shear stiffness of sandwich panels with a corrugated core by numerical homogenization. *Materials*, **14**, 1976, 2021.

[10] T. Garbowski, T. Gajewski, J.K. Grabski, The role of buckling in the estimation of compressive strength of corrugated cardboard boxes. *Materials*, **13**, 4578, 2020.

[11] N. Staszak, T. Garbowski, B. Ksit, Application of the generalized nonlinear constitutive law in numerical analysis of hollow-core slabs. *Archives of Civil Engineering*, **68**, 125-145, 2022.

[12] E.W.C. Coenen, V.G. Kouznetsova, M.G.D. Geers, Multi-scale modeling of discontinuities in heterogeneous materials. *International Journal for Numerical Methods in Engineering*, **92**, 963-994, 2012.

[13] K. Terada, M. Hori, Simulations of the multi-scale convergence in computational homogenization. *International Journal of Solids and Structures*, **40**, 7329-7353, 2003.

[14] M. Rogalka, J.K. Grabski, T. Garbowski, Identification of Geometric Features of the Corrugated Board Using Images and Genetic Algorithm. *Sensors*, **23**, 6242, 2023.

[15] A. Cornaggia, T. Gajewski, A. Knitter-Piątkowska, T. Garbowski, Influence of Humidity and Temperature on Mechanical Properties of Corrugated Board-Numerical Investigation. *BioResources*, **18**, 7490-7509, 2023.

[16] D. Savvas, G. Stefanou, Determination of random material properties of graphene sheets with different types of defects. *Composites Part B: Engineering*, **143**, 47-54, 2018.

[17] N. Staszak, A. Szymczak-Graczyk, T. Garbowski, Elastic Analysis of Three-Layer Concrete Slab Based on Numerical Homogenization with an Analytical Shear Correction Factor. *Applied Sciences*, **12**, 9918, 2022.

[18] N. Fantuzzi, A. Vidwans, A. Dib, P. Trovalusci, J. Agnelli, A. Pierattini, Flexural characterization of a novel recycled-based polymer blend for structural applications. *Structures*, **57**, 104966, 2023.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

ACTIVE LEARNING ENHANCED STOCHASTIC SIZE-DEPENDENT DYNAMIC ANALYSIS OF THE PEROVSKITE SOLAR CELL

Luo Bo, and Jize Zhang*

Department of Civil and Environmental Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong
First author e-mail: boluo@ust.hk

*Corresponding author e-mail: cejize@ust.hk

Abstract

The perovskite solar cell (PSC), recognized as a revolutionary advancement in next-generation photovoltaic devices, has emerged as one of the fastest-growing market competitors, characterized by its inherent micro-scale composite thickness. Investigating multi-scale vibrations of the PSC is a critical task for the overall dynamic system, as unfavourable responses can be minimized to the greatest extent possible. Furthermore, various physically inspired uncertainties inherently exist in practical engineering implementations. As a result, free vibration analysis that employs a deterministic model while neglecting intrinsic randomness is merely an approximation of realistic structural behaviour. Therefore, this research represents the first attempt to perform a stochastic size-dependent dynamic analysis of the PSC, taking into account multi-dimensional uncertainties with diverse distributions. An active learning technique, specifically the sequential design of experiment enhanced adaptive Kriging (SDOE-AKriging), is implemented to capture the underpinned and complex relationship between intrinsic uncertainties and the concerned size-dependent free vibration behaviour. Comprehensive statistical information regarding composite dynamic responses, encompassing the mean, standard deviation, coefficient of variation, probability density function, and cumulative distribution function, is furnished for structural safety assessment and reliability-based design. Numerical experiments demonstrate the superiority of the SDOE-AKriging over existing meta-model workhorses. Besides, the SDOE-AKriging is capable of delivering accurate vibration response forecasts for the PSC under rapidly changing conditions with tremendously alleviated computational toll, inspiring rapid information updates and timely dynamic behaviour monitoring in real engineering scenarios.

Keywords: Perovskite Solar Cell, Stochastic Size-dependent Vibration Analysis, Multi-scale Dynamics, Uncertainty Quantification, Active Learning.

1 INTRODUCTION

The perovskite solar cell (PSC) with micro-scale thickness has instigated a remarkable technological advancement in third-generation photovoltaics, thanks to its favourable traits in easy production, high solar absorption efficiency, and plentiful raw materials [1–3]. Investigating the free vibration behaviour of the PSC is a key factor in analyzing the structural mechanical performance in practical applications [4–6]. Whereas, due to its ultra-thin characteristics, the structural properties and dynamic behaviour can become size-dependent [7]. In this context, traditional continuum theory in macro-mechanical analysis, which does not account for scale effects and internal material length scale parameters, may not provide accurate investigation results for composite responses [8]. In addition, numerous physically inspired uncertainties intrinsically exist in dynamic systems, and their effects on structural responses must be scrupulously quantified in composite safety assessment against reliability criteria [9]. Therefore, this work represents the first attempt to perform a stochastic size-dependent dynamic analysis of the micro-scale PSC under multi-dimensional uncertainties with various distributions. The refined shear deformation theory and modified strain gradient theory (MSGT) [10] are employed to investigate the size-dependent response incorporating scale effects. An active learning technique, specifically the sequential design of experiment enhanced adaptive Kriging (SDOE-AKriging) [11], is implemented to depict the underpinned and intricate relationship between input uncertainties and the focused size-dependent dynamic behaviour of the micro-PSC. Comprehensive statistical information regarding composite dynamic responses-including the mean, standard deviation (Std), probability density function (PDF), and cumulative distribution function (CDF)-is provided for structural safety assessment and reliability-based design. Numerical studies demonstrate the advantages of SDOE-AKriging against established meta-modelling tactics. Furthermore, the SDOE-AKriging is capable of offering accurate dynamic response forecasts for the PSC under swiftly varying environments while significantly reducing computational costs, inspiring rapid information updates and timely dynamic behaviour monitoring in real engineering scenarios.

2 SIZE-DEPENDENT FREE VIBRATION ANALYSIS OF THE PEROVSKITE SOLAR CELL

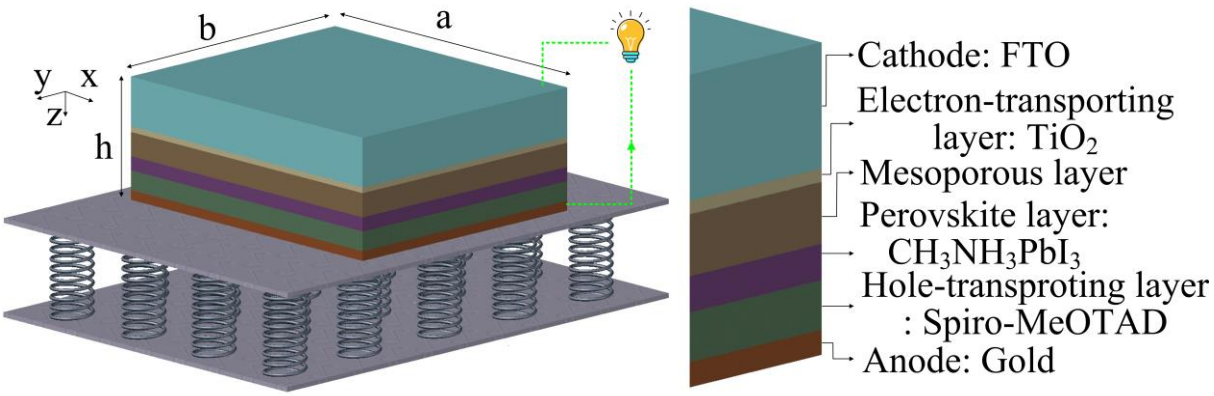


Figure 1: The micro-scale PSC on the Winkler-Pasternak elastic foundation and its architecture.

The architecture of the ultra-thin PSC is exhibited in Figure 1. The refined shear deformation theory [8] is leveraged and expressed as

$$\begin{cases} u_x(x,y,z,t) = u(x,y,t) - z\dfrac{\partial w_b}{\partial x} + (\dfrac{1}{4}z - \dfrac{5z^3}{3h^2})\dfrac{\partial w_s}{\partial x}, \\ u_y(x,y,z,t) = v(x,y,t) - z\dfrac{\partial w_b}{\partial y} + (\dfrac{1}{4}z - \dfrac{5z^3}{3h^2})\dfrac{\partial w_s}{\partial y}, \\ u_z(x,y,z,t) = w_b(x,y,t) + w_s(x,y,t). \end{cases} \tag{1}$$

Considering the inherent micro-scale thickness, the MSGT is implemented to describe the size-dependent effects. The constitutive expressions, which involve the classical stress tensor and higher-order stresses in the MSGT, are formulated as

$$\sigma_{ij} = \lambda\varepsilon_{mm}\delta_{ij} + 2\mu\varepsilon_{ij}, \tag{2}$$

$$p_i = 2\mu l_0^2 \gamma_i, \tag{3}$$

$$\tau_{ijk}^{(1)} = 2\mu l_1^2 \eta_{ijk}^{(1)}, \tag{4}$$

$$m_{ij}^{(s)} = 2\mu l_2^2 \chi_{ij}^{(s)}, \tag{5}$$

where $\eta_{ijk}^{(1)}$, γ_i, and $\chi_{ij}^{(s)}$ indicate the deviatoric stretch gradient tensor, dilatation gradient tensor, and symmetric rotation gradient tensor, respectively. l_1, l_0, and l_2 denote the material length scale parameters associated with the deviatoric gradient, dilatation gradient, and symmetric rotation gradient, respectively.

Implementing the Navier scheme, Hamilton's principle, and Galerkin approach [12,13], the following formulation is obtained

$$([\overline{Kl}] - \omega_{mn}^2[\overline{Ml}])\{\Delta\} = \{0\}, \tag{6}$$

where

$$[\overline{Kl}] = \begin{bmatrix} Kl_{mn}^{11} & Kl_{mn}^{12} & Kl_{mn}^{13} & Kl_{mn}^{14} \\ Kl_{mn}^{21} & Kl_{mn}^{22} & Kl_{mn}^{23} & Kl_{mn}^{24} \\ Kl_{mn}^{31} & Kl_{mn}^{32} & Kl_{mn}^{33} & Kl_{mn}^{34} \\ Kl_{mn}^{41} & Kl_{mn}^{42} & Kl_{mn}^{43} & Kl_{mn}^{44} \end{bmatrix}, [\overline{Ml}] = \begin{bmatrix} Ml_{mn}^{11} & Ml_{mn}^{12} & Ml_{mn}^{13} & Ml_{mn}^{14} \\ Ml_{mn}^{21} & Ml_{mn}^{22} & Ml_{mn}^{23} & Ml_{mn}^{24} \\ Ml_{mn}^{31} & Ml_{mn}^{32} & Ml_{mn}^{33} & Ml_{mn}^{34} \\ Ml_{mn}^{41} & Ml_{mn}^{42} & Ml_{mn}^{43} & Ml_{mn}^{44} \end{bmatrix}. \tag{7}$$

The deterministic natural frequencies of the micro-PSC can be computed by resolving this eigenvalue task.

3 STOCHASTIC SIZE-DEPENDENT DYNAMIC ANALYSIS AND ACTIVE LEARNING TECHNIQUE

Taking into account the inherent physically inspired multi-dimensional uncertainties $\boldsymbol{\theta}_r$, the probabilistic eigenvalue equation in free vibration analysis represents

$$\{[\overline{Kl}(\boldsymbol{\theta}_r)] - \omega_{mn}^2[\overline{Ml}(\boldsymbol{\theta}_r)]\}\{\Delta\} = \{0\}. \tag{8}$$

For the above expression, classical analytical or mathematical schemes are nearly infeasible for this size-dependent stochastic dynamic problem. Hence, an effective active learning strategy, namely, the sequential design of experiment enhanced adaptive Kriging [11], is introduced to seize the sophisticated relationship between manifold multi-dimensional uncertainties and the concerned size-dependent free vibration behaviour of the PSC with scale

effects, while furnishing sufficient statistical information on the composite response. The main components of the SDOE-AKriging encompass the Kriging meta-model, robust integrated mean squared error target function geared toward sequential experimental design, and advisable self-adjusting covariance matrix adaptation evolution strategy for addressing complex challenges in selecting optimal experiments and tuning hyperparameters.

4 NUMERICAL STUDIES

4.1 Verification

The free vibration behaviour of the PSC is examined and validated against published literature and commercial software (Table 1). As shown in the Table, the results of the presented work align well with established studies, confirming the accuracy of the developed dynamic analysis.

Mode(m,n)	Presented	[4]	ABAQUS	Max |relative difference|
(1,1)	525.7068	525.7047	525.7524 (1st)	0.008673%
(1,2)	1314.2659	1314.2697	1314.7312 (2nd)	0.03539%
(1,3)	2628.5280	2628.6030	2630.9904 (5th)	0.09359%
(2,3)	3417.0834	3417.0566	3419.4440 (7th)	0.06903%
(1,4)	4468.4885	4468.4342	4476.5511 (9th)	0.1801%

Table 1: Comparisons of natural frequencies of the PSC (Hz).

4.2 Numerical experiments

The physically inspired manifold uncertainties are incorporated and tabulated in Table 2. The crude Monte Carlo simulation (MCS) is performed with 10000 cycles to generate reference results for the stochastic size-dependent free vibration analysis of the PSC with scale effects. Figure 2 illustrates the accuracy improvement rates of different surrogate models as the training sample sizes increase. By virtue of the integrated active learning technique, the presented SDOE-AKriging demonstrates a superior accuracy enhancement speed compared to classical machine learning algorithms.

Random variables	E_{Au} (GPa)	$E_{\text{Spiro-MeOTAD}}$ (GPa)	$\nu_{\mathrm{Perovskite}}$	$E_{\mathrm{Perovskite}}$ (GPa)	$E_{\mathrm{Mesoporous}}$ (GPa)	$\rho_{\mathrm{TiO_2}}$ (kg/m^3)	$\rho_{\mathrm{TiO_2}}$ (kg/m^3)	k_w (N/m^3)
Distribution	Lognormal	Lognormal	Beta	Normal	Lognormal	Normal	Normal	Normal
Mean	78	15	0.33	17.8	210.44	4230	6850	2000
Std	7.8	1.5	0.013	1.068	21.044	846	411	200

Table 2: Statistical information of inspected random variables in the micro-PSC.

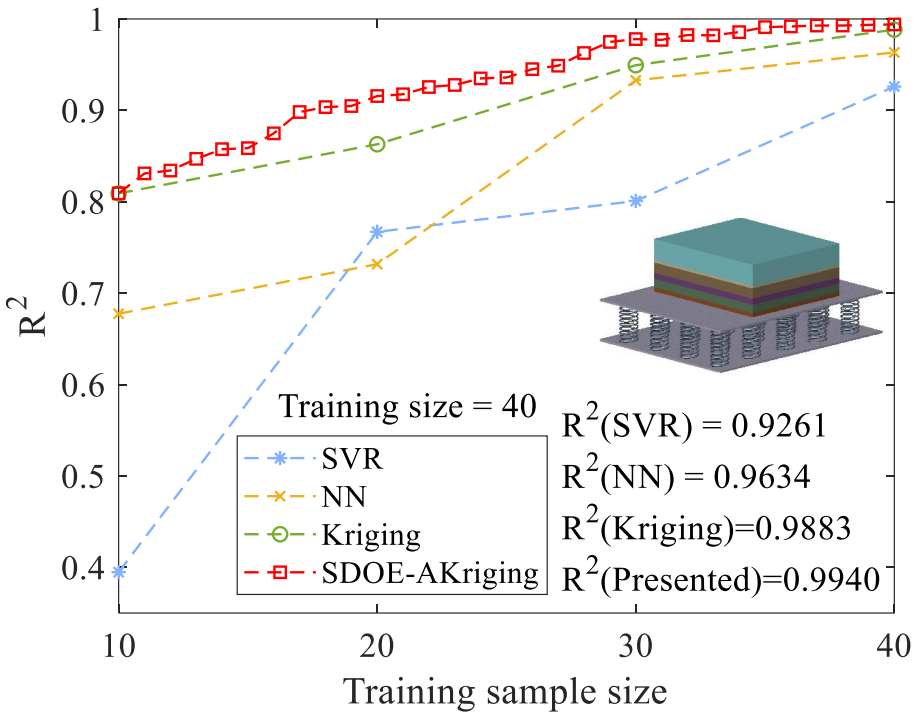


Figure 2: The R^2 of predicted size-dependent fundamental natural frequencies of the micro-PSC under different training sample sizes.

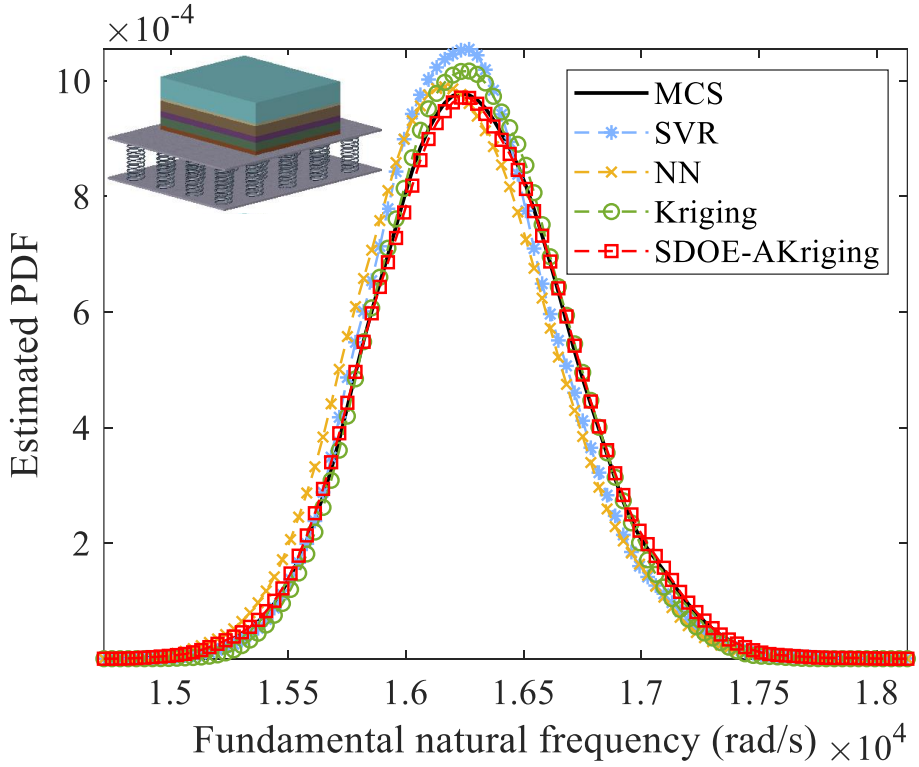


Figure 3: The estimated PDF of size-dependent fundamental natural frequencies of the micro-PSC.

The quantified statistical information of the PDF, CDF, and statistical moments of the mean and Std are manifested in Figures 3 and 4, as well as Table 3. By sequentially selecting a total of 40 optimal experiments, the presented SDOE-AKriging is competent to deliver ac-

curately quantified statistical diagrams and moments of micro-composite dynamic behaviour, which match well with the brute-force MCS references. The higher R^2 value, superior Kolmogorov-Smirnov (K-S) statistic, and smaller relative errors in the mean and Std confirm the significant advantages of the SDOE-AKriging over existing meta-models. Furthermore, the computation costs of the MCS and presented active learning technique are 98.61min and 4.176min, respectively, implying the compelling efficiency of the proposed method.

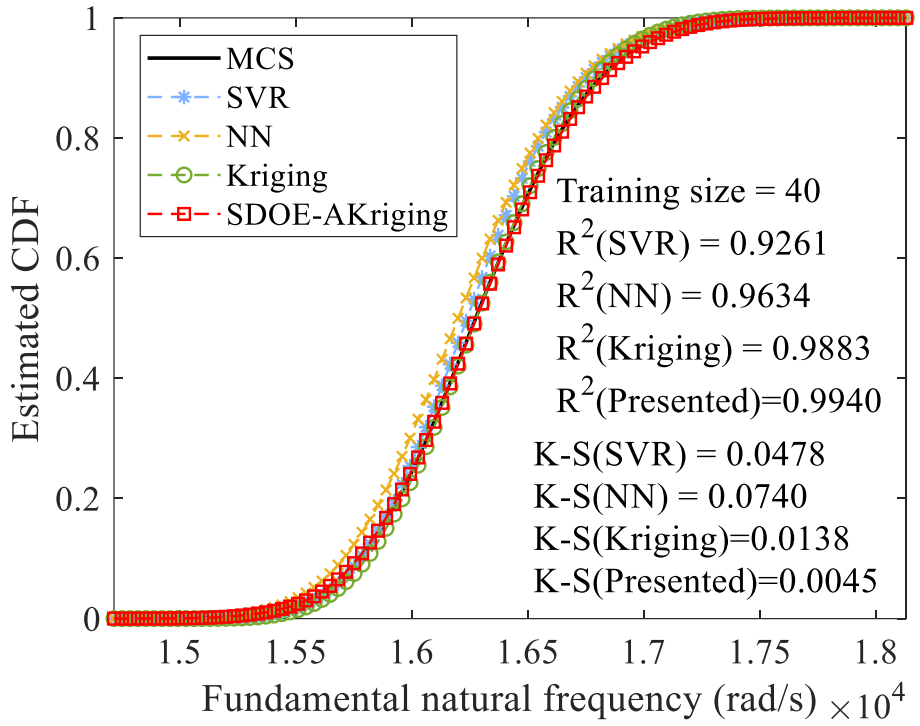


Figure 4: The estimated CDF of size-dependent fundamental natural frequencies of the micro-PSC.

Moment	Method	Fundamental natural frequency (rad/s)	Relative error against MCS(%)
Mean	SVR	1.6252×10^4	-0.2088%
	NN	1.6210×10^4	-0.4667%
	Kriging	1.6288×10^4	0.01228%
	SDOE-AKriging	1.6287×10^4	0.006140%
	MCS	1.6286×10^4	--
Std	SVR	3.7702×10^2	-5.5940%
	NN	3.9519×10^2	-1.0442%
	Kriging	3.7366×10^2	-6.4353%
	SDOE-AKriging	4.0236×10^2	0.7512%
	MCS	3.9936×10^2	--

Table 3: The estimated moments of size-dependent fundamental natural frequencies of the micro-PSC.

Moreover, the previously established SDOE-AKriging inherently possesses the merit of rapid information updates. Two arbitrary new sets of micro-PSC properties (shown in Table 4) are examined outside the original variation range. The results in Table 5 demonstrate the effectiveness and precision of the presented active learning technique in swift information up-

dates under rapidly changing conditions, facilitating immediate composite decision-making in in real engineering scenarios.

Property	E_{Au} (GPa)	$E_{Spiro\text{-}MeOTAD}$(GPa)	$\nu_{Perovskite}$	$E_{Perovskite}$ (GPa)	$E_{Mesoporous}$ (GPa)	ρ_{TiO_2} (kg/m^3)	ρ_{TiO_2} (kg/m^3)	k_w (N/m^3)
Set1	90.43	13.29	0.3611	15.06	264.7	2494	5885	2571
Set2	60.56	18.93	0.2942	25.36	156.1	6460	8362	2068

Table 4: New input data sets of the micro-PSC properties.

Model comparison	SDOE-AKriging ($\times10^4$ rad/s)	Physical free vibration analysis (PFVA, $\times10^4$ rad/s)	Relative error	SDOE-AKriging time (s)	PFVA time (s)
Set1	1.7777	1.7780	-0.01687%	0.000613	0.4317
Set2	1.4636	1.4659	-0.1569%	0.000276	0.3273

Table 5: Predicted size-dependent fundamental natural frequencies for new property input sets.

5 CONCLUSIONS

This research implements the stochastic size-dependent free vibration analysis of the micro-scale PSC under multi-dimensional uncertainties. With the aim of examining the scale effects on composite response, the refined shear deformation theory and modified strain gradient theory are utilized. An active learning strategy, specifically the sequential design of experiment enhanced adaptive Kriging, is incorporated to describe the underpinned and sophisticated relationship between various uncertainties and the focused size-dependent dynamic behaviour with tremendously reduced computational costs. Sufficient statistical information is provided, which is essential for the safety assessment and reliability-based design of micro-composites. Additionally, the superiority of the SDOE-AKriging over popular metamodels is confirmed through numerical experiments. Moreover, the developed tactic is capable of predicting the future size-dependent response of the PSC under rapidly varying environments, facilitating swift information updates in realistic engineering applications.

6 ACKNOWLEDGMENTS

This research is supported by the Hong Kong Innovation and Technology Fund (ITF) grant [PRP/011/23FX]. The authors would like to express their sincere appreciation for the funding support.

REFERENCES

[1] Green MA, Ho-Baillie A, Snaith HJ. The emergence of perovskite solar cells. *Nat Photonics* 2014;8:506–14. https://doi.org/10.1038/nphoton.2014.134.

[2] Bo L. *Nonlinear Dynamic Behaviour of Solar Cells With Advanced Materials*. Ph.D. University of New South Wales (Australia), 2023.

[3] Zhao S, Zhang Y, Kitipornchai S, Yang J. Opto-electro-thermo-mechanical behaviours of perovskite plates. *Int J Mech Sci* 2024;267:109016. https://doi.org/10.1016/j.ijmecsci.2024.109016.

[4] Bo L, Gao W, Yu Y, Chen X. Geometrically nonlinear dynamic analysis of the stiffened perovskite solar cell subjected to biaxial velocity impacts. *Nonlinear Dyn* 2022;110:281–311. https://doi.org/10.1007/s11071-022-07619-9.

[5] Bo L, Li Q, Tian Y, Wu D, Yu Y, Chen X, et al. Nonlinear dynamic investigation of the perovskite solar cell with GPLR-FGP stiffeners under blast impact. *Int J Mech Sci* 2022;213:106866. https://doi.org/10.1016/j.ijmecsci.2021.106866.

[6] Bo L, Li Q, Chen X, Gao W. Nonlinear dynamic instability of the perovskite solar cell under biaxial mechanical impacts. *Eng Fail Anal* 2022;139:106444. https://doi.org/10.1016/j.engfailanal.2022.106444.

[7] Bo L. Nonlinear dynamic analysis of the perovskite solar cell under blast impacts based on the modified strain gradient theory. *Acta Mech* 2023;234:1649–85. https://doi.org/10.1007/s00707-022-03444-8.

[8] Jones RM. *Mechanics Of Composite Materials*. 2nd ed. Boca Raton: CRC Press; 2018. https://doi.org/10.1201/9781498711067.

[9] Bo L, Zhang J, Wang H. Advanced virtual modelling aided stochastic nonlinear dynamic stability analysis of the GPLR-FGP plate in thermal environments. *Thin-Walled Struct* 2024;199:111833. https://doi.org/10.1016/j.tws.2024.111833.

[10] Zhang B, He Y, Liu D, Shen L, Lei J. An efficient size-dependent plate theory for bending, buckling and free vibration analyses of functionally graded microplates resting on elastic foundation. *Appl Math Model* 2015;39:3814–45. https://doi.org/10.1016/j.apm.2014.12.001.

[11] Bo L, Zhang J, Gao K, Wang H. Adaptive virtual modelling enhanced dynamic and reliability analysis of SGPLRP-MEE plates. *Int J Mech Sci* 2025;285:109827. https://doi.org/10.1016/j.ijmecsci.2024.109827.

[12] Bo L, Wang H. Nonlinear dynamic buckling and multi-objective design optimisation of FG-GPLRP plates. *Int J Mech Sci* 2023;256:108516. https://doi.org/10.1016/j.ijmecsci.2023.108516.

[13] Bo L, Wang H. Nonlinear dynamic instability and multi-objective design optimization of the GPLR-FGP plate under biaxial impacts. *Compos Struct* 2023;312:116854. https://doi.org/10.1016/j.compstruct.2023.116854.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

STOCHASTIC OPTIMIZATION FOR SPACE MISSION ANALYSIS

Thomas Caleb[1], Roberto Armellin[2], Spencer Boone[1], and Stéphanie Lizy-Destrez[1]

[1]DCAS/SaCLaB, ISAE-SUPAERO
Toulouse, 31055, France
e-mail: {thomas.caleb,spencer.boone,stephanie.lizy-destrez}@isae-supaero.fr

[2]Te Pūnaha Ātea – Space Institute, University of Auckland
Auckland, 1142, New Zealand
e-mail: roberto.armellin@auckland.ac.nz

Abstract. *Space missions are increasingly complex, with trajectories sensitive to perturbations that can lead to mission failure or costly maneuvers. Consequently, mission analysis robustness must be carefully evaluated, driving up development costs. This work aims to reduce failure risks while minimizing fuel consumption. Stochastic optimization methods help address these challenges and reduce mission design time. These methods rely on three components: 1. Uncertainty propagation to carry random distributions forward. 2. Transcription that reformulates stochastic constraints into deterministic ones. 3. A deterministic solver for the transcribed problem. Existing stochastic methods often struggle with onboard applications due to computational demands, limited precision, or conservatism. For instance, Stochastic Differential Dynamic Programming (SDDP) [1] uses sequential quadratic optimization and propagates Gaussian uncertainties through linearized dynamics, but its transcription method, based on the spectral radius of constraint covariances, can be overly conservative.*

To address these limitations, we introduce a Taylor polynomial-based SDDP solver, using Taylor polynomials [2] for automatic differentiation, dynamic approximations, and uncertainty propagation. Combining these functions into one framework, we significantly accelerate the code, making it viable for onboard computations. Additionally, we propose a novel transcription method for constraints and cost functions, reducing conservatism. Our tool was tested on an Earth-Mars transfer scenario, yielding a robust control law that meets performance expectations. With a failure risk limit of 5%*, the code runs in under a minute on a standard laptop, with a failure risk of* 2.9%*, compared to* 99.6% *for the non-robust version, with similar fuel consumption. This control law can be implemented onboard, balancing cost control and mission reliability. Thus, this work marks a significant step toward practical onboard applications of stochastic optimization, providing a fast, less conservative approach for reliable control of complex missions.*

Keywords: Space mission analysis, Stochastic optimization, Chance constraints, Differential dynamic programming, High-order Taylor polynomials.

1 INTRODUCTION

Modern space missions increasingly rely on sophisticated dynamical systems, requiring precise trajectory design to ensure success. Interplanetary missions often rely on gravity assists, as in European space agency (ESA) JUICE mission [see 9], or exploit sensitive phenomena like ballistic capture [32], a key factor in the ESA and Japan aerospace exploration agency (JAXA)'s BepiColombo mission [3]. Additionally, advanced dynamical models, such as the circular restricted three-body problem (CR3BP), play a key role in enabling sustainable lunar exploration [30].

However, the non-linear and chaotic nature of these dynamical systems poses significant challenges for trajectory design and mission robustness [27]. Small variations in a spacecraft's position due to navigation uncertainties, measurement errors, or maneuver inaccuracies can lead to substantial deviations from the intended path. Ensuring mission success, therefore, requires trajectory designs that are both efficient and resilient to these uncertainties.

Traditionally, trajectory planning has relied on deterministic optimization, where an optimal path is computed without accounting for stochastic effects. Once determined, the robustness of this trajectory is tested against potential uncertainties, often requiring additional fuel for corrective maneuvers. While effective in many cases, this method can yield suboptimal or even infeasible solutions when dealing with highly sensitive or constrained missions.

To address these limitations, stochastic optimization techniques have gained interest in space mission analysis [6, 24, 2]. These approaches incorporate uncertainty directly into the trajectory design process, allowing for more robust solutions that mitigate risks while maintaining mission feasibility. Recent methodologies have been generally structured into three main blocks:

(i) Uncertainty propagation: models how uncertainties evolve over time, affecting the spacecraft's trajectory.

(ii) Transcription method: transforms stochastic constraints and objective function into a deterministic, computationally tractable form. However, existing approaches often introduce conservatism or are limited to specific cases [6, 29].

(iii) Deterministic optimization solver: used to solve the transcribed problem. Off-the-shelf solvers are commonly used [14, 2, 21]. Recent research efforts have focused on specialized solvers that perform common computations for all three blocks to improve efficiency and feasibility [24, 23].

To address the limitations of existing methods, we introduce linear stochastic optimization with differential algebra (L-SODA) solver, a constrained stochastic differential dynamic programming (DDP) solver that pools computations using the differential algebra (DA) framework framework to enhance efficiency and accuracy. Moreover, it implements novel transcription methods that strongly reduce conservatism and improve convergence.

Section 2 introduces constrained DDP, stochastic optimization, transcription methods, and the role of DA in scientific computing. Then, we present novel transcription techniques and introduce the L-SODA solver in Section 3. Section 4 presents validation and application of this solver, respectively to a double integrator theoretical problem, and to a standard fuel-optimal low-thrust Earth-Mars transfer. Finally, conclusions are drawn in Section 5.

2 BACKGROUND

This section briefly presents the theory on constrained DDP, stochastic solvers in astrodynamics and and the DA.

2.1 Constrained Differential Dynamic Programming

This part presents a constrained DDP solver, a deterministic solver for constrained optimization. We first introduce DDP and then present the augmented Lagrangian (AUL) formulation.

2.1.1 Differential Dynamic Programming

DDP tackles unconstrained optimization problems divided into N stages with dynamics of the type: $\boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k) = \boldsymbol{x}_{k+1}$ [22, 16], where $k \in \mathbb{N}_{N-1} = \{0, 1, \ldots, N-1\}$, $\boldsymbol{x}_k$ is the state vector of size N_x, $\boldsymbol{u}_k$ is the control vector of size N_u, and $\boldsymbol{f}$ represents the dynamics. The initial and target state vectors $\boldsymbol{x}_0$ and $\boldsymbol{x}_t$ of size N_x are given, and the objective function to minimize is:

$$J(\boldsymbol{U}, \boldsymbol{x}_0, \boldsymbol{x}_t) = \sum_{k=0}^{N-1} l(\boldsymbol{x}_k, \boldsymbol{u}_k) + \phi(\boldsymbol{x}_N, \boldsymbol{x}_t) \tag{1}$$

where $\boldsymbol{U} = (\boldsymbol{u}_0, \boldsymbol{u}_1, \ldots, \boldsymbol{u}_{N-1})$ is the vector of controls, $\boldsymbol{x}_N$ is the final propagated state vector, l is the stage cost, and ϕ is the terminal cost. DDP consists of two phases looping until convergence:

(i) **Backward sweep**: Given a control $\boldsymbol{U}$, a list of states $\boldsymbol{X} = (\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)$, and costs $l_k = l(\boldsymbol{x}_k, \boldsymbol{u}_k)$ and $\phi_N = \phi(\boldsymbol{x}_N, \boldsymbol{x}_t)$, determine the list of corrections $\boldsymbol{a}_k$ and $\boldsymbol{K}_k$ to perform on $\boldsymbol{u}_k$ from $k = N-1$ to $k = 0$ to reduce the overall cost. These corrections are computed using Bellman's principle and by successive quadratic approximation of the cost-to-go function: $V_k(\boldsymbol{x}_k) = \min_{\boldsymbol{U}_k} \left[\sum_{i=k}^{N-1} l(\boldsymbol{x}_k, \boldsymbol{u}_k) + \phi(\boldsymbol{x}_N, \boldsymbol{x}_t) \right]$. This phase requires the first and second-order derivatives of $\boldsymbol{f}_k = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k)$, l_k, and ϕ_N with respect to $\boldsymbol{x}$ and $\boldsymbol{u}$.

(ii) **Forward pass**: Given the corrections $\boldsymbol{A} = (\boldsymbol{a}_0, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_{N-1})$ and $\boldsymbol{K} = (\boldsymbol{K}_0, \boldsymbol{K}_1, \ldots, \boldsymbol{K}_{N-1})$, and starting from $k = 0$ to $k = N-1$, compute the new controls $\boldsymbol{u}_k^*$ depending on the new state $\boldsymbol{x}_k^*$: $\boldsymbol{u}_k^* = \boldsymbol{u}_k + \boldsymbol{a}_k + \boldsymbol{K}_k \cdot (\boldsymbol{x}_k^* - \boldsymbol{x}_k)$, with $\boldsymbol{x}_0^* = \boldsymbol{x}_0$. This phase requires the evaluation of the dynamics, which is the most computation-intensive part [7].

2.1.2 Augmented Lagrangian Formulation

The DDP solver does not implement constraints by design, so they must be added to solve typical space trajectory design problems. We present the method of the ALTRO solver from Howell et al. [15], which uses the AUL formulation. In this approach, constraints are added to the objective function using a dual state and a penalty factor. The constrained problem can now be solved

using an unconstrained solver, such as DDP. The constrained problem is formulated as follows:

$$\begin{aligned}
&\min_{\boldsymbol{U}} J\left(\boldsymbol{U}, \boldsymbol{x}_0, \boldsymbol{x}_t\right) = \sum_{k=0}^{N-1} l\left(\boldsymbol{x}_k, \boldsymbol{u}_k\right) + \phi\left(\boldsymbol{x}_N, \boldsymbol{x}_t\right), \\
&\text{subject to:} \\
&\boldsymbol{x}_{k+1} = \boldsymbol{f}\left(\boldsymbol{x}_k, \boldsymbol{u}_k\right), \ \forall k \in \mathbb{N}_{N-1}, \\
&\boldsymbol{g}\left(\boldsymbol{x}_k, \boldsymbol{u}_k\right) \preceq \mathbf{0}_{N_g}, \ \forall k \in \mathbb{N}_{N-1}, \\
&\boldsymbol{g}_t\left(\boldsymbol{x}_N, \boldsymbol{x}_t\right) \preceq \mathbf{0}_{N_{gt}},
\end{aligned} \tag{2}$$

where $\boldsymbol{g}_k = \boldsymbol{g}\left(\boldsymbol{x}_k, \boldsymbol{u}_k\right)$ are the path constraints of size N_g, $\boldsymbol{g}_N = \boldsymbol{g}_t\left(\boldsymbol{x}_N, \boldsymbol{x}_t\right)$ are the terminal constraints of size N_{gt}, and $\forall \left(\boldsymbol{y}, \boldsymbol{z}\right) \in \left(\mathbb{R}^d\right)^2$: $\boldsymbol{y} \preceq \boldsymbol{z} \Leftrightarrow \forall i \in \mathbb{N}_d, \ y^i \leq z^i$.

This problem is solved by augmenting the stage costs with the path constraints to obtain $\tilde{l}$ and the terminal cost with the terminal constraints to obtain $\tilde{\phi}$:

$$\begin{cases}
\tilde{l}\left(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\lambda}_k, \boldsymbol{\mu}_k\right) & = l_k + \left[\boldsymbol{\lambda}_k + \frac{\mathcal{I}_{\boldsymbol{\mu},k}}{2} \cdot \boldsymbol{g}_k\right]^T \cdot \boldsymbol{g}_k, \ k \in \mathbb{N}_{N-1} \\
\tilde{\phi}\left(\boldsymbol{x}_N, \boldsymbol{x}_t, \boldsymbol{\lambda}_N, \boldsymbol{\mu}_N\right) & = \phi_N + \left[\boldsymbol{\lambda}_N + \frac{\mathcal{I}_{\boldsymbol{\mu},N}}{2} \cdot \boldsymbol{g}_N\right]^T \cdot \boldsymbol{g}_N
\end{cases} \tag{3}$$

where the tilde denotes augmented cost functions, $\boldsymbol{\lambda}_k$ is the dual state of size N_g if $k \in \mathbb{N}_{N-1}$ or of size N_{gt} if $k = N$, and $\boldsymbol{\mu}_k$ is the penalty vector of the same size as $\boldsymbol{\lambda}_k$. Please refer to Howell et al. [15] for more details on dual state and penalty updates. The augmented problem is solved using DDP until the maximum constraint violation $g_{\max}$ is below the target constraint satisfaction $\varepsilon_{\text{AUL}} > 0$.

2.2 Stochastic Optimization Problem and Existing Chance-Constraints Transcription Methods

Several assumptions are made: uncertainties follow a multivariate Gaussian distribution [31], functions are sufficiently regular, and covariance norms are small. These ensure all random variables can be modeled as Gaussian distributions.

2.2.1 Constrained Stochastic Trajectory Optimization Problem

We now formulate the constrained stochastic trajectory optimization problem within the DDP framework. The states and controls are modeled as Gaussian distributions with respective means and covariances $\bar{\boldsymbol{x}}_k$, $\bar{\boldsymbol{u}}_k$, $\boldsymbol{\Sigma}_{\boldsymbol{x},k}$, and $\boldsymbol{\Sigma}_{\boldsymbol{u},k}$. The nominal controls are augmented with a linear feedback gain $\boldsymbol{K} = \left(\boldsymbol{K}_0, \boldsymbol{K}_1, \ldots, \boldsymbol{K}_{N-1}\right)$. The goal is to find a control $\left(\bar{\boldsymbol{U}}, \boldsymbol{K}\right)$ of failure risk below a given $\beta \in]0, 1[$, the target failure risk. These controls must minimize $\tilde{J}$, the $(1-\beta)$–quantile of $J\left(\bar{\boldsymbol{U}}, \boldsymbol{K}, \boldsymbol{x}_0, \boldsymbol{x}_t\right)$ (or $(1-\beta)$–cost). In other words, there is a probability $1-\beta$ that the cost function will be below the optimized objective function $\tilde{J}$. The problem is

formulated as follows:

$$
\begin{aligned}
&\min_{\bar{\boldsymbol{U}},\boldsymbol{K}} \tilde{J}\left(\bar{\boldsymbol{U}},\boldsymbol{K},\boldsymbol{x}_0,\boldsymbol{x}_t,\beta\right),\\
&\text{subject to:}\\
&J\left(\bar{\boldsymbol{U}},\boldsymbol{K},\boldsymbol{x}_0,\boldsymbol{x}_t\right)=\sum_{k=0}^{N-1} l\left(\boldsymbol{x}_k,\boldsymbol{u}_k\right)+\phi\left(\boldsymbol{x}_N,\boldsymbol{x}_t\right),\\
&\boldsymbol{x}_{k+1}=\boldsymbol{f}\left(\boldsymbol{x}_k,\boldsymbol{u}_k\right)+\boldsymbol{w}_{x,k+1},\ \forall k\in\mathbb{N}_{N-1},\\
&\boldsymbol{u}_k=\bar{\boldsymbol{u}}_k+\boldsymbol{K}_k\cdot\left(\boldsymbol{x}_k-\bar{\boldsymbol{x}}_k\right),\ \forall k\in\mathbb{N}_{N-1},\\
&\mathbb{P}\left[J\left(\bar{\boldsymbol{U}},\boldsymbol{K},\boldsymbol{x}_0,\boldsymbol{x}_t\right)\leq\tilde{J}\left(\bar{\boldsymbol{U}},\boldsymbol{K},\boldsymbol{x}_0,\boldsymbol{x}_t,\beta\right)\right]\geq 1-\beta,\\
&\mathbb{P}\left[\boldsymbol{g}\left(\boldsymbol{x}_k,\boldsymbol{u}_k\right)\preceq\boldsymbol{0}_{N_g}\right]\geq 1-\beta,\ \forall k\in\mathbb{N}_{N-1},\\
&\mathbb{P}\left[\boldsymbol{g}_t\left(\boldsymbol{x}_N,\boldsymbol{x}_t\right)\preceq\boldsymbol{0}_{N_{gt}}\right]\geq 1-\beta,
\end{aligned}
\tag{4}
$$

where $\boldsymbol{x}_0\sim\mathcal{N}\left(\bar{\boldsymbol{x}}_0,\boldsymbol{\Sigma}_{\boldsymbol{x},0}\right)$ is the initial distribution, $\boldsymbol{x}_t\sim\mathcal{N}\left(\bar{\boldsymbol{x}}_t,\boldsymbol{\Sigma}_{\boldsymbol{x},t}\right)$ is the target distribution, and $\boldsymbol{w}_{\boldsymbol{x},k}\sim\mathcal{N}\left(\boldsymbol{0}_{Nx},\boldsymbol{\mathcal{Q}}_{\boldsymbol{x},k}\right)$ is the navigation error. Robust solvers' transcription methods are designed to enforce hard constraints with a failure risk lower than a given β. First of all, recall that $\boldsymbol{g}\left(\boldsymbol{x}_k,\boldsymbol{u}_k\right)$ and $\boldsymbol{g}_t\left(\boldsymbol{x}_N,\boldsymbol{x}_t\right)$ are Gaussian variables of respective size N_g and N_{gt}. Therefore, finding a transcription method for constraints of the type $\mathbb{P}\left(\boldsymbol{y}\preceq\boldsymbol{0}_d\right)\geq 1-\beta$ for any $\boldsymbol{y}\sim\mathcal{N}\left(\bar{\boldsymbol{y}},\boldsymbol{\Sigma}\right)$ of size $d\in\mathbb{N}^*$ will allow us to transcribe the constraints of Eq. (4). These methods essentially consist of finding a tractable sufficient condition $\mathcal{A}\left(\bar{\boldsymbol{y}},\boldsymbol{\Sigma},\beta\right)$: $\mathcal{A}\left(\bar{\boldsymbol{y}},\boldsymbol{\Sigma},\beta\right)\implies\mathbb{P}\left(\boldsymbol{y}\preceq\boldsymbol{0}_d\right)\geq 1-\beta$.

2.2.2 Transcription methods

Certain methods were designed to tackle specific types of chance-constraints. For instance, Ridderhof et al. [29] transcribed control norm constraints used in Ridderhof et al. [29], Benedikter et al. [2], Marmo and Zavoli [20]. As shown in Caleb et al. [12], this method rests on an spectral radius domination and are not easily transposed to other constraint types. Generic transcriptions exist for any univariate Gaussian distribution y. Blackmore et al. [6] start from a multivariate Gaussian random variable $\boldsymbol{\lambda}\sim\mathcal{N}\left(\bar{\boldsymbol{\lambda}},\boldsymbol{\Sigma}_\lambda\right)$ of size v and tackle the following chance-constraint:

$$
\mathbb{P}\left(\boldsymbol{h}^T\cdot\boldsymbol{\lambda}\leq a\right)\geq 1-\beta \tag{5}
$$

where $\boldsymbol{h}\in\mathbb{R}^v$ and $a\in\mathbb{R}$. They provide the following necessary and sufficient condition:

$$
\boldsymbol{h}^T\cdot\bar{\boldsymbol{\lambda}}+\Phi_G^{-1}\left(1-\beta\right)\sqrt{\boldsymbol{h}^T\boldsymbol{\Sigma}_\lambda\boldsymbol{h}}\leq a\ \Leftrightarrow\ \mathbb{P}\left(\boldsymbol{h}^T\cdot\boldsymbol{\lambda}\leq a\right)\geq 1-\beta \tag{6}
$$

where Φ_G^{-1} is the inverse cumulative distribution function (CDF) of a standard Gaussian distribution.

Setting $y=\boldsymbol{h}^T\cdot\boldsymbol{\lambda}-a$ allows us to generalize:

$$
\bar{y}+\Phi_G^{-1}\left(1-\beta\right)\sqrt{\Sigma}\leq 0\ \Leftrightarrow\ \mathbb{P}\left(y\leq 0\right)\geq 1-\beta \tag{7}
$$

This method is widely used in the literature [see 28, 18, 25], though it cannot tackle multidimensional constraints.

2.3 High-Order Taylor Polynomial Algebra

DA is a framework for manipulating high-order Taylor expansions [5]. It represents a differentiable function $\boldsymbol{h}$ of v variables with its Taylor expansion $\mathcal{P}_h$, providing a description of $\boldsymbol{h}$ over its domain. Algebraic and functional operations, including polynomial inversion, are well-defined on Taylor polynomials [4]. It has many practical applications such as:

- **Function Approximation**: Uses Taylor-Lagrange's theorem for evaluating functions at points near a given value. It provides a polynomial approximation with bounded error. For instance, evaluating $\sin$ at a point y near x can be done by evaluating the polynomial $\sum_{k=0}^{p} \frac{\sin^{(k)}(x)}{k!}(y-x)^k$ at $\delta x = y - x$, with an absolute error bounded by $\frac{\sin^{(p+1)}(\xi)}{(p+1)!}\delta x^{p+1}$. The run time of this method is faster than standard evaluation for large sampling [1]. This method is extensively used for sampling, Monte Carlo validation, or faster optimization [10, 19].
- **Automatic Differentiation**: Differentiates functions without using finite differences, which are computationally intensive and imprecise, and without computing and implementing derivatives by hand. The derivatives are accurate up to machine precision. The derivatives are simply retrieved from the coefficient of the Taylor polynomials. This property is useful for parameter-dependent functions and quick testing, allowing easy changes in dynamics, constraints, etc. The differential algebra-based differential dynamic programming (DADDy)[1] solver implements this feature, along with function approximation implemented in Caleb et al. [11].
- **Uncertainty Propagation**: Uses Taylor polynomials to approximate how uncertain states propagate over time. Given an initial uncertain state $\mathcal{P}_{\boldsymbol{x},0} = \boldsymbol{x}_0 + \delta\boldsymbol{x}_0$ and a dynamical system $\boldsymbol{f}$ such that $\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k)$, the mapping $\mathcal{P}_{\boldsymbol{x},t}$ of the state $\boldsymbol{x}$ at any time t can be computed as a function of $\delta\boldsymbol{x}_0$. This method simulates the final state and distribution with arbitrary precision using either high-order polynomials or automatic domain splitting (ADS) [33]. Work has been done to specifically represent Gaussian distributions using DA [26, 8, 19].

3 METHODOLOGY

3.1 Chance-Constraints Transcription

This part present two general d–dimensional transcription methods. Let $\boldsymbol{y} \sim \mathcal{N}(\bar{\boldsymbol{y}}, \boldsymbol{\Sigma})$, $\beta \in]0, 1[$ the target failure risk, and $\rho(\boldsymbol{\Sigma})$ the positive square root of the spectral radius of $\boldsymbol{\Sigma}$, i. e., its largest eigenvalue. A first transcription results inspired by the spectral radius domination of Ridderhof et al. [29] is given:

Theorem 1 (SPECTRAL RADIUS TRANSCRIPTION.)**.**

$$\bar{\boldsymbol{y}} + \Psi_d^{-1}(\beta)\,\rho(\boldsymbol{\Sigma}) \preceq \mathbf{0}_d \implies \mathbb{P}(\boldsymbol{y} \preceq \mathbf{0}_d) \geq 1 - \beta \tag{8}$$

where $\Psi_d^{-1}(\beta) = \sqrt{\Phi_d^{-1}(1-\beta)}$*, and* Φ_d^{-1} *is the inverse CDF of a standard* χ^2 *distribution with* d *degrees of freedom.*

Proof. See Caleb et al. [12]. □

[1]Library available at: `https://github.com/ThomasClb/DADDy.git` [last accessed Feb 20, 2025].

We also provide a second less conservative result:
Theorem 2 (FIRST ORDER TRANSCRIPTION.)**.** *Let $\boldsymbol{\sigma}$ be the vector of the positive square roots of the diagonal coefficients of $\boldsymbol{\Sigma}$, then:*

$$\bar{\boldsymbol{y}} + \Psi_d^{-1}(\beta)\,\boldsymbol{\sigma} \preceq \mathbf{0}_d \implies \mathbb{P}(\boldsymbol{y} \preceq \mathbf{0}_d) \geq 1-\beta \tag{9}$$

Proof. See Caleb et al. [12]. □

These two new results can be used for constraints and cost function transcription provided that the functions are sufficiently regular and that the covariances are small enough so that the functions are locally linear.

Indeed, if the chance constraint is formulated as $\mathbb{P}(\boldsymbol{h}(\boldsymbol{x},\boldsymbol{u}) \preceq \mathbf{0}) \geq 1-\beta$, where $\boldsymbol{h}$ is a linear function of the state and the control that outputs in $\mathbb{R}^d$. Moreover, we define $\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x},\boldsymbol{u})$. If $\delta\boldsymbol{x} = \boldsymbol{x} - \bar{\boldsymbol{x}}$, then:

$$\boldsymbol{y} = \boldsymbol{h}(\bar{\boldsymbol{x}},\bar{\boldsymbol{u}}) + [\boldsymbol{h}_{\boldsymbol{x}} + \boldsymbol{h}_{\boldsymbol{u}} \cdot \boldsymbol{K}] \cdot \delta\boldsymbol{x} \tag{10}$$

where $\boldsymbol{h}_{\boldsymbol{x}}$ and $\boldsymbol{h}_{\boldsymbol{u}}$ are respectively the gradient of $\boldsymbol{h}$ with respect to $\boldsymbol{x}$ and $\boldsymbol{u}$. Note that $\delta\boldsymbol{x} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{x}})$, therefore, since $\boldsymbol{y}$ is a linear combination of Gaussian variables, $\boldsymbol{y} \sim \mathcal{N}(\bar{\boldsymbol{y}}, \boldsymbol{\Sigma})$ with:

$$\bar{\boldsymbol{y}} = \boldsymbol{h}(\bar{\boldsymbol{x}},\bar{\boldsymbol{u}}),\ \boldsymbol{\Sigma} = (\boldsymbol{h}_{\boldsymbol{x}} + \boldsymbol{h}_{\boldsymbol{u}} \cdot \boldsymbol{K}) \cdot \boldsymbol{\Sigma}_x \cdot (\boldsymbol{h}_{\boldsymbol{x}} + \boldsymbol{h}_{\boldsymbol{u}} \cdot \boldsymbol{K})^T \tag{11}$$

Thus, if a given function $\boldsymbol{h}$ is linear, chance constraints of type: $\mathbb{P}(\boldsymbol{h}(\boldsymbol{x},\boldsymbol{u}) \preceq \mathbf{0}) \geq 1-\beta$ can now be transcribed.

3.2 Linear Stochastic Optimization solver

In this part we introduce the L-SODA solver that implement our novel transcription methods, and our DADDy deterministic solver [11].

The first uncertainty source is the error on the initial state $\boldsymbol{x}_0 \sim \mathcal{N}(\bar{\boldsymbol{x}}_0, \boldsymbol{\Sigma}_{\boldsymbol{x},0})$. Then, we consider navigation errors at each time step: $\widehat{\boldsymbol{x}}_{k+1} = \widehat{\boldsymbol{f}}_k + \boldsymbol{w}_{x,k+1}$, where $\boldsymbol{w}_{\boldsymbol{x},k} \sim \mathcal{N}(\mathbf{0}_{Nx}, \boldsymbol{\mathcal{Q}}_{\boldsymbol{x},k})$. Therefore, the state $\boldsymbol{x}_k \sim \mathcal{N}(\bar{\boldsymbol{x}}_k, \boldsymbol{\Sigma}_{\boldsymbol{x},k})$ can be updated as follows [2]:

$$\begin{cases} \boldsymbol{\Sigma}_{\boldsymbol{u},k} & = \boldsymbol{K}_k \boldsymbol{\Sigma}_{\boldsymbol{x},k} \boldsymbol{K}_k^T \\ \bar{\boldsymbol{x}}_{k+1} & = \bar{\boldsymbol{f}}_k \\ \boldsymbol{\Sigma}_{\boldsymbol{x},k+1} & = \left(\bar{\boldsymbol{f}}_{k,\boldsymbol{x}} + \boldsymbol{K}_k \bar{\boldsymbol{f}}_{k,\boldsymbol{u}}\right) \cdot \boldsymbol{\Sigma}_{\boldsymbol{x},k} \cdot \left(\bar{\boldsymbol{f}}_{k,\boldsymbol{x}} + \boldsymbol{K}_k \bar{\boldsymbol{f}}_{k,\boldsymbol{u}}\right)^T + \boldsymbol{\mathcal{Q}}_{\boldsymbol{x},k+1} \end{cases} \tag{12}$$

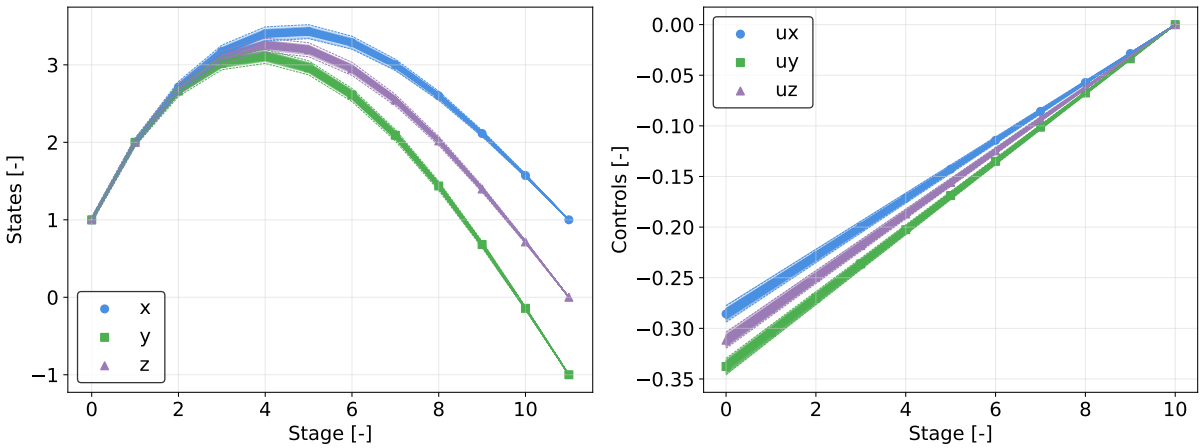
These computations can be done effortlessly using the gradients already computed during the DDP round.

From this, the cost functions l, ϕ, and the constraints $\boldsymbol{g}$ and $\boldsymbol{g}_t$ can be linearized and computed. They are then modified using the transcription method. For instance, l and $\boldsymbol{g}$ are concatenated together: $\boldsymbol{y} = \left[l(\boldsymbol{x},\boldsymbol{u}), \boldsymbol{g}(\boldsymbol{x},\boldsymbol{u})^T\right]^T \sim \mathcal{N}(\bar{\boldsymbol{y}}, \boldsymbol{\Sigma})$. Let us define:

$$\tilde{\boldsymbol{y}} = \bar{\boldsymbol{y}} + \Psi_d^{-1}(\beta)\,\boldsymbol{\sigma} \tag{13}$$

where $d = N_g + 1$. We also set $\tilde{l}$ as the first component of $\tilde{\boldsymbol{y}}$ and $\tilde{\boldsymbol{g}}$ as the N_g remaining components. Then:

$$\mathbb{P}\left[\left\{l(\boldsymbol{x},\boldsymbol{u}) \leq \tilde{l}(\boldsymbol{x},\boldsymbol{u})\right\} \cap \left\{\boldsymbol{g}(\boldsymbol{x},\boldsymbol{u}) \leq \tilde{\boldsymbol{g}}(\boldsymbol{x},\boldsymbol{u})\right\}\right] \geq 1-\beta \tag{14}$$



(a) State vectors.

(b) Controls.

Figure 1: Solution to the double integrator problem.

Thus, replacing l and $\boldsymbol{g}$ by $\tilde{l}$ and $\tilde{\boldsymbol{g}}$, and similarly for ϕ and $\boldsymbol{g}_t$, allows us to solve the stochastic problem of Eq. (4).

Finally, the linearity hypothesis is checked via the non-linearity index (NLI) from Losacco et al. [19]. This index corresponds to the norm of the Hessian divided by the norm of the gradient. A high NLI means the Hessian has similar magnitude as the gradient, i. e., the dynamics is not linear. Having a small NLI shows the second-order terms of the Taylor expansion can be neglected. In other words, the linear hypothesis is valid. This criteria is computed efficiently thanks to previous polynomial computations of the DA framework.

4 RESULTS

4.1 Validation: Double Integrator Problem

The solver developed for this work was validated against the unconstrained double integrator optimization problem from [17]. The optimization problem is defined by:

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) &= \left[\boldsymbol{v}^T, \boldsymbol{u}^T\right]^T \\ l(\boldsymbol{x}, \boldsymbol{u}) &= \boldsymbol{u}^T \cdot \boldsymbol{u} \\ \phi(\boldsymbol{x}, \boldsymbol{x}_t) &= (\boldsymbol{r} - \boldsymbol{r}_t)^T \cdot (\boldsymbol{r} - \boldsymbol{r}_t) \end{aligned} \tag{15}$$

where $N_x = 6$, $N_u = 3$, $\boldsymbol{x} = [x, y, z, v^x, v^y, v^z]^T = \left[\boldsymbol{r}^T, \boldsymbol{v}^T\right]^T$, with $\boldsymbol{r}$ the position of size $\frac{N_x}{2}$, and $\boldsymbol{v} = \dot{\boldsymbol{r}}$ the velocity. The number of stages is $N = 11$, the initial conditions and target of the problem are $\boldsymbol{x}_0 = [1, 1, 1, 1, 1, 1]^T$ and $\boldsymbol{x}_t = [1, -1, 0, 0, 0, 0]^T$, the initial covariance is $\boldsymbol{\Sigma}_{\boldsymbol{x},0} = \text{diag}\left(10^{-2}, 10^{-2}, 10^{-2}, 10^{-2}, 10^{-2}, 10^{-2}\right)^2$, and the navigation errors have a covariance $\boldsymbol{\mathcal{Q}}_{\boldsymbol{x},k} = \frac{1}{100}\boldsymbol{\Sigma}_{\boldsymbol{x},0}$. This problem was solved with as first guess $\boldsymbol{U}_0$: a $N \cdot N_u$ vector with all components equal to 10^{-5}. The result, shown in Fig. 1, is found in a single iteration. Fig. 1a shows the states and Fig. 1b the controls. The dots correspond to the nominal system state or control and the lines correspond to sample trajectories. The results match the ones of [17] and Table 1 provides additional information on the total cost and run time of the method. The two solvers converged towards a similar robust solution, since the problem is convex. However, the run time is higher for the robust method due to significant overhead.

Table 1: Double integrator convergence data.

Solver	**Run time** [s]	**Nominal cost**	$(1-\beta)$**–quartile cost**
Deterministic	5.32×10^{-3}	1.1273	1.3616
Stochastic $\beta = 5\%$	9.73×10^{-3}	1.1273	1.3616

Table 2: Earth-Mars two-body problem transfer initial and target states.

Type	x [km]	y [km]	z [km]	$\dot{x}$ [km/s]	$\dot{y}$ [km/s]	$\dot{z}$ [km/s]	m [kg]
Departure	−140 699 693	−51 614 428	980	9.774 596	−28.078 28	$4.337\,725 \times 10^{-4}$	1000
Target	−172 682 023	176 959 469	7 948 912	−16.427 384	−14.860 506	$9.214\,86 \times 10^{-2}$	–

4.2 Stochastic Earth-Mars Fuel-Optimal Transfer

The solver was then tested against a fuel-optimal low-thrust Earth-Mars transfer optimization problem adapted from [17]:

$$
\begin{aligned}
\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) &= \boldsymbol{x} + \int_0^{\Delta t} \left[\boldsymbol{v}^T, \dot{\boldsymbol{v}}^T, \dot{m}\right]^T dt \\
\dot{\boldsymbol{v}} &= -\frac{\mu}{\|\boldsymbol{r}\|_2^3}\boldsymbol{r} + \frac{\boldsymbol{u}}{m} \\
\dot{m} &= -\frac{\|\boldsymbol{u}\|_2}{g_0 \cdot \mathrm{Isp}} \\
l(\boldsymbol{x}, \boldsymbol{u}) &= \|\boldsymbol{u}\|_2 \\
\phi(\boldsymbol{x}, \boldsymbol{x}_t) &= (\boldsymbol{r} - \boldsymbol{r}_t)^T \cdot (\boldsymbol{r} - \boldsymbol{r}_t) + (\boldsymbol{v} - \boldsymbol{v}_t)^T \cdot (\boldsymbol{v} - \boldsymbol{v}_t) \\
\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}) &= \left[\boldsymbol{u}^T \cdot \boldsymbol{u} - u_{\max}^2, m_{\text{dry}} - m\right]^T
\end{aligned}
\tag{16}
$$

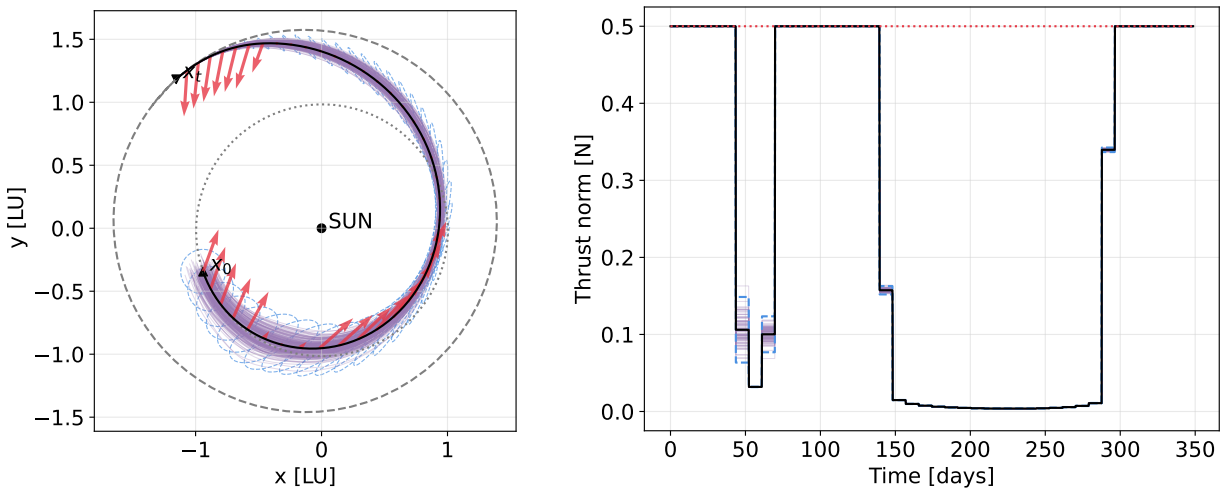
where $N_x = 7$, $N_u = 3$, the state vector $\boldsymbol{x}$ can be written: $[x, y, z, \dot{x}, \dot{y}, \dot{z}, m]$ or $\left[\boldsymbol{r}^T, \boldsymbol{v}^T, m\right]^T$, and m is the spacecraft mass. The number of stages is $N = 40$, the time-of-flight (ToF) is $348.79\,\mathrm{d}$, $\Delta t = \frac{\text{ToF}}{N}$, the initial conditions and target are given in Table 2. The initial covariance is $\boldsymbol{\Sigma}_{\boldsymbol{x},0} = \mathrm{diag}\left(10^{-6}, 10^{-6}, 10^{-6}, 5 \times 10^{-6}, 5 \times 10^{-6}, 5 \times 10^{-6}\right)^2$, and the navigation errors have a covariance $\boldsymbol{\mathcal{Q}}_{\boldsymbol{x},k} = \frac{1}{100}\boldsymbol{\Sigma}_{\boldsymbol{x},0}$. These errors were selected such that the highest value of the NLI is small (around 10^{-4}).

Normalization units and various dynamics parameters are reported in Table 3. All mass parameters in this work were obtained from JPL DE431 ephemerides [2] [13]. The first guess $\boldsymbol{U}_0$ is a $N \cdot N_u$ vector with all components equal to $10^{-6}\,\mathrm{N}$. Fig. 2 shows the solution to this optimization problem. Fig. 2a shows the trajectory from Earth to Mars with errors magnified $\times 50\,000$. The nominal trajectory is in black, the red arrows represent the thrust vectors, the theoretical $(1-\beta)$–ellipsoids are shown in blue and the purple lines each represent a different sample. Fig. 2b represents the control norm with errors magnified $\times 200$. The black line is the nominal control, the dotted red line is the control maximum magnitude $u_{\max}$, the dashed blue line represents the $(1-\beta)$–ellipsoids (two lines in 1D), and the purple lines correspond to the controls of the samples. These plots show the random samples follow a control law similar to the nominal one, yet it is clear the stochastic maneuvers are performed at the transition between

[2] Publicly available at: `https://naif.jpl.nasa.gov/pub/naif/generic_kernels/pck/gm_de431.tpc` [retrieved on Feb 20, 2025].

Table 3: Sun-centered normalization units and dynamics parameters.

Type	Unit	Symbol	Value	Comment
Normalization units	Mass parameter [-]	μ	$1.327\,124\,400\,41 \times 10^{11}$	Sun's mass parameter
	Length [km]	LU	149 597 870.7	Astronomical unit of length
	Time [s]	TU	5 022 642.891	$(\mathrm{LU}^3/\mu)^{1/2}$
	Velocity [km/s]	VU	29.784 691 83	LU/TU
Parameters	Standard gravity [m/s^2]	g_0	9.81	
	Specific impulse [s]	Isp	2000	
	Spacecraft dry mass [kg]	m_{dry}	500	
	Maximum spacecraft thrust [N]	$u_{\max}$	0.5	



(a) Trajectory in the x–y plane.

(b) Thrust profile.

Figure 2: Solution to the stochastic fuel-optimal Earth-Mars transfer.

Table 4: Stochastic fuel-optimal Earth-Mars transfer convergence data for $\beta = 5\%$.

Solver	**Run time** [s]	**Nominal error [-]**	$(1-\beta)$**–cost** [kg]	**Failure rate [%]**	**Conservatism**
Deterministic	45.9	4×10^{-13}	396.9	99.6	–
Stochastic SR	69.2	-6×10^{-6}	397.6	$<3 \times 10^{-3}$	>1700
Stochastic FO	45.9	-1×10^{-10}	396.9	2.93	1.7

the nominal thrust and cruise arcs. The covariance is at its largest at the beginning of the transfer and shrinks when performing these correction maneuvers.

Quantitative results comparing the deterministic solution of this problem and the stochastic solving using the spectral radius transcription (SR) and the first order transcription (FO) are shown in Table 4. The $(1-\beta)$–cost, the real failure rate, and the conservatism were computed with Monte-Carlo (MC) sampling of size 10^5. It is clear that deterministic solving leads to unsafe solutions when adding uncertainties. Comparing the two transcription methods, it appears that the spectral radius method is highly conservative. Indeed, the nominal is far below 0 while the first order method is four orders of magnitude closer. This leads to an extra fuel consumption of $0.7\,\mathrm{kg}$ of fuel, which is non-negligible considering the small magnitude of the errors. In addition, due to the repeated computation of eigenvalues of matrices with poor conditioning, the solver struggles to converge more than with the first order method that appears smoother. It results in a 51% longer runtime. On the other hand, the first order method has the same runtime and $(1-\beta)$–cost as the deterministic method and a conservatism below 1.7 while having a satisfactory failure rate.

5 CONCLUSION

This work presented the L-SODA solver, a novel stochastic solver that uses synergies between applications of DA. It uses automatic differentiation, dynamics approximation, and covariance propagation to efficiently compute robust control laws while maintaining a reasonable runtime.

Two novel transcription methods were tested: a spectral radius method, adapted from existing approaches, and a novel first-order method designed for enforcing robustness while minimizing deviations from optimality. The first order method shows demonstrates strong robustness, achieving a failure rate below the specified threshold with a runtime of under a minute. Moreover, the fuel consumption is similar to the non-robust solver. These results highlight the potential of L-SODA for robust trajectory design.

FUNDING SOURCES

This work was funded by SaCLaB (grant number 2022-CIF-R-1), a research group of ISAE-SUPAERO.

References

[1] R. Armellin, P. Di Lizia, F. Bernelli-Zazzera, and M. Berz. Asteroid close encounters characterization using differential algebra: the case of Apophis. *Celestial Mechanics and Dynamical Astronomy*, 107(4):451–470, 6 2010. doi: 10.1007/s10569-010-9283-5.

[2] B. Benedikter, A. Zavoli, Z. Wang, S. Pizzurro, and E. Cavallini. Convex approach to covariance control with application to stochastic low-thrust trajectory optimization. *Journal*

of Guidance, Control, and Dynamics, 45(11):2061–2075, 2022. doi: 10.2514/1.G006806. URL https://doi.org/10.2514/1.G006806.

[3] J. Benkhoff, G. Murakami, W. Baumjohann, S. Besse, E. Bunce, M. Casale, G. Cremosese, K.-H. Glassmeier, H. Hayakawa, D. Heyner, H. Hiesinger, J. Huovelin, H. Hussmann, V. Iafolla, L. Iess, Y. Kasaba, M. Kobayashi, A. Milillo, I. G. Mitrofanov, E. Montagnon, M. Novara, S. Orsini, E. Quemerais, U. Reininghaus, Y. Saito, F. Santoli, D. Stramaccioni, O. Sutherland, N. Thomas, I. Yoshikawa, and J. Zender. Bepicolombo - mission overview and science goals. *Space Science Reviews*, 217(8), Dec. 2021. ISSN 1572-9672. doi: 10.1007/s11214-021-00861-4.

[4] M. Berz. High-order computation and normal form analysis of repetitive systems. In *AIP Conference Proceedings*, pages 1–36, 1992. doi: 10.1063/1.41975.

[5] M. Berz. *Modern Map Methods in Particle Beam Physics*, chapter 2. Elsevier, 1999. doi: 10.1016/s1076-5670(08)70227-1.

[6] L. Blackmore, M. Ono, and B. C. Williams. Chance-constrained optimal path planning with obstacles. *IEEE Transactions on Robotics*, 27(6):1080–1094, Dec. 2011. ISSN 1941-0468. doi: 10.1109/tro.2011.2161160.

[7] S. Boone and J. Mcmahon. Rapid local trajectory optimization using higher-order state transition tensors and differential dynamic programming. In *2020 AAS/AIAA Astrodynamics Specialist Conference*, number AAS 20-582, Aug. 2020.

[8] S. Boone and J. McMahon. Non-gaussian chance-constrained trajectory control using gaussian mixtures and risk allocation. In *IEEE 61st Conference on Decision and Control (CDC)*, pages 3592–3597, 2022.

[9] A. Boutonnet, Y. Langevin, and C. Erd. Designing the juice trajectory. *Space Science Reviews*, 220(6), Sept. 2024. ISSN 1572-9672. doi: 10.1007/s11214-024-01093-y.

[10] T. Caleb, R. Armellin, and S. Lizy-Destrez. Fast gradient evaluation and mapping method for bi-impulsive transfers within periodic orbits. *Journal of Guidance, Control, and Dynamics*, 0(0):1–9, 2024. doi: 10.2514/1.G008038. URL https://arc.aiaa.org/doi/abs/10.2514/1.G008038.

[11] T. Caleb, R. Armellin, S. Boone, and S. Lizy-Destrez. A polynomial-based constrained solver for fuel-optimal low-thrust trajectory optimization. 2025. URL https://doi.org/10.48550/ARXIV.2502.00398.

[12] T. Caleb, R. Armellin, S. Boone, and S. Lizy-Destrez. Chance constraints transcription and failure risk estimation for stochastic trajectory optimization. 2025. URL https://doi.org/10.48550/arXiv.2502.15949.

[13] W. Folkner, J. Williams, D. Boggs, R. Park, and P. Kuchynka. The Planetary and Lunar Ephemerides DE430 and DE431. Technical Report 42-196, Jet Propulsion Laboratory, California Institute of Technology, 2014.

[14] C. Greco, S. Campagnola, and M. Vasile. Robust space trajectory design using belief optimal control. *Journal of Guidance, Control, and Dynamics*, 45(6):1060–1077, June 2022. ISSN 1533-3884. doi: 10.2514/1.g005704.

[15] T. A. Howell, B. E. Jackson, and Z. Manchester. Altro: A fast solver for constrained trajectory optimization. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–6, 2019. doi: 10.1109/IROS40897.2019.8967788. URL https://doi.org/10.1109/IROS40897.2019.8967788.

[16] G. Lantoine and R. P. Russell. A hybrid differential dynamic programming algorithm for constrained optimal control problems. part 1: Theory. *Journal of Optimization Theory and Applications*, 154(2):382–417, 4 2012. doi: 10.1007/s10957-012-0039-0. URL https://doi.org/10.1007/s10957-012-0039-0.

[17] G. Lantoine and R. P. Russell. A hybrid differential dynamic programming algorithm for constrained optimal control problems. part 2: Application. *Journal of Optimization Theory and Applications*, 154(2):418–442, 8 2012. doi: 10.1007/s10957-012-0038-1. URL https://doi.org/10.1007/s10957-012-0038-1.

[18] T. Lew, R. Bonalli, and M. Pavone. Chance-constrained sequential convex programming for robust trajectory optimization. In *2020 European Control Conference (ECC)*, pages 1871–1878. IEEE, May 2020. doi: 10.23919/ecc51009.2020.9143595.

[19] M. Losacco, A. Fossà, and R. Armellin. Low-order automatic domain splitting approach for nonlinear uncertainty mapping. *Journal of Guidance, Control, and Dynamics*, 47(2): 291–310, Feb. 2024. ISSN 1533-3884. doi: 10.2514/1.g007271.

[20] N. Marmo and A. Zavoli. Chance-constraint method for covariance control of low-thrust interplanetary missions. In *AIAA SCITECH 2024 Forum*. American Institute of Aeronautics and Astronautics, Jan. 2024. doi: 10.2514/6.2024-0630.

[21] N. Marmo, A. Zavoli, N. Ozaki, and Y. Kawakatsu. A hybrid multiple-shooting approach for covariance control of interplanetary missions with navigation errors. In *33rd AAS/AIAA Space Flight Mechanics Meeting*, number AAS 23-208, 01 2023.

[22] D. Mayne. A second-order gradient method for determining optimal trajectories of non-linear discrete-time systems. *International Journal of Control*, 3(1):85–95, 1966. doi: 10.1080/00207176608921369. URL https://doi.org/10.1080/00207176608921369.

[23] K. Oguri and J. W. McMahon. Stochastic primer vector for robust low-thrust trajectory design under uncertainty. *Journal of Guidance, Control, and Dynamics*, 45(1):84–102, Jan. 2022. ISSN 1533-3884. doi: 10.2514/1.g005970.

[24] N. Ozaki, S. Campagnola, R. Funase, and C. H. Yam. Stochastic differential dynamic programming with unscented transform for low-thrust trajectory design. *Journal of Guidance, Control, and Dynamics*, 41(2):377–387, 2018. doi: 10.2514/1.G002367. URL https://doi.org/10.2514/1.G002367.

[25] N. Ozaki, S. Campagnola, and R. Funase. Tube stochastic optimal control for nonlinear constrained trajectory optimization problems. *Journal of Guidance, Control, and Dynamics*, 43(4):645–655, 2020. doi: 10.2514/1.G004363. URL https://doi.org/10.2514/1.G004363.

[26] R. S. Park and D. J. Scheeres. Nonlinear semi-analytic methods for trajectory estimation.

Journal of Guidance, Control, and Dynamics, 30(6):1668–1676, Nov. 2007. ISSN 1533-3884. doi: 10.2514/1.29106.

[27] H. Poincaré. *Les Méthodes Nouvelles de la Mécanique Céleste*. Gauthier-Villars, Paris, France, 1892.

[28] J. Ridderhof, K. Okamoto, and P. Tsiotras. Nonlinear uncertainty control with iterative covariance steering. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 3484–3490. IEEE, Dec. 2019. doi: 10.1109/cdc40024.2019.9029993.

[29] J. Ridderhof, J. Pilipovsky, and P. Tsiotras. Chance-constraints covariance control for low-thrust minimum-fuel trajectory optimization. In *AIAA/AAS Astrodynamics Specialist Conference*, pages 1–20, 2020.

[30] M. Smith, D. Craig, N. Herrmann, E. Mahoney, J. Krezel, N. McIntyre, and K. Goodliff. The Artemis Program: An Overview of NASA's Activities to Return Humans to the Moon. In *2020 IEEE Aerospace Conference*, pages 1–10, 2020. doi: 10.1109/AERO47225.2020.9172323.

[31] Y. L. Tong. *The Multivariate Normal Distribution*, chapter 3. Springer-Verlag New York Inc., 1990. doi: 10.1007/978-1-4613-9655-0. URL `https://doi.org/10.1007/978-1-4613-9655-0`.

[32] F. Topputo and E. Belbruno. Earth–mars transfers with ballistic capture. *Celestial Mechanics and Dynamical Astronomy*, 121(4):329–346, Feb. 2015. ISSN 1572-9478. doi: 10.1007/s10569-015-9605-8.

[33] A. Wittig, P. Di Lizia, R. Armellin, K. Makino, F. Bernelli-Zazzera, and M. Berz. Propagation of large uncertainty sets in orbital dynamics by automatic domain splitting. *Celestial Mechanics and Dynamical Astronomy*, 122(3):239–261, 5 2015. doi: 10.1007/s10569-015-9618-3.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

BAYESIAN CALIBRATION OF A NON-LINEAR CONTROLLED DYNAMIC SYSTEM WITH MISSING DATA : APPLICATION TO HIGH SPEED-TRAIN

R. Jorge Do Marco[1,2,3], G. Perrin[1], C. Funfschilling[2], and C. Soize[3]

[1]COSYS, Université Gustave Eiffel
14-20 Boulevard Newton, 77447 Marne-la-Vallée, France
e-mail: guillaume.perrin@univ-eiffel.fr

[2]SNCF, DTIPG
1-3 Av François Mitterrand, 93574 Saint-Denis, France
e-mail: {romain.jorge-domarco,christine.funfschilling}@sncf.fr

[3]Université Gustave Eiffel, MSME UMR 8208
5 Bd Descartes, 77454 Marne-La-Vallée, France
e-mail: christian.soize@univ-eiffel.fr

Abstract. *As systems become more complex, physical model calibration has become an essential industrial step. In industrial applications, model calibration is often hampered by the presence of unknown environmental variables that affect the behavior of the system. Traditional approaches assume that all input variables are well known, which can lead to significant errors if this assumption is violated. This issue is particularly critical in dynamic systems where inputs and outputs are functional, making direct measurements inherently approximate. To address this challenge, we adopt a Bayesian framework in which a probability distribution is assigned to the unknown environmental variables. Instead of calibrating the model for a single, arbitrarily chosen set of inputs, we compute the posterior distribution of the calibration parameters and average it over the distribution of the unknown variables. This process not only accounts for input uncertainty, but also helps to mitigate model errors, resulting in a more robust and reliable calibration. The effectiveness of this approach is demonstrated in a railway case study, where numerical experiments highlight its advantages over standard deterministic calibration methods.*

Keywords: Bayesian Calibration, Missing data, Dynamic system, Nonlinearity, Railway dynamic

1 INTRODUCTION

Modern engineering increasingly relies on complex physical models, the accuracy of which depends on precise parameter tuning through a process known as calibration. Calibration integrates multiple sources of information, including measurements and expert knowledge, to refine model predictions. However, industrial applications often suffer from prediction errors due to measurement inaccuracies, model incompleteness (epistemic errors) or inherent randomness (aleatory errors) [1]. In this context, uncertainty quantification is crucial for assessing confidence in calibrated models and for tracing the origin of errors. Among existing calibration approaches, the Bayesian framework introduced by Kennedy and O'Hagan [2] provides a powerful way to incorporate uncertainty while combining multiple sources of information. However, its application to dynamical systems presents several challenges. These systems are governed by differential equations, have a high degree of freedom, and involve functional inputs and outputs that are often difficult to measure simultaneously. In practice, missing or imperfectly known inputs are a major obstacle to effective calibration [3]. While missing data problems have been extensively studied in statistics [4], their impact on model calibration, particularly for dynamic systems, remains an open question. This paper extends the framework of Kennedy and O'Hagan by explicitly addressing the challenge of calibrating models with not perfectly known input variables. Our approach is to assign a probability distribution to these partially known inputs and compute expectations over all plausible values. This process not only enhances robustness, but also helps to mitigate model errors by naturally integrating uncertainty propagation. However, this probabilistic treatment significantly increases the computational cost, necessitating a dual optimization strategy: (i) constructing a generator that ensures input-output coherence, and (ii) using surrogate models to maintain computational feasibility. In addition, we analyze how uncertainty propagates throughout the calibration process to improve the reliability of stochastic surrogate models. The proposed methodology is applied to the railway industry, where numerical experiments highlight its advantages over standard deterministic calibration methods.

The outline of this work is as follows. Section 2 presents the theoretical framework associated with the handling of missing data and model discrepancy. An application to the railway industry is then given in Section 3.

2 METHODOLOGY

2.1 Problem definition

In this work, we are interested in estimating the physical properties of a dynamical system of interest based on $n_{\text{meas}} > 1$ independent measurements taken in different environments. For each measurement $i \in [\![1; n_{\text{meas}}]\!]$, let $t_{f,i}$ be the duration of the measurement i, $t \in [0; t_{f,i}]$ be the time, $\mathbf{x}_i := \{x_i(t) \in \mathbb{X} \subset \mathbb{R}^p, t \in [0; t_{f,i}]\}$ be the state vector of the system, $u_i := \{u_i(t), t \in [0; t_{f,i}]\}$ be the system control function, and $\mathcal{T}_i$ be the environmental properties. The function u_i is assumed to belong to $\mathbb{U}(t_{f,i}) := \mathcal{F}([0; t_{f,i}]; [-1; 1])$, which is the set of functions defined on $[0; t_{f,i}]$ whose values are in $[-1; 1]$. For a given control function $u_i \in \mathbb{U}(t_{f,i})$, the state vector $\mathbf{x}$ should verify a deterministic parametric equation given by

$$\forall t \in [0; t_{f,i}], \quad \dot{\mathbf{x}}_i(t) = \mathbf{f}(\mathbf{x}_i(t), u_i(t), \mathcal{T}_i; \mathbf{z}_f). \tag{1}$$

In this equation, the model $\mathbf{f}$ is assumed to be known, but the vector of parameters $\mathbf{z}_f \in \mathbb{Z}_f$ on which it depends is *a priori* unknown. The model $\mathbf{f}$ is considered simplified in the sense that it provides a reasonable representation of the dynamics of the system, without being perfect

(presence of model discrepancies due to modeling simplifications). It is also considered deterministic in the sense that, for given values of $\mathcal{T}_i$ and u_i, and for a given initial state $\mathbf{x}_{0,i} = \mathbf{x}_i(0)$ and a given value of $\mathbf{z}_f$, it yields a unique value for the state function, denoted by

$$\mathbf{x}(\cdot; u_i, \mathbf{z}_f, \mathcal{T}_i, \mathbf{x}_{0,i}) := \begin{cases} [0; t_{f,i}] & \to & \mathbb{X} \\ t & \mapsto & \mathbf{x}(t; u_i, \mathbf{z}_f, \mathcal{T}_i, \mathbf{x}_{0,i}) \end{cases}. \tag{2}$$

The system dynamics must also satisfy a set of constraints, which can be represented by an envelope $\mathbb{X}_i^{\text{valid}}$ containing lower and upper bounds for each component of $\mathbf{x}_i$ at each time t. The performance evaluation of the system under consideration involves another real-valued function h, which also depends on $\mathbf{x}_i$, u_i and $\mathcal{T}_i$. The function h is assumed to be known, but may additionally depend on parameters that are not perfectly known, denoted by $\mathbf{z}_h \in \mathbb{Z}_h$. We then denote by e_i the integral in time of this quantity of interest, so that for each measurement $1 \leq i \leq n_{\text{meas}}$ and each $t \in [0; t_{f,i}]$,

$$e_i(t) = \int_0^t h(\mathbf{x}(\tau; u_i, \mathbf{z}_f, \mathcal{T}_i, \mathbf{x}_{0,i}), u_i; \mathbf{z}_h) d\tau. \tag{3}$$

Similar to the state vector, there may be constraints on the values of h in the form of time-dependent lower and upper bounds. In the same way as for the dynamics model, the function h can also be affected by a model discrepancy due to modeling simplifications. The quantity e_i, as a composition of the models h and $\mathbf{f}$, is thus *a priori* affected by two different model errors, whose amplitudes are characterized by two scalars σ_f and σ_p (the precise definitions of these scalars, which depend on the application considered, are given in Section 3.1). We also introduce $\boldsymbol{\sigma} = (\sigma_f, \sigma_p) \in \mathbb{S}$ as the vector of hyperparameters characterizing the model errors.

The aim of this work is to propose a method for estimating the value of $\mathbf{z} := (\mathbf{z}_f, \mathbf{z}_h) \in \mathbb{Z} := \mathbb{Z}_f \times \mathbb{Z}_h$ as accurately as possible from point measurements of $\mathbf{x}_i$ and e_i for each $i \in [\![1; n_{\text{meas}}]\!]$ in the context of missing data, in the sense that the control values u_i applied during the n_{meas} measurements are imperfectly known.

2.2 Bayesian framework

In this paper we adopt a Bayesian framework for estimating the vector $\mathbf{z}$. It is therefore modeled as a random vector $\mathbf{Z}$, and we denote its Probability Density Function (PDF) by $f_{\mathbf{Z}}$. Obtaining this PDF depends on the prior available information about this vector [1, 5], an example is given in Section 3. In addition to this prior knowledge, we have access to discrete and noisy observations of the system behavior, which are assumed to be gathered in the vector $\mathbf{y} \in \mathbb{Y}$. Estimating $\mathbf{Z}$ thus amounts to approximating the PDF of $\mathbf{Z}|\mathbf{Y} = \mathbf{y}$, which can be written as follows using Bayes' theorem:

$$f_{\mathbf{Z}|\mathbf{Y}=\mathbf{y}}(\mathbf{z}) = c \times f_{\mathbf{Z}}(\mathbf{z}) \times f_{\mathbf{Y}|\mathbf{Z}=\mathbf{z}}(\mathbf{y}), \tag{4}$$

where c is a constant that does not depend on $\mathbf{z}$ and $f_{\mathbf{Y}|\mathbf{Z}=\mathbf{z}}$ is the likelihood function. To make this likelihood function explicit, let us denote by $\mathbf{y}^{\text{mod}}(\mathbf{z}, \mathbf{u})$ the vector of predictions of the quantities $\mathbf{y}$ that can be made by evaluating the codes $\mathbf{f}$ and h. This vector depends on the vector $\mathbf{z}$ to be identified, but also requires knowledge of the various control functions u_i applied during each measurement, which are grouped together in the command list $\mathbf{u}$. The link between the observations and the code predictions is then written:

$$\mathbf{y} = \mathbf{y}^{\text{mod}}(\mathbf{z}, \mathbf{u}) + \boldsymbol{\varepsilon}^{\text{meas}} + \boldsymbol{\varepsilon}^{\text{mod}}(\boldsymbol{\sigma}), \tag{5}$$

where $\varepsilon^{\text{meas}}$ is the measurement noise, modeled as a random vector with known PDF, and $\varepsilon^{\text{mod}}(\boldsymbol{\sigma})$ is the model error, also modeled as a random vector whose PDF is chosen in a parametric family indexed by $\boldsymbol{\sigma}$. Since this vector $\boldsymbol{\sigma}$ is *a priori* unknown, we also propose to model it as a random vector noted $\mathbf{S}$ with prior PDF $f_{\mathbf{S}}$. It comes:

$$f_{\mathbf{Z}|\mathbf{Y}=\mathbf{y}}(\mathbf{z}) = c \times f_{\mathbf{Z}}(\mathbf{z}) \times \mathbb{E}_{\mathbf{S}|\mathbf{Z}=\mathbf{z}}\left[f_{\mathbf{Y}|\mathbf{Z}=\mathbf{z},\mathbf{S}}(\mathbf{y})\right]. \tag{6}$$

The peculiarity of this work also stems from the fact that the commands gathered in $\mathbf{u}$ are only partially known. We therefore propose to model them again as random quantities, in order to incorporate their uncertain nature into the calibration process. We then denote by $\mathbf{U}$ the list of random commands and we assume that we are capable of generating independent realizations of U. This allows us to adapt Eq. (6) to obtain

$$f_{\mathbf{Z}|\mathbf{Y}=\mathbf{y}}(\mathbf{z}) = c \times f_{\mathbf{Z}}(\mathbf{z}) \times \mathbb{E}_{\mathbf{U}|\mathbf{Z}=\mathbf{z}}\left[\mathbb{E}_{\mathbf{S}|\mathbf{Z}=\mathbf{z},\mathbf{U}}\left[f_{\mathbf{Y}|\mathbf{Z}=\mathbf{z},\mathbf{U},\mathbf{S}}(\mathbf{y})\right]\right], \tag{7}$$

where for each couple of realizations $(\mathbf{u}, \boldsymbol{\sigma})$ of $(\mathbf{U}, \mathbf{S})$, it is now possible to calculate the PDF $f_{\mathbf{Y}|\mathbf{Z}=\mathbf{z},\mathbf{U}=\mathbf{u},\mathbf{S}=\mathbf{s}}(\mathbf{y})$ by evaluating the numerical codes $\mathbf{f}$ and h at each measurement. However, the presence of this double expectation makes it very difficult to evaluate $f_{\mathbf{Z}|\mathbf{Y}=\mathbf{y}}$ at any value of $\mathbf{z}$. Significant adaptations and simplifications (by combining Markov Chain Monte Carlo (MCMC) (see [1]) approaches with machine learning approaches such as Gaussian process regression [6]) have therefore to be proposed to obtain a set of values of $\mathbf{z}$ that can be assumed to correspond to a sample of identical and identically distributed (i.i.d.) realizations of $\mathbf{Z}$ according to $f_{\mathbf{Z}|\mathbf{Y}=\mathbf{y}}$.

3 IDENTIFICATION OF THE PARAMETERS OF A HIGH-SPEED TRAIN

3.1 Context

This work focuses on a railway application, and we are interested in calibrating two simplified models (see [7, 8] for their precise definition): a dynamic model $\mathbf{f}$ and an energetic model h. Each of these models is parameterized, and we note $\mathbf{z}$ the vector of parameters that needs to be specified for these models to be evaluated. Vector $\mathbf{z}$ is made of 9 components,

$$\mathbf{z} = (m, a, b, c, a_\eta, b_\eta, c_\eta, d_\eta, p_{\text{aux}}), \tag{8}$$

where m is the mass of the train, a, b, and c are Davis coefficients (see [9]), a_η, b_η, c_η and d_η are yield coefficients and p_{aux} is the train auxiliary power. Applying the maximum entropy principle (see [1]), it is possible to associate prior PDFs to the components of $\mathbf{z}$, which are shown in Figure 1. The two mechanical models are assumed to be affected by model errors, which are modeled by two Gaussian white noises of amplitudes σ_f and σ_p respectively: σ_f is the standard deviation of the noise for the train dynamical efforts, and σ_p is the standard deviation of the noise affecting the modeling of the electric power. These two errors are gathered in $\boldsymbol{\sigma}$, and we also associate a prior PDF to this vector (we are concentrating here on Jeffreys' priors [10] to minimize the impact of this choice on the calibration results).

In this work, we focus on a numerical experiment using synthetic data. We consider 20 measurements over several railway tracks and several running environments for the train. For these different measurements, simulated values of train dynamics and energy consumption are then calculated for a reference value of $\mathbf{z}$ noted $\mathbf{z}^*$. From these simulations, we extract the train speed and the power consumed by the train as a function of the curvilinear abscissa of the track at a discretization step of $0.2s$. The reference values for these quantities are then noised to give

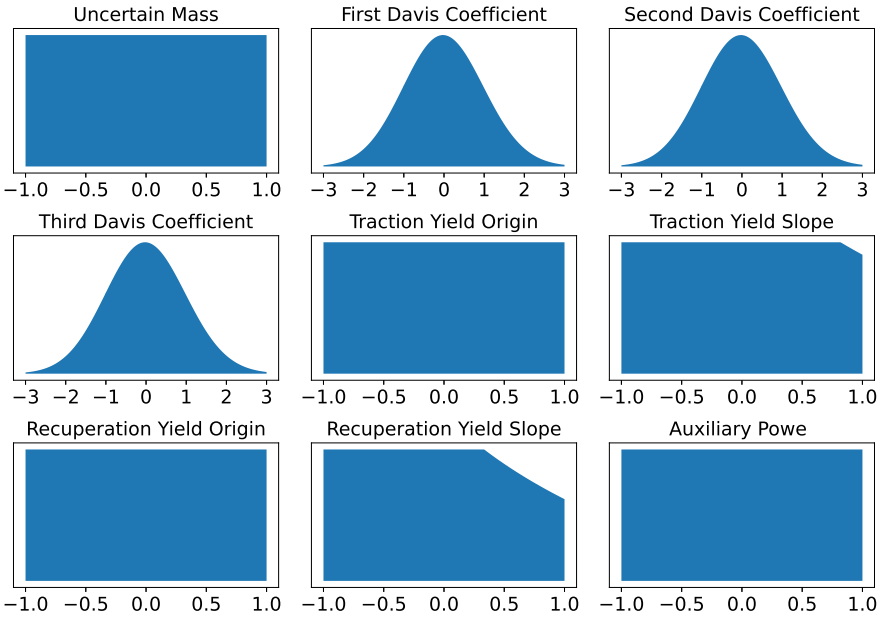


Figure 1: Normalized prior of $\mathbf{Z}$.

the observations available for calibrating the model parameters. Half of these measurements (the first ten) are used for the calibration phase, while the other half (the last ten measurements) will be used to assess the relevance of the results obtained. The observations associated with the first ten measurements are gathered in the vector $\mathbf{y}$. In our case, the model error arises from a difference in the discretization of the differential equations modeling the movement of the train. Whereas the reference values are based on a Runge Kutta numerical scheme of order 4 with a step of $0.01s$, the model we are trying to calibrate is solved by a Runge Kutta numerical scheme of order 4 but with a step of $0.1s$.

3.2 Control definition

The dynamic models are highly sensitive to the choice of the driver commands u_i. In our case, a value of $u_i(t) = 1$ indicates maximum mechanical traction, while a value of $u_i(t) = -1$ indicates maximum braking. For this work, we assume that we only have a discretization of these commands every $0.2s$ (using the same time step as the speed and power measurements). This discretization step does not exactly match the command changing instants (it is nevertheless assumed that by discretizing every $0.2s$, no command changes are missed). We would therefore like to quantify the influence of this loss of information on the estimate of $\mathbf{z}^*$. To this end, we study three cases:

1. We call *true control* the case where we assume that the value of the driver commands is known. This configuration will be used as a reference for the results.

2. We call *unique control* the case where we assume that the true control is exactly given by a constant piecewise interpolation of the available discretizations of the u_i.

3. We call *Generator control* the case where we do not consider a single control in the inversion, but a collection of controls, all associated with control change times uniformly distributed over the $0.2s$ separating the potential control jumps (see Figure 2 for an illustration).

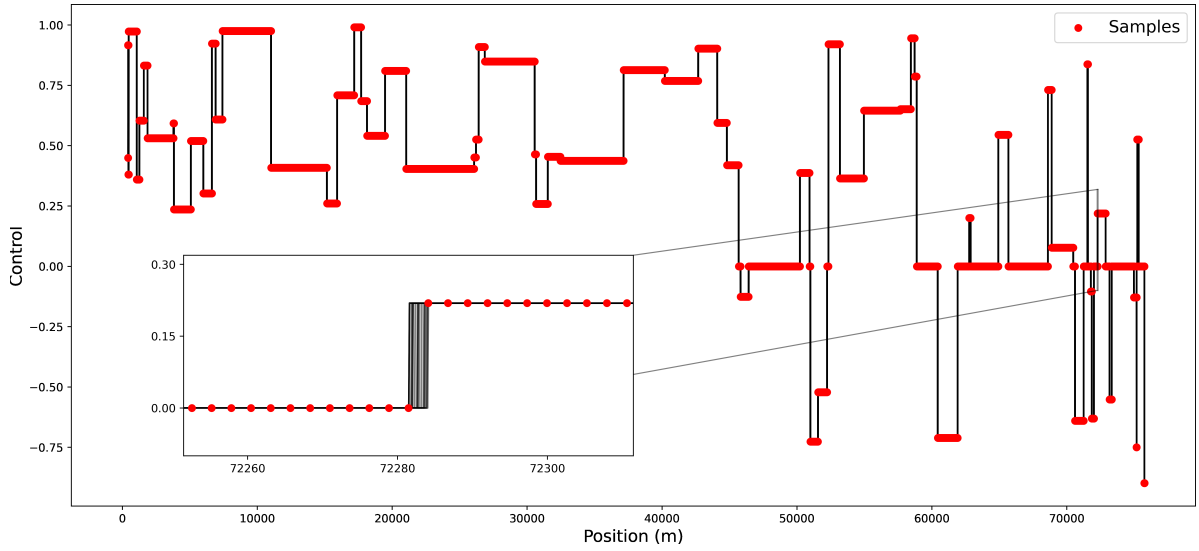


Figure 2: Representation of several command functions (solid black lines) associated with a same discretization (red dots) of the true but unknown driver command.

Note that the two first cases are specific draws of the third case.

3.3 Results

The *a posteriori* PDF of the model parameters was calculated independently in the three configurations listed above using Markov Chain Monte Carlo (MCMC) algorithms. The evaluation of the results is based on several criteria:

1. The Mean Square Error (MSE) of $\mathbf{Z}^{\text{post}} = (\mathbf{Z}|\mathbf{Y} = \mathbf{y})$ is noted

$$\delta_{\text{mse}} = \mathbb{E}\left[\|\mathbf{Z}^{\text{post}} - \mathbf{z}^*\|_2^2\right] \tag{9}$$

   and assesses the extent to which the obtained *a posteriori* PDF is focused on true reference value $\mathbf{z}^*$;

2. The estimated amplitudes of the model error terms, which are defined such that

$$\boldsymbol{\sigma}^* = \arg\left(\max_{\boldsymbol{\sigma}\in\mathbb{S}}\left(\max_{\mathbf{z}\in\mathbb{Z}}\left(f_{\mathbf{Z},\mathbf{S}|\mathbf{Y}=\mathbf{y}}(\mathbf{z},\boldsymbol{\sigma})\right)\right)\right), \tag{10}$$

   allow us to quantify the impact on the mechanical models of not perfectly knowing the true command;

3. The reconstruction errors are given by

$$\forall i \in [\![1;10]\!], \quad \delta^i_{\text{rec,tot}} = \frac{\|\dot{s}_{\text{true},i} - \mathbb{E}_{\mathbf{Z}}\left[\dot{s}\left(\mathbf{Z},u_i^*\right)\right]\|_2^2}{\|\dot{s}_{\text{true},i}\|_2^2} + \frac{\|\dot{e}_{\text{true},i} - \mathbb{E}_{\mathbf{Z}}\left[\dot{e}\left(\mathbf{Z},u_i^*\right)\right]\|_2^2}{\|\dot{e}_{\text{true},i}\|_2^2}, \tag{11}$$

   where for each measurement i that was not used during the calibration phase, $\dot{s}_{\text{true},i}$ and $\dot{e}_{\text{true},i}$ are the true train speed and true consumed electrical power (that is to say associated

with the true control function u_i^* and the true model parameters $\mathbf{z}^*$, whereas $\dot{s}\left(\mathbf{z}, u_i^*\right)$ and $\dot{e}\left(\mathbf{z}, u_i^*\right)$ are the train speed and consumed electrical power we obtain when running the models by setting the model parameters to $\mathbf{z}$.

The results are presented in Table 1 and Figure 3. They show that averaging over the control (*Generator control*) allows a better calibration with smaller model errors, a smaller MSE and a smaller reconstruction error. Surprisingly, we also note that we have a lower value for σ_p^* for the *unique control* case compared to the *true control* case: this is mainly due to the fact that the error in the effort (mainly governed by σ_f) is also propagated in the electrical power. Thus, the increased value of σ_f^* may cause a small decrease in σ_p^*. Behind a smaller model error we expect a better identification of the parameters. This is the case as shown in Table 1 (note that the high values of the MSE obtained for the *generator control* and the *unique control* cases are led by a unique poorly identified model parameter). According to Figure 3, we also observe that the *generator control* case leads to reconstruction results that are between the ones associated with the *unique control* and the *true control* cases, which is exactly what we wanted to do. Indeed, by averaging over the driver controls, we expect to reduce the model errors that need to be introduced to match the measurements to the simulations, and through this, we expect a more accurate estimation of the model parameters.

Metric	**True control**	**Generator**	**Unique control**
δ_{mse}	1.18	5.34	5.72
σ_f^*	3.07	3.10	3.45
σ_p^*	5.89	5.87	5.88

Table 1: Synthesis of the results obtained for the MSE (obtained by averaging over 50 000 individuals drawn from the posterior) and for the estimated model errors.

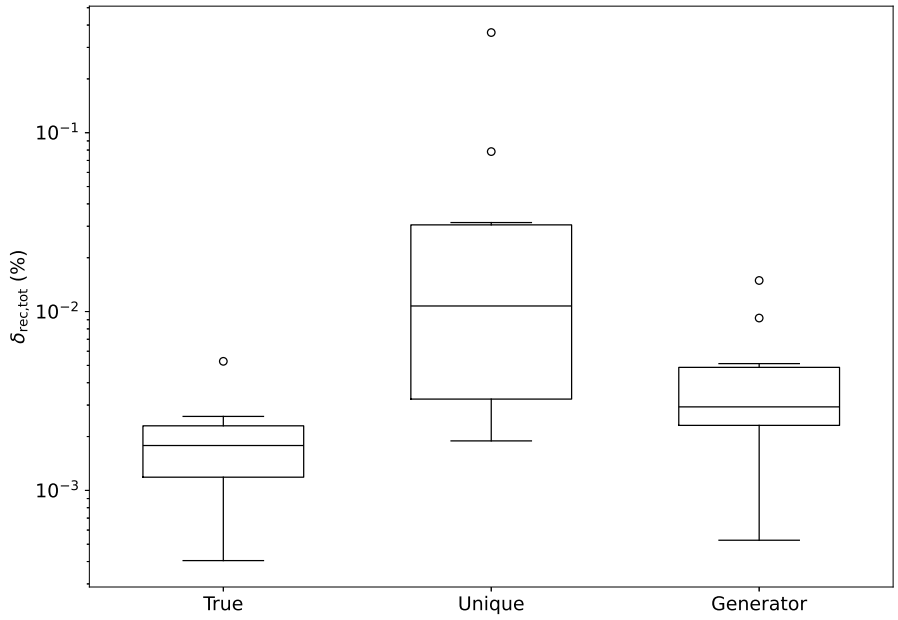


Figure 3: Reconstruction errors obtained for both speed profiles and power profiles for the 10 measurements that were not used in the calibration process.

4 CONCLUSIONS

In conclusion, this work proposes an application of the Kennedy and O'Hagan framework to handle configurations in presence of missing data. By considering, in the model calibration phase, a control generator rather than focusing on a unique control, we better explain the model variability and thus reduce the model errors. Furthermore, we also show that the reconstruction error is reduced when a control generator is used. As a perspective for this work, we could study the influence of the generator definition. Indeed, in our case the lack of knowledge is limited. It could be useful to study cases where the lack of knowledge increases and tends towards the case of completely unmeasured controls.

References

1. Soize, C. *Uncertainty quantification: An accelerated course with advanced applications in computational engineering* ISBN: 978-3-319-85372-7 (Springer International Publishing, Cham, Switzerland, 2018).
2. Kennedy, M. C. & O'Hagan, A. Bayesian calibration of computer models. en (2001).
3. Perrin, G. & Durantin, C. Taking into account input uncertainties in the Bayesian calibration of time-consuming simulators. *Journal de la société française de statistique* **160,** 24–46 (2019).
4. Little, R. J. A. & Rubin, D. B. *Statistical analysis with missing data* 3rd ed. ISBN: 978-0-470-52679-8 (John Wiley & Sons, Nashville, TN, 2019).
5. Robert, C. P. *The Bayesian choice: from decision-theoretic foundations to computational implementation* 2nd ed. en. ISBN: 978-0-387-71598-8 (Springer, New York, 2007).
6. Rasmussen, C. E. & Williams, C. K. I. *Gaussian processes for machine learning* OCLC: ocm61285753. ISBN: 978-0-262-18253-9 (MIT Press, Cambridge, Mass, 2006).
7. Jorge Do Marco, R., Perrin, G., Funfschilling, C. & Soize, C. *Constrained Optimization of Driver Control to Limit Energy Consumption* in *Proceedings of the Sixth International Conference on Railway Technology: Research, Development and Maintenance* (ed Pombo, J.) **CCC 7** (Civil-Comp Press, Edinburgh, UK, 2024), Paper 21.3.
8. Nespoulous, J., Soize, C., Funfschilling, C. & Perrin, G. Optimisation of train speed to limit energy consumption. en. *Vehicle System Dynamics* **60,** 3540–3557. ISSN: 0042-3114, 1744-5159. https://www.tandfonline.com/doi/full/10.1080/00423114.2021.1965628 (2023) (Oct. 2022).
9. Davis, W. J. *The tractive resistance of electric locomotives and cars* en (General Electric, 1926).
10. Jeffreys, H. An invariant form for the prior probability in estimation problems. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* **186.** _eprint: https://royalsocietypublishing.org/doi/pdf/10.1098/rspa.1946.0056, 453–461. https://royalsocietypublishing.org/doi/abs/10.1098/rspa.1946.0056 (1946).

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
on Uncertainty Quantification in Computational Science and Engineering
M. Papadrakakis, M. Fragiadakis (eds.)
Rhodes, Greece, 15-–18 June 2025

VIRTUAL LOAD MONITORING OF OFFSHORE WIND FARMS VIA UNCERTAINTY-AWARE DEEP LEARNING MODELS

Nandar Hlaing[1], Pablo G. Morato[2], Francisco de N. Santos[1], Wout Weijtjens[1], and Christof Devriendt[1]

[1]OWI-Lab, Vrije Universiteit Brussel
1050 Brussels, Belgium
e-mail: {nandar.hlaing, francisco.de.nolasco.santos, wout.weijtjens, christof.devriendt}@vub.be

[2] Faculty of Architecture and the Built Environment, Delft University of Technology
2628 BL Delft, Netherlands
e-mail: p.g.moratodominguez@tudelft.nl

Abstract. *This work explores probabilistic deep learning models as offshore farm-wide virtual load sensors, including Bayesian neural networks, Monte Carlo dropout, and deep neural network ensembles. The aim is to develop models offering uncertainty-aware predictions of damage equivalent loads using SCADA and accelerometer data. This study uses the data from a Belgian offshore wind farm's five fleet-leader turbines. After training the neural networks with one year of collected data, these models are deployed to another year, facing out-of-distribution data due to changes in operational conditions. The analysis assesses generalization and uncertainty quantification abilities, providing insights into their strengths and weaknesses. Ultimately, our work supports the industrial adoption of probabilistic deep learning virtual monitoring models, enabling informed asset management decisions with predictions and uncertainty measures.*

Keywords: Structural health monitoring, virtual load monitoring, probabilistic deep learning, uncertainty quantification, offshore wind farms.

1 INTRODUCTION

Benefiting from the continuous improvement of sensor technologies, learning algorithms, and high-performance computational hardware, data-driven solutions for the assessment and management of offshore wind assets are increasingly being adopted. In particular, the global approximation qualities of deep neural networks can be leveraged to effectively extract complex patterns from large datasets, rendering deep learning approaches highly useful in wind energy applications—ranging from wind speed and direction forecasting [Mbuvha et al., 2021] to fault detection [Cho et al., 2021] and structural health monitoring [Hlaing et al., 2022], among others.

Deterministic neural networks do not, however, explicitly capture model uncertainty, which is a crucial factor in the safety and reliability of offshore wind assets exposed to stochastic loads and stressors. Since environmental and operational conditions vary over time, already trained deep learning models may receive out-of-distribution data in practical scenarios, i.e., data unseen by the model during the training stage. In other scenarios, such as farm-wide load monitoring, prediction models trained on one or multiple fleet leaders may be applied to other wind turbines, some of which may belong to different design clusters. When load measurements are unavailable to validate model predictions, deterministic estimates do not provide sufficient information for effective decision-making.

On the other hand, probabilistic neural networks explicitly provide model uncertainty indicators [Pimenta et al., 2022, Zou et al., 2023, Singh et al., 2024]. It has been shown that probabilistic models based on Bayesian principles are capable of intrinsically reporting higher model uncertainties when tested on out-of-distribution data, for which the predictions are likely to be inaccurate [Hlaing et al., 2024]. This capability is of importance for the safe deployment of deep learning models in practical decision-making applications— such as inspection and maintenance planning, lifetime extension— where the quantification of imperfect information plays a vital role [Hlaing, 2024].

In this work, we investigate the effectiveness of probabilistic deep learning models as farm-wide virtual load sensors. Specifically, we compare probabilistic neural network variants, including Monte Carlo dropout, Bayes by backprop, and neural network ensembles. The primary objective of this comparative study is to identify models that not only effectively predict damage equivalent loads but also capture the uncertainty associated with model predictions. Such uncertainty-aware models can support informed decision-making in the dynamic and uncertain environment characteristic of offshore wind farms. Our analysis is based on data collected over 2 years from five fleet-leader turbines (i.e., turbines equipped with SCADA system and strain gauges) at a Belgian offshore wind farm. Data collected from the first year is used to train the neural networks, while data collected from the second year— during which the turbines operate under different control conditions— is used for testing. This scenario introduces a distribution shift between the training and testing datasets. Our evaluation focuses on the generalization and uncertainty quantification of the tested probabilistic neural networks. Throughout the analysis, we highlight the strengths and limitations of each model. The findings of this study are expected to accelerate the industrial adoption of probabilistic deep learning models for virtual load monitoring.

2 PROBABILISTIC NEURAL NETWORKS

2.1 Bayesian neural networks

A Bayesian neural network (BNN) is an artificial neural network that integrates deep learning with Bayesian inference principles to generate probabilistic predictions. Unlike deterministic artificial neural networks (ANNs), the distinctive feature of BNNs lies in their stochastic neural network components, including stochastic weights and/or biases. Stochastic model parameters, $\boldsymbol{\theta}$, each characterized by a probability distribution $P(\boldsymbol{\theta})$, encode multiple possible model parametrizations, thus capturing the model's epistemic uncertainty. BNNs are more interpretable than their deterministic counterparts, as they signal high model uncertainty when making predictions for samples that are far from those contained in the training dataset. This feature makes them particularly useful in scenarios with limited data availability.

BNNs conduct Bayesian inference to estimate the posterior distribution of model parameters given the training data. Conditioned to the training dataset D, the posterior $P(\boldsymbol{\theta} \mid D)$ can be computed via Bayes' theorem:

$$P(\boldsymbol{\theta} \mid D) = \frac{P(D \mid \boldsymbol{\theta})P(\boldsymbol{\theta})}{\int P(D \mid \boldsymbol{\theta})P(\boldsymbol{\theta})d\boldsymbol{\theta}}. \tag{1}$$

Computing the posterior distribution is intractable in practical high-dimensional scenarios. To overcome this limitation, approximate inference methods have been proposed in the literature, e.g., stochastic gradient Markov Chain Monte Carlo [Welling and Teh, 2011, Zhang et al., 2019] and variational methods [Graves, 2011].

2.1.1 Variational inference

Variational inference (VI) is a technique used in Bayesian statistics to approximate complex posterior distributions [Graves, 2011, Osawa et al., 2019]. In VI, a simpler distribution, known as the variational distribution, is used to approximate the posterior distribution of model weights. The main objective of VI is to derive a variational distribution $q_\lambda(\boldsymbol{\theta})$ as close as possible to the true unknown posterior $P(\boldsymbol{\theta} \mid D)$. The similarity between both distributions is measured using the Kullback-Leibler (KL) divergence, which quantifies the non-symmetric statistical difference between the two distributions. The KL divergence between $q_\lambda(\boldsymbol{\theta})$ and $P(\boldsymbol{\theta} \mid D)$ can be mathematically expressed as:

$$\mathbb{KL}\left(q_\lambda(\boldsymbol{\theta}) \mid\mid P(\boldsymbol{\theta} \mid D)\right) = \int q_\lambda(\boldsymbol{\theta}) \log \frac{q_\lambda(\boldsymbol{\theta})}{P(\boldsymbol{\theta} \mid D)} d\boldsymbol{\theta}. \tag{2}$$

VI seeks to find the variational parameters λ that minimize the KL divergence formulated above, resulting in the maximizatioin of the evidence lower bound objective (ELBO) function:

$$\text{ELBO} = -\int q_\lambda(\boldsymbol{\theta}) \log \frac{q_\lambda(\boldsymbol{\theta})}{P(\boldsymbol{\theta})} d\boldsymbol{\theta} + \int q_\lambda(\boldsymbol{\theta}) \log P(D \mid \boldsymbol{\theta}) d\boldsymbol{\theta}. \tag{3}$$

The ELBO loss includes two terms: a regularization term enforcing consistency with prior knowledge, and an expected log-likelihood term controlling that the model fits the training data. The relative importance of each term in the loss can be adjusted to optimize model performance. During training, the negative log-likelihood can be computed by drawing only one neural network sample per forward pass, rather than computing the expectation over multiple model realizations, thus reducing computational effort. Gradient descent can still be effectively performed despite the presence of noisy gradient estimates.

Bayes by backprop

Bayes by backprop follows the aforementioned VI approach to approximate the posterior distribution over neural network weights [Blundell et al., 2015]. This approach combines the principles of Bayesian inference with the optimization techniques of backpropagation. By assuming the variational posterior as a multivariate standard Gaussian, i.e, $q_\lambda \sim \mathcal{N}(\mathbf{0}, I)$, Bayes by backprop uses the reparametrization trick:

$$\boldsymbol{\theta} = \mu_{\boldsymbol{\theta}} + \sigma_{\boldsymbol{\theta}} \cdot \varepsilon, \text{ where } \varepsilon \sim \mathcal{N}(0, 1). \tag{4}$$

Random weights θ are sampled from the variational distribution using the reparameterization trick. This formulation also enables the computation of the gradients and updates of the variational parameters using standard optimization techniques. Following Equation 3, the model loss at the k^{th} epoch is can be computed as:

$$\mathcal{L} = \alpha \sum_{j=1}^{N_\theta} \mathbb{KL}\left(q_\lambda(\theta_k^j) \mid\mid P(\theta_0^j)\right) + \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2, \tag{5}$$

where α is the KL loss factor, N_θ is the number of model parameters, N is the size of training dataset, y and $\hat{y}$ are the target labels and model predictions.

Monte Carlo dropout

Dropout is a regularization technique where, during training, random units (neurons) are "dropped out" or set to zero with a certain probability. This helps prevent overfitting and encourages robustness of the model [Wager et al., 2013]. Monte Carlo dropout (MC-Dropout) is an approach that extends the idea of dropout regularization in neural networks to approximate Bayesian inference [Gal and Ghahramani, 2016]. During the training of MC-Dropout networks, a dropout rate $p_d \in [0, 1]$ is applied, randomly deactivating a portion of neurons in each forward and backward pass, introducing stochasticity to the model. The loss function can be formulated as:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 + \phi\|\boldsymbol{\theta}\|_2^2. \tag{6}$$

in which the second term corresponds to L2 regularization and ϕ is the regularization coefficient. During the forward prediction of BNNs, dropout is still applied, being opposed to the traditional dropout in which dropout is only applied during training, but all neurons are active during prediction. Multiple forward passes are performed through the network, each time randomly deactivating neurons with a specified dropout probability. This process is similar to drawing multiple samples from a probabilistic model. MC-Dropout provides a practical and efficient way to obtain uncertainty estimates in neural network predictions and can be seen as an approximation of Bayesian neural networks, while full Bayesian inference is computationally expensive.

2.2 Deep ensembles

Deep ensemble networks, also known as model ensembles, are a technique in machine learning that involves training multiple deep neural networks independently and then combining their predictions to produce more robust and accurate predictions. Since they do not apply Bayesian principles, deep ensembles may not be categorized as Bayesian neural networks. The key idea

behind deep ensemble networks is to leverage the diversity of multiple models to improve generalization and reduce overfitting, standing as a sound reference for comparison.

Instead of training a single deep neural network, multiple networks are trained independently through a different random initialization and/or different training hyperparameters. Once all the individual networks are trained, they are used to make predictions on new, unseen data points. Each network produces its own prediction which are then combined to produce a final prediction. Mathematically, if N_n individual neural networks are considered, each producing a prediction denoted as $\hat{y}_i$ for the same input data point, the ensemble prediction $\hat{y}_{\text{ensemble}}$ can be computed as:

$$\hat{y}_{\text{ensemble}} = \sum_{i=1}^{N_n} w_i \hat{y}_i \tag{7}$$

where w_i represents the importance weight assigned to each neural network model's prediction. The weights can be uniform (equal for all models) or learned during the training process to optimize the performance of the ensemble.

3 BENCHMARKING PROBABILISTIC NEURAL NETWORKS AS VIRTUAL SENSORS FOR OFFSHORE WIND FARMS

Typically, equipping an entire offshore wind farm with strain sensors is not cost-effective, primarily due to the high installation and maintenance expenses associated with this process. In this context, virtual load monitoring offers a practical solution. It involves generating load-related information using readily accessible monitoring data, which often includes SCADA and accelerometer data. In this regard, one or a selected group of turbines, often called the 'fleet-leader(s)', are fully equipped, allowing the collection of concurrent SCADA, acceleration, and load data required for the training of virtual models. The trained model can later be used to replace the damaged strain sensors of the fleet-leader(s) or predict the loads on other turbines in the farm.

In this study, we compare the afore-mentioned probabilistic neural network models with a specific focus on fleet-leader-based virtual monitoring application. This benchmark aims to assess these methods in terms of generalization ability and robustness of uncertainty indications. In addition, the computational requirements for the training and deployment of the models are also compared.

3.1 Uncertainty estimation in probabilistic neural networks

The collected predictions from multiple passes of the probabilistic neural networks form a distribution of possible outcomes. One can calculate various statistics of this distribution from the predicted samples. For instance, the expected value and the predictive uncertainty can be quantified as:

$$\mathbb{E}[\hat{y} \mid \boldsymbol{x}] = \frac{1}{N_f} \sum_{i=1}^{N_f} f\left(\hat{y} \mid \boldsymbol{x}, \boldsymbol{\theta}^{(i)}\right), \tag{8}$$

$$\mathbb{V}(\hat{y} \mid \boldsymbol{x}) = \frac{1}{N_f} \sum_{i=1}^{N_f} \left(f\left(\hat{y} \mid \boldsymbol{x}, \boldsymbol{\theta}^{(i)}\right)\right)^2 - \left(\mathbb{E}[\hat{y} \mid \boldsymbol{x}]\right)^2, \tag{9}$$

where the network parameters, $\boldsymbol{\theta}$ are random samples from the posterior weights and biases associated distributions, f represents the probabilistic neural network model, and N_f is the number of forward predictions.

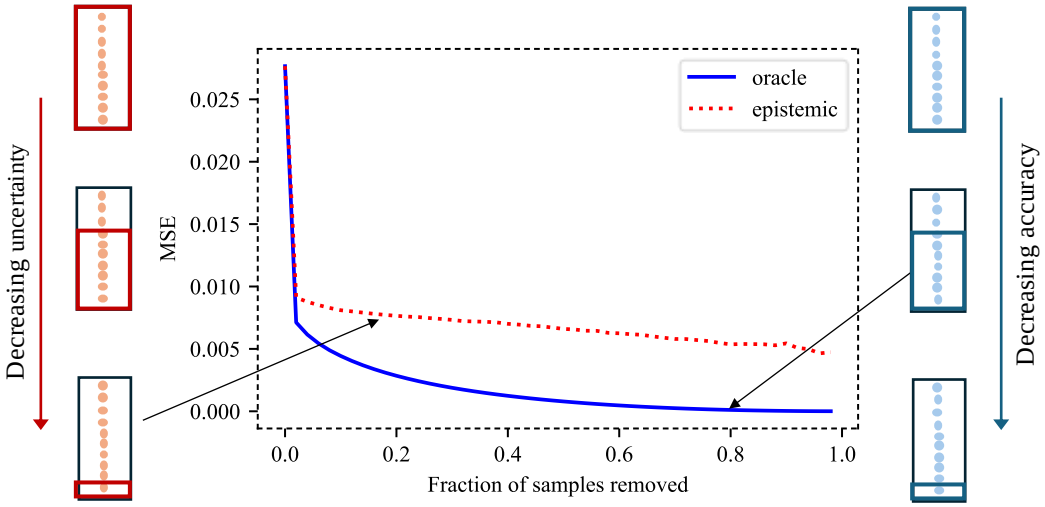


Figure 1: Illustration of an uncertainty-error sparsification plot.

If the output of neural network models is a distribution, i.e., aleatory uncertainty is also considered, the retrieved predictive uncertainty estimate $\mathbb{V}(\hat{y} \mid \boldsymbol{x})$ encompasses both aleatory and epistemic contributions. If the network is modeled to produce point estimates only, the predictive uncertainty solely represents the epistemic part. For a detailed explanation on the uncertainty decomposition of Bayesian neural networks, the reader is referred to [Hlaing et al., 2024].

3.2 Metrics for comparing probabilistic models

To compare the model uncertainty provided by the tested probabilistic neural networks, we evaluate the coverage probability, represent sparsification plots, and quantify the AUSE (area under sparsification error) metric. In order to compute the coverage probability, multiple forward simulations are generated for each test input. From the simulated predictions, we estimate the 2.5^{th} and 97.5^{th} percentiles and check if the target value lies within the interval. The coverage probability on the test dataset can thus be computed as:

$$CP = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I} \left[\hat{y}_i^{(2.5)} \leq y_i \leq \hat{y}_i^{(97.5)} \right] \tag{10}$$

where $\hat{y}_i^{(2.5)}$ and $\hat{y}_i^{(97.5)}$ are the $2.5^{th}$and 97.5^{th} quantiles of the predicted samples for observation i, N is the total number of test observations, and $\mathbb{I}[.]$ is an indicator function that returns 1 if the condition inside is true and 0 otherwise.

Sparsification plots have already been used in computer vision to visualize and compare different probabilistic neural network models with respect to their performance in estimation tasks while considering uncertainty estimates [Gustafsson et al., 2020, Ilg et al., 2018]. The vertical axis in a sparsification plot is the metric of accuracy, e.g., mean absolute error, mean squared error, etc., and the percentage of removed samples is indicated on the horizontal axis. Sparsification plots are obtained through step-by-step removal of test samples and re-assessing the model accuracy, as illustrated in Figure 1. The *oracle* curve which eliminates the samples in a decreasing order of the model accuracy is always monotonic and used as the reference. The

epistemic curve plots the model accuracy of test samples sorted from high to low uncertainty. Note that when the model uncertainty is biased- e.g., over-confidence for wrong predictions, the epistemic curve can be non-monotonic. The closer the epistemic curve aligns with the oracle curve, the better the model's uncertainty ranking.

Subsequently, AUSE - the area between the oracle and epistemic curves - is a performance metric used to quantitatively evaluate the quality of uncertainty estimation. A lower AUSE value indicates that the model-provided uncertainties align with the error with respect to ground truths. For non-monotonic epistemic curves, AUSE will increase, indicating inconsistency between accuracy and uncertainty.

3.3 Experimental setup

Monitoring setup and data description

This case study uses the data collected form five monopile-supported offshore wind turbines collected during a 2-year period (from late 2022 to 2024). The monitoring setup consists of a SCADA system which records turbine's environmental and operational conditions, nacelle accelerometers and tri-axial strain gauges on the transition piece specifically located at 14 m above the water level. The signals collected from SCADA and accelerometers are stored in the form of statistics at every 10-minute - mean, standard deviation, minimum, maximum, and root mean square (rms), etc.

In common practice, the temporal strain signals are often post-processed into bi-axial damage equivalent moments, DEM_{tl} in side-to-side (SS) direction and DEM_{tn} fore-aft (FA) directions, as follows:

$$DEM = \left(\frac{\sum_{i=1}^{N_m} n_i \cdot M_{r,i}^m}{N_{eq}} \right)^{1/m} . \tag{11}$$

The term damage equivalent load (or moment) is defined such that the damage caused by a predefined number of DEL range N_{eq} (usually specified as 10^7) is equal to that of the original load spectrum. N_m represents the number of bins in the load spectrum, $M_{r,i}$ and n_i are the reference moment value of the i^{th} bin and the corresponding number of cycles and 'm' is the slope of the linear SN curve. In this work, we train the model to predict the moment in SS direction DEM_{tl}.

Table 1 lists the input and output features of the virtual monitoring model. Although all SCADA and acceleration statistics are recorded, some statistical features are not used as model inputs depending on their influence on the output load (based on pre-conducted feature importance analysis), and data availability constraints. The collected data that spans 2 years is split into one year each for training and testing. The test data is further decomposed into two different sets: the samples that are similar to the training data (Test 1) and the others outside the distribution of the training data (Test 2). In this regard, the test samples are put in the second test dataset if one input feature is beyond 3 standard deviations in its normalized space of the training data. In particular, 35% of the 1-year test data is Test 1 and the remainder forms Test 2.

Neural network parameters

The probabilistic neural networks used in this investigation are implemented in PyTorch, a Python-supported tensor library for deep learning. Table 2 summarizes the network configurations and training parameters for all the methods. The same network architecture is used for all neural network models with three hidden layers with 32, 64, and 32 neurons respectively,

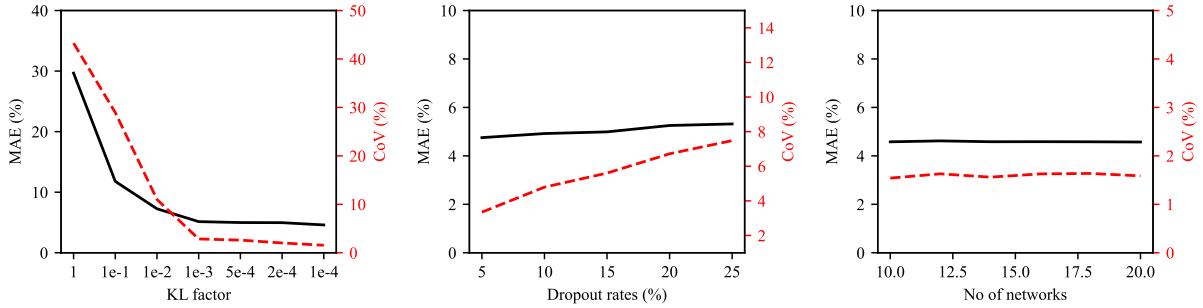
Table 1: Description of the input and output features.

	Sensor	Monitoring signal	Symbol	Units
Input	SCADA	Wind speed (mean)	$\mu[WSpd]$	m/s
		Wind speed (std)	$\sigma[WSpd]$	m/s
		Power (mean)	$\mu[Power]$	kW
		Yaw angle (mean)	$\mu[Yaw]$	deg
		Pitch angle (mean)	$\mu[Pitch]$	deg
		Pitch angle (std)	$\sigma[Pitch]$	deg
		Rotational speed (mean)	$\mu[RPM]$	rpm
		Rotational speed (std)	$\sigma[RPM]$	rpm
		Wind direction (mean)	$\mu[WDir]$	deg
		Turbulence intensity (mean)	$\mu[Turb]$	-
	Nacelle accelerometers	Z acceleration (mean)	$\mu[acc_Z]$	g
		Z acceleration (max)	$max[acc_Z]$	g
		Z acceleration (min)	$min[acc_Z]$	g
		Z acceleration (rms)	$rms[acc_Z]$	g
		FA acceleration (max)	$max[acc_{FA}]$	g
		FA acceleration (min)	$min[acc_{FA}]$	g
		FA acceleration (rms)	$rms[acc_{FA}]$	g
		SS acceleration (max)	$max[acc_{SS}]$	g
		SS acceleration (min)	$min[acc_{SS}]$	g
		SS acceleration (rms)	$rms[acc_{SS}]$	g
Output	Strain gauges	DEM (side-to-side)	DEM_{tl}	MNm

Table 2: Neural network architecture and training parameters.

	Bayes by backprop	MC-Dropout	Deep ensembles
No of neurons:			
Input layer	20	20	20
Hidden layers	32, 64, 32	32, 64, 32	32, 64, 32
Output layer	1	1	1
Optimizer	Adam	Adam	Adam
Learning rate	0.001	0.001	0.001
Batch size	2400	2400	2400
No of episodes	1000	1000	100
*KL loss factor	1-1e-4	-	-
*Dropout probability	-	0.1-0.25	-
*No of networks	-	-	10-20

* A sensitivity analysis is performed to analyze the hyper-parameters.



(a) Bayes by backprop (b) MC-Dropout (c) Deep ensembles

Figure 2: Sensitivity analysis. The influence of key hyperparameters is assessed for all tested probabilistic neural networks.

activated by rectified linear unit (*ReLU*) functions. The input layer receives 20 statistical features of SCADA and acceleration parameters and the output layer predicts DEM_{tl}. We do not implicitly model the aleatory uncertainty in this case study, and therefore the output layer has only one neuron. However, the BNN can also be alternatively laid out with additional neurons in the output layer to capture aleatory uncertainty [Hlaing et al., 2024].

During the course of the training task, the neural networks are trained on mini-batches of 2400 training samples, minimizing the respective formulated loss functions. The networks' weights are adjusted according to *Adam* optimizer at the learning rate of 0.001. Bayes by backprop and MC-Dropout are trained for 1000 episodes and the best weights with minimum loss are stored. In deep ensembles, each deterministic network is trained for only 100 epochs to avoid potential overfitting problems. Since each implemented inference method consists of different hyper-parameters to optimize, we also performed sensitivity analyses on those parameters. The hyper-parameters and tested values are also shown in Table 2.

The virtual sensing models trained by changing the values of hyperparameters are tested on the same test dataset to study their effects on the accuracy (MAE) and uncertainty estimation (CoV), computed as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N} \mid \hat{y}_i - y_i \mid, \tag{12}$$

$$CoV = \frac{1}{N}\sum_{i=1}^{N} \frac{\sqrt{\mathbb{V}(\hat{y}_i \mid \boldsymbol{x})}}{\mathbb{E}[\hat{y}_i \mid \boldsymbol{x}]}. \tag{13}$$

The test dataset used in the sensitivity analysis only consists of the samples that are inside the training region. In Bayes by backprop, the weight factor of the KL loss α is varied between 1 and 1e-4. With higher weights of the KL loss, the mean absolute error (MAE) and uncertainty tend to increase, as shown in Figure 2a. With high KL factor, the posterior weights remain closer to the initial standard Gaussian weights and the model's fitting capacity to the training data is limited, thus affecting the prediction accuracy. The MAE and model uncertainty converges for KL factors less than 1e-3.

The dropout rates in the MC-dropout do not have a significant effect on the averaged mean absolute error. However, the epistemic uncertainty changes by varying the dropout rates such that it increases for higher dropout rates. However, smaller uncertainties might not necessarily mean that the model predictions are better, as it can be seen in Figure 2b that the MAE re-

mains the same. With small values of dropout rates, the model weights just becomes almost deterministic, thus showing smaller CoVs.

Deep ensembles of multiple deterministic networks are tested - varying the number of models from 10 to 20. Neither MAE nor COV is dependent on the number of models, with consistent values of $\approx 4.5\%$ of accuracy and ≈ 1.5 for CoV in Figure 2c. Based on the sensitivity results, the models trained by Bayes by backprop with KL loss factor 1e-4, MC-Dropout with dropout probability 10%, and deep ensembles of 20 neural networks are selected for the benchmark study.

4 RESULTS AND DISCUSSION

4.1 Comparative study

The goal of this comparative study is to test popular BNN variants in terms of their ability to generalize and represent uncertainty. Table 3 lists the mean absolute error (MAE) and area under the sparsification error (AUSE) curve on the two test datasets. All models achieve an MAE of approximately 4.5-5% on the first dataset (i.e., Test 1), where the samples are closer to the training data distribution. Additionally, the models are tested on out-of-distribution data (i.e., Test 2) to assess their generalization capabilities. As expected, the MAE for all models increases significantly on Test 2 compared to Test 1. Interestingly, Bayes by backprop outperforms the other probabilistic neural network variants, resulting in an MAE of 10%, whereas MC-Dropout and deep ensembles result in MAEs of 24% and 29%, respectively.

To analyze the ability of the models to rank predictions in terms of model uncertainty, the sparsification curves are shown in Figure 3 for Test 1, Test 2, and the combined dataset (i.e., Test 1+2). The sparsification curves are mostly monotonic for Test 1, showing an overall correspondence between model error and uncertainty ranking. The sparsification curves do not, however, fully align with their corresponding oracle curves, and the MAE only slightly reduces, although the model becomes more confident. We attribute this result to the fact that prediction errors are mostly small in this case; the model will not give significantly higher model uncertainties, resulting in flat epistemic curves. This behavior can also be expected considering that test samples are close to the training dataset, and the model is naturally confident in the predictions despite possible deviations from ground-truth labels.

It is worth noting that the epistemic sparsification curve of MC-Dropout is not monotonic on Test 2, which can be caused by incorrect uncertainty estimates. On the other hand, despite high prediction errors, deep ensemble models' oracle and epistemic curves almost fully align with each other, indicating the model's strong uncertainty awareness. The epistemic curve of Bayes by backprop rapidly decreases as the sparsity increases, yet it does not completely align with the oracle.

Table 3: Performance of the tested probabilistic neural networks on two test datasets. Note that samples contained in *Test 2* dataset are farther to the training data distribution than samples included in *Test 1*.

	Test 1		Test 2	
	MAE (%)	AUSE	MAE (%)	AUSE
Bayes by backprop	4.59	0.41	10.19	0.33
MC-Dropout	4.92	0.47	24.2	0.24
Deep ensembles	4.58	0.37	29.36	0.11

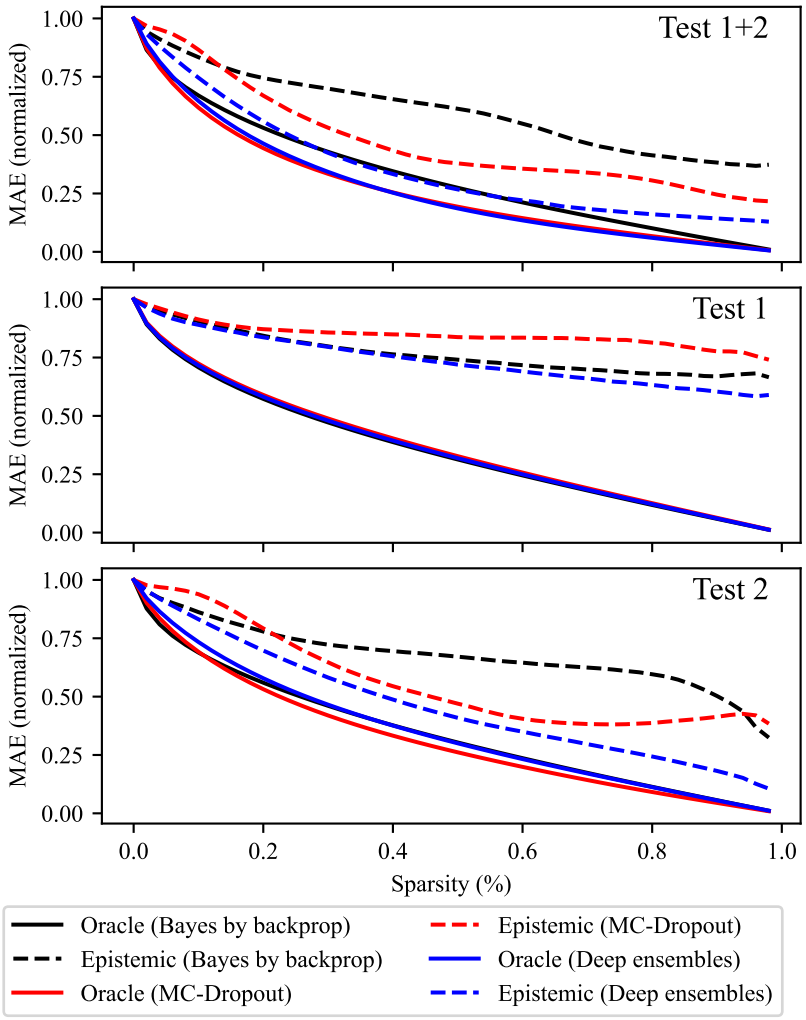


Figure 3: Sparsification curves for the investigated probabilistic models. Samples contained in *Test 2* dataset are farther to the training data distribution than samples included in *Test 1*.

By observing the sparsification curves of Test 1+2, it can be seen that the epistemic curves of Test 1+2 up to ∼70% sparsity also correspond to those of Test 2, and the remaining tails are similar to Test 1. It demonstrates that the models themselves can distinguish between similar data and shifted data. Especially, Bayes by backprop and deep ensembles are able to provide robust uncertainty estimates for out-of-distribution data. Areas under the sparsification error are also listed in Table 3. As one can expect, the AUSE values on Test 1 are similar among all the models. In Test 2, the minimum AUSE of deep ensembles represents the afore-mentioned accuracy-uncertainty correspondence.

To further assess the quality of uncertainty indicators, we computed the coverage probability — a frequentist measure of uncertainty representativeness— which checks if the ground-truth values lie within the credible intervals predicted by the models. The 95% intervals are computed and described as a ratio with respect to the predicted values. The CPs are computed for test samples of which the interval width to mean ratio is higher than the indicated threshold. The CPs of the tested approaches are plotted in Figure 4 for combined Test 1+2. In Bayes by

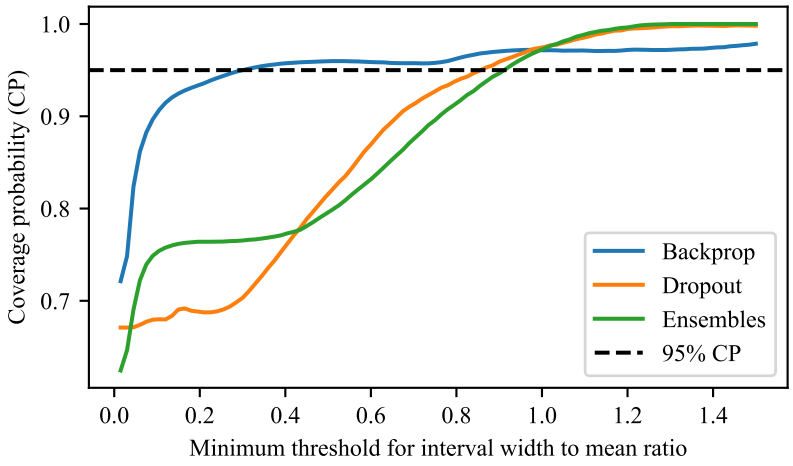


Figure 4: The 95% coverage probability (CP) of the investigated probabilistic neural networks is represented as a function of the specified threshold. CP is computed by selecting test samples for which the predicted interval width to mean ratio is higher than the indicated threshold.

backprop, 95% coverage probability is achieved when the interval width to mean ratio goes higher than 20%. For the other test samples, i.e., the interval width to mean ratio is less than 0.2, the BNN model is confident but the target value is out of the predicted interval. However, it does not necessarily mean that the prediction error, e.g., mean absolute error, will be high. The other two models, MC-Dropout and deep ensembles achieve the 95% CP only when the interval width is higher than 80% of the mean value, meaning that the model is, in general, over-confident for most of the test samples. If one would like to deploy these models in the decision-making problem, re-calibration of the uncertainty bounds should be considered for safe application.

4.2 Computational requirements

We summarize in Table 4 the computational requirements for training and deployment of the investigated approaches on an 8-core processor operating at a clock speed of 3.2 GHz and 8 GB memory. For Bayes by backprop and MC-Dropout models, the training time is mainly influenced by the model complexity, i.e., the number of neurons, and the number of training samples. The indicated computational requirements are for training the dataset consisting of 219,820 samples. The training time of deep ensembles further depends on the number of neural networks. In the above case study, we use 20 models and require around 2300 seconds of training time, making it the most computationally expensive.

The computational efforts for prediction are indicated for 237,305 test points. One needs to simulate multiple forward passes to compute the predicted value and the model uncertainty. Whereas the number of forward simulations required is case-specific, we run 100 passes of BNNs (Bayes by backprop and MC-Dropout) in this work. Bayes by backprop is more efficient, only requiring 2.7 seconds, while MC-Dropout takes 15.7 seconds. In deep ensembles, it is not necessary or impossible to perform a large number of forward passes as in Bayes by backprop and MC-Dropout, and the number of forward simulations is just the same as the number of neural networks, making it the most efficient during the prediction phase.

Table 4: Computational requirements for training and deployment of the probabilistic models.

	Training	Prediction
Bayes by backprop	∼1100 s	∼2.7 s (100 simulations)
MC-Dropout	∼1200 s	∼15.7 s (100 simulations)
Deep ensembles	∼2300 s	∼0.5 s (20 simulations)

5 CONCLUSIONS

This study assesses the effectiveness of probabilistic deep learning models in serving as virtual load sensors for offshore wind turbines. The results show that the tested probabilistic models can generally signal uncertain samples, which are most likely characterized by high prediction errors, resulting in monotonic uncertainty-error sparsification curves. Among the approaches investigated, deep ensembles lead to a better agreement between error and uncertainty ranking compared to the other tested Bayesian neural networks, which can yield over-confident uncertainty estimates when receiving out-of-distribution data. In terms of probability calibration, the results reveal that all tested models tend to be over-confident when generating predictions for samples close to the training data distribution, whereas more reliable uncertainty metrics can be obtained when the models are presented with out-of-distribution data. Overall, probabilistic models are able to accurately predict loads under conditions similar to the training data while signaling high uncertainty when making predictions from data collected under different conditions, thereby offering a useful tool for managing the risk of offshore wind assets.

While a set of probabilistic deep learning models is investigated in terms of generalization, uncertainty rankings, and probability calibration, this research can be extended by exploring inference methods, such as Markov Chain Monte Carlo, to quantify the discrepancy between the computed approximate and exact posterior distributions over neural network parameters.

ACKNOWLEDGMENTS

This research is funded by the Belgian Energy Transition Fund (FPS Economy) through the SmartLife project, and the European Union (GA No. 1011122184) through the WILLOW project. However, views and opinions expressed do not necessarily reflect those of the funders. None of the granting authorities can be held responsible for them.

DATA AVAILABILITY

Due to its proprietary nature and confidentiality concerns, supporting data of this research cannot be made publicly available.

REFERENCES

[Blundell et al., 2015] Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. (2015). Weight uncertainty in neural network. In *International conference on machine learning*, pages 1613–1622. PMLR.

[Cho et al., 2021] Cho, S., Choi, M., Gao, Z., and Moan, T. (2021). Fault detection and diagnosis of a blade pitch system in a floating wind turbine based on kalman filters and artificial neural networks. *Renewable Energy*, 169:1–13.

[Gal and Ghahramani, 2016] Gal, Y. and Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR.

[Graves, 2011] Graves, A. (2011). Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems*, volume 24, pages 2348–2356.

[Gustafsson et al., 2020] Gustafsson, F. K., Danelljan, M., and Schon, T. B. (2020). Evaluating scalable bayesian deep learning methods for robust computer vision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 318–319.

[Hlaing, 2024] Hlaing, N. (2024). *Data-driven virtual monitoring and life-cycle management of offshore wind support structures*. PhD thesis, Universite de Liege (Belgium).

[Hlaing et al., 2024] Hlaing, N., Morato, P. G., de Nolasco Santos, F., Weijtjens, W., Devriendt, C., and Rigo, P. (2024). Farm-wide virtual load monitoring for offshore wind structures via bayesian neural networks. *Structural Health Monitoring*, 23(3):1641–1663.

[Hlaing et al., 2022] Hlaing, N., Morato, P. G., and Rigo, P. (2022). Probabilistic virtual load monitoring of offshore wind substructures: A supervised learning approach. In *The Proceedings of the 32nd International Ocean and Polar Engineering Conference, ISOPE-2022*, volume 4, pages 3137–3144.

[Ilg et al., 2018] Ilg, E., Cicek, O., Galesso, S., Klein, A., Makansi, O., Hutter, F., and Brox, T. (2018). Uncertainty estimates and multi-hypotheses networks for optical flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 652–667.

[Mbuvha et al., 2021] Mbuvha, R., Mongwe, W. T., and Marwala, T. (2021). Separable shadow Hamiltonian hybrid Monte Carlo for Bayesian neural network inference in wind speed forecasting. *Energy and AI*, 6:100–108.

[Osawa et al., 2019] Osawa, K., Swaroop, S., Khan, M. E. E., Jain, A., Eschenhagen, R., Turner, R. E., and Yokota, R. (2019). Practical deep learning with Bayesian principles. In *Advances in Neural Information Processing Systems*, volume 32.

[Pimenta et al., 2022] Pimenta, F., Pacheco, J., Pereira, S., and Magalhães, F. (2022). Reconstructing the bending moments time history of wind turbine tower from acceleration measurements using Gaussian processes. *Journal of Physics: Conference Series*, 2265(3):032080.

[Singh et al., 2024] Singh, D., Dwight, R., and Viré, A. (2024). Probabilistic surrogate modeling of damage equivalent loads on onshore and offshore wind turbines using mixture density networks. *Wind Energy Science Discussions*, 2024:1–28.

[Wager et al., 2013] Wager, S., Wang, S., and Liang, P. S. (2013). Dropout training as adaptive regularization. In Burges, C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

[Welling and Teh, 2011] Welling, M. and Teh, Y. W. (2011). Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 681–688. Citeseer.

[Zhang et al., 2019] Zhang, R., Li, C., Zhang, J., Chen, C., and Wilson, A. G. (2019). Cyclical stochastic gradient mcmc for bayesian deep learning. *arXiv preprint arXiv:1902.03932*.

[Zou et al., 2023] Zou, J., Cicirello, A., Iliopoulos, A., and Lourens, E.-M. (2023). Gaussian process latent force models for virtual sensing in a monopile-based offshore wind turbine. In *EWSHM 2022: European Workshop on Structural Health Monitoring.*, pages 290–298. Springer International Publishing.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

BAYESIAN ESTIMATION OF SEISMIC FRAGILITY CURVES BASED ON VARIATIONAL REFERENCE PRIORS USING NEURAL NETWORKS

Nils Baillie[1,2], Antoine Van Biesbroeck[1,2], Cyril Feau[2] and Clément Gauchy[3]

[1]CMAP, CNRS, École polytechnique, Institut Polytechnique de Paris,
91120 Palaiseau, France
nils.baillie@polytechnique.edu

[2] Université Paris-Saclay, CEA, Service d'Études Mécaniques et Thermiques,
91191 Gif-sur-Yvette, France
{nils.baillie, antoine.vanbiesbroeck, cyril.feau}@cea.fr

[3] Université Paris-Saclay, CEA, Service de Génie Logiciel pour la Simulation,
91191 Gif-sur-Yvette, France
clement.gauchy@cea.fr

Abstract. *Seismic fragility curves are key quantities in seismic probabilistic risk assessment studies. They express the probability of failure of a mechanical structure of interest conditional to a scalar measure derived from the ground motion and called intensity measure (IM). The evaluation of such curves using Monte Carlo methods with artificial seismic signals is common today. However, when they are based on complex high-fidelity numerical modeling or on results from shaking table experiments, the amount of data available is limited. In this context, the Bayesian approach can provide an efficient learning of the parameters that define the fragility curve, often expressed by a probit-lognormal model. Nevertheless, the choice of the prior can have an important influence on the posterior and the parameter estimation. To limit the added subjectivity from a priori information, one can use the framework of reference priors. In this work, we propose a variational inference approach to approximate the prior. It takes the form of the output of a neural network, optimized to solve the maximization problem that defines the reference priors. Our algorithm is flexible: it can retrieve reference priors when constraints are specified in the optimization problem to ensure the solution is proper, and it proposes a simple method to recover a relevant approximation of the parametric posterior distribution using Markov Chain Monte Carlo methods even if the density function of the parametric prior is not known in general. We validate our methodology on a case study taken from the nuclear industry. Our results show the usefulness of this approach in recovering the target distribution.*

Keywords: Reference priors, Variational inference, Neural networks, Fragility curves

1 INTRODUCTION

Seismic fragility curves are a fundamental component of the seismic probabilistic risk assessment (SPRA) framework. Originally introduced in the 1980s for seismic risk assessment studies in the context of the nuclear industry [1–5], they have since become a key-tool in performance-based earthquake and engineering (PBEE) [5]. These curves express the probability of failure of a given mechanical equipment as a function of the seismic hazard. The latter being generally reduced to a single intensity measure (IM) that is derived from seismic ground motions, under the so-called "sufficiency assumption" [5, 6]. Common choices for IMs include peak ground acceleration (PGA), pseudo spectral acceleration (PSA), and peak ground velocity (PGV), among others.

Different data sources can be exploited to estimate seismic fragility curves (e.g. experimental data [4] or analytical results given by numerical models and artificial seismic excitation [7–21]). In this work we address problems within which the observed data are (i) scarce, (ii) independent, and (iii) limited to binary outcomes. They take the form of independent realizations of the tuple $X = (z, a)$ where $z \in \{0, 1\}$ represents the outcome —failure or non-failure— and $a \in \mathcal{A} \subset (0, \infty)$ represents the IM. In such cases, fragility curves are commonly parameterized using the probit-lognormal model [7–9, 11–18, 21–24], which introduces a parameter θ to describe the probabilistic relationship between z and a.

Among the various methods that exist to estimate the parameter θ, the Bayesian approach has gained increasingly popularity in the literature [10, 16, 18, 21, 24–33]. The Bayesian framework offers several advantages over classical frequentist methods, particularly in avoiding the generation of irregular fragility curves —such as unit step functions— that may arise from limited data. The challenge of this approach, however, lies in selecting an appropriate prior distribution as it can significantly influence the posterior distribution and, consequently, the resulting estimates.

A well-established tool for addressing the prior selection issue is the reference prior theory, first introduced in [34]. Reference priors are designed to maximize the influence of the observed data in Bayesian inference by maximizing the mutual information between the prior and the posterior. This property makes them particularly suitable for fragility curve estimation when data are sparse, as they minimize the influence of arbitrary subjective assumptions. A notable application of this approach in seismic fragility curve estimation is suggested in [31, 32] where a reference prior is specifically derived for this problem. Their implementation relies on a numerical approximation of the "complex" theoretical form of the prior, which can be cumbersome to evaluate.

In this work, we propose a novel methodology for estimating seismic fragility curves using a variational approximation of the reference prior (VA-RP). The VA-RP, as introduced in [35], parameterizes the reference prior as the output of a neural network, permitting efficient computation of *a priori* samples. The weights of the neural network are then trained to maximize the mutual information. In our study, the neural network architecture is specially tailored to the context of fragility curves estimations. Additionally, because a significant issue in the estimation of fragility curves with reference priors is the yielding of improper posteriors when the data are scarce, we propose the incorporation of constraints to the VA-RP. These problem-specific constraints ensure that both the prior and the posterior are proper. This additional feature enhances the robustness of Bayesian inference, making the method applicable even in cases with severely limited observational data, which is common in earthquake engineering.

After reviewing the reference prior theory in Section 2, we introduce the VA-RP methodol-

ogy in Section 3. The tailoring of the framework for fragility curve estimation is described in Section 4. Then, in Section 5, we evaluate our approach on a case study taken from the nuclear industry. Finally, concluding remarks are discussed in Section 6.

2 REFERENCE PRIORS THEORY

We consider the following situation: we observe i.i.d data samples $\mathbf{X} = (X_1, ..., X_N) \in \mathcal{X}^N$ with $\mathcal{X} \subset \mathbb{R}^d$. The likelihood function $L_N(\mathbf{X} \,|\, \theta) = \prod_{i=1}^N L(X_i \,|\, \theta)$ is known and $\theta \in \Theta \subset \mathbb{R}^q$ is the parameter to be inferred. For this purpose, θ is considered to be stochastic in the Bayesian framework, so that it admits a prior distribution that we denote by π. The marginal distribution is defined as the one whose density is $p_{\pi,N}(\mathbf{X}) = \int_\Theta \pi(\theta) L_N(\mathbf{X} \,|\, \theta) d\theta$. In those settings, the posterior distribution is defined as follows:

$$\pi(\theta \,|\, \mathbf{X}) = \frac{\pi(\theta) L_N(\mathbf{X} \,|\, \theta)}{p_{\pi,N}(\mathbf{X})}, \tag{1}$$

it is used to generate *a posteriori* estimates of θ.

The reference prior theory has been introduced in [34], and formalized further in [36]. The theory suggests to select the prior π as one that maximizes the quantity called mutual information and denoted by $I(\pi; L_N)$. It represents the amount of information that differs between the prior and the posterior. Thus, the definition of the reference prior amounts to solve the following optimization problem:

$$\pi^* \in \arg\max_{\pi \in \mathcal{P}} I(\pi; L_N) \,, \tag{2}$$

where $\mathcal{P}$ is a class of admissible probability distributions. The mutual information itself is defined by:

$$I(\pi; L_N) = \int_{\mathcal{X}^N} \mathrm{KL}(\pi(\cdot \,|\, \mathbf{X}) \,||\, \pi) p_{\pi,N}(\mathbf{X}) d\mathbf{X}, \tag{3}$$

where KL designates the Kullback-Leibler divergence. Therefore, π^* is a prior that maximizes the Kullback-Leibler divergence between itself and the corresponding posterior averaged by the marginal distribution of datasets. The Kullback-Leibler divergence between two probability measures of density p and q defined on a generic set Ω writes:

$$\mathrm{KL}(p \,||\, q) = \int_\Omega \log\left(\frac{p(\omega)}{q(\omega)}\right) p(\omega) d\omega. \tag{4}$$

Thus, π^* is the prior that maximizes the influence of the data on the posterior distribution, justifying its reference (or objective) properties. Using Fubini's theorem and Bayes' theorem, a more practical form for the mutual information can be obtained:

$$I(\pi; L_N) = \int_\Theta \mathrm{KL}(L_N(\cdot \,|\, \theta) || p_{\pi,N}) \pi(\theta) d\theta, \tag{5}$$

with $p_{\pi,N}$ denoting the distribution of the data $\mathbf{X}$. The above equation expresses $I(\pi; L_N)$ as the amount of information that differs between the random variable the random variable of the parameters $\theta \sim \pi$ and the random variable of the data $\mathbf{X} \sim p_{\pi,N}$. A more generalized definition of RPs has been proposed in [37]. The generalized mutual information with a f-divergence is defined by

$$I_{\mathrm{D}_f}(\pi; L_N) = \int_\Theta \mathrm{D}_f(p_{\pi,N} || L_N(\cdot \,|\, \theta)) \pi(\theta) d\theta, \tag{6}$$

with

$$\mathrm{D}_f(p\,||\,q) = \int_\Omega f\left(\frac{p(\omega)}{q(\omega)}\right) q(\omega) d\omega, \tag{7}$$

where f is convex and verifies $f(1) = 0$. The classical mutual information corresponds to the special case $f = -\log$, indeed, $\mathrm{D}_{-\log}(p\,||\,q) = \mathrm{KL}(q\,||\,p)$. When $N \to +\infty$, it has been shown in [37, 38] that the solution of this asymptotic problem is the Jeffreys prior when the mutual information is expressed as in Equation (3), or when it is defined using an δ-divergence, as in Equation (6) with $f = f_\delta$, where:

$$f_\delta(x) = \frac{x^\delta - \delta x - (1-\delta)}{\delta(\delta-1)}, \quad \delta \in (0,1). \tag{8}$$

The Jeffreys prior is defined as: $J(\theta) \propto \det(\mathcal{I}(\theta))^{1/2}$ with $\mathcal{I}$ being the Fisher information matrix.

The Jeffreys prior is a non-informative prior, known to be improper (i.e. it integrates to infinity) in many circumstances. Moreover, its decay rates can produce improper posteriors. Improper posteriors are a problem because they prevent the generation of *a posteriori* estimates by Markov Chain Monte Carlo (MCMC) methods. To handle this issue, we adapted the algorithm to the framework of [39] that suggests solving a constrained version of the initial optimization problem to ensure the solution is proper:

$$\tilde{\pi}^* \in \underset{\substack{\pi\in\mathcal{P} \\ \text{s.t.}\,\mathcal{C}(\pi)<\infty}}{\arg\max}\, I_{\mathrm{D}_{f_\delta}}(\pi; L_N), \tag{9}$$

where the constraint is $\mathcal{C}(\pi) = \int_\Theta a(\theta)\pi(\theta)d\theta$ with a positive function a. When a δ-divergence is used in the definition of the mutual information, and the two following integrals are finite:

$$\int_\Theta J(\theta)a(\theta)^{1/\delta} d\theta < \infty \quad \text{and} \quad \int_\Theta J(\theta)a(\theta)^{1+1/\delta} d\theta < \infty, \tag{10}$$

a result in [39] states that the proper constrained RP $\tilde{\pi}^*$ has the following asymptotic form:

$$\tilde{\pi}^*(\theta) \propto J(\theta)a(\theta)^{1/\delta}. \tag{11}$$

3 VARIATIONAL APPROXIMATION OF THE REFERENCE PRIOR (VA-RP)

3.1 Definition

Variational inference encompasses a range of techniques designed to approximate probability distributions by solving an optimization problem —such as maximizing the evidence lower bound (ELBO) [40]. Given that our objective involves finding the prior that maximizes the mutual information defined in Equation (5), variational inference naturally lends itself to the approximation of RPs.

We present in this section the approach developed in [35], which is inspired by [41, 42] regarding the variational approximation of the RP, referred as the VA-RP. We limit the family of priors to a parametric class $\{\pi_\lambda,\ \lambda \in \Lambda\}$, with $\Lambda \subset \mathbb{R}^L$, thereby reducing the problem to a finite-dimensional setting. Consequently, the optimization tasks in equations (2) and (9) translate into finding $\arg\max_{\lambda\in\Lambda} I_{\mathrm{D}_f}(\pi_\lambda; L_N)$. We define the priors π_λ implicitly through the transformation:

$$\theta \sim \pi_\lambda \iff \theta = g(\lambda, \varepsilon), \quad \text{where} \quad \varepsilon \sim p_\varepsilon. \tag{12}$$

Here, g is a parameterized measurable function —typically a neural network— where λ represents its weights and biases, it has to be differentiable with respect to λ. The variable ε serves as a latent variable, drawn from a simple, easy-to-sample distribution p_ε. In practice, we choose a centered multivariate Gaussian $\mathcal{N}(0, \mathbb{I}_{p\times p})$.

This implicit construction provides considerable flexibility in defining prior distributions. However, except in simpler cases, the density function of π_λ remains unknown and cannot be evaluated directly. Instead, we obtain samples $\theta \sim \pi_\lambda$.

The method in [35] extends the ideas in [41, 42] to higher-dimensional models while incorporating a broader definition of mutual information based on f-divergences, as given in Equation (5).

3.2 Objective function and gradient

The numerical resolution of the optimization problem is based on stochastic gradient ascent with an objective function corresponding to a lower bound of the mutual information. This function was first introduced in [42] and a generalization to f-divergences with f decreasing is given in [35]:

$$B_{\mathrm{D}_f}(\pi; L_N) = \int_\Theta \int_{\mathcal{X}^N} f\left(\frac{L_N(\mathbf{X}\,|\,\hat{\theta}_{MLE})}{L_N(\mathbf{X}\,|\,\theta)}\right) \pi(\theta) L_N(\mathbf{X}\,|\,\theta) d\mathbf{X} d\theta, \tag{13}$$

with $\hat{\theta}_{MLE}$ being the maximum likelihood estimator (MLE). Since the function f_δ used in δ-divergence (Equation (8)) is not decreasing, we replace it with $\hat{f}_\delta$ defined hereafter, because $\mathrm{D}_{f_\delta} = \mathrm{D}_{\hat{f}_\delta}$:

$$\hat{f}_\delta(x) = \frac{x^\delta - 1}{\delta(\delta-1)} = f_\delta(x) + \frac{1}{\delta - 1}(x-1). \tag{14}$$

The gradient of the lower bound can be written as:

$$\frac{\partial B_{\mathrm{D}_f}}{\partial \lambda_l}(\pi_\lambda; L_N) = \mathbb{E}_\varepsilon\left[\sum_{j=1}^{q} \frac{\partial \tilde{B}}{\partial \theta_j}(g(\lambda,\varepsilon)) \frac{\partial g_j}{\partial \lambda_l}(\lambda,\varepsilon)\right], \tag{15}$$

where:

$$\frac{\partial \tilde{B}}{\partial \theta_j}(\theta) = \mathbb{E}_{\mathbf{X}\sim L_N(\cdot\,|\,\theta)}\left[\frac{\partial \log L_N}{\partial \theta_j}(\mathbf{X}\,|\,\theta) F\left(\frac{L_N(\mathbf{X}\,|\,\hat{\theta}_{MLE})}{L_N(\mathbf{X}\,|\,\theta)}\right)\right], \tag{16}$$

with $F(x) = f(x) – x f'(x)$. The previous gradient can be approximated via Monte Carlo using samples $\varepsilon \sim p_\varepsilon$ and $\mathbf{X} \sim L_N(\cdot|\theta)$. For the MLE, we also use samples of ε when no formula is available:

$$L_N(\mathbf{X}\,|\,\hat{\theta}_{MLE}) \approx \max_{t\in\{1,\dots,T\}} L_N(\mathbf{X}\,|\,g(\lambda,\varepsilon_t)) \quad \text{where} \quad \varepsilon_1, \dots, \varepsilon_T \sim p_\varepsilon. \tag{17}$$

The implementation in [35] relies on the PyTorch package [43] and the Adam optimizer [44], where the differentiation operations and the sampling operations are separated when computing the gradient with automatic differentiation.

3.3 Constraints incorporation

The authors in [35] suggest a method to incorporate a constraint in the optimization process. The constraint on the prior must take the form of $\mathbb{E}_{\theta\sim\pi}[a(\theta)] = b$, with $a : \Theta \longrightarrow (0, \infty)$ being a positive function, and $b \in (0, \infty)$ being a positive scalar. The optimization problem becomes the following:

$$\underset{\substack{\lambda\in\Lambda,\\ \text{subject to } \mathcal{C}(\pi_\lambda)=0}}{\arg\max} \quad \mathcal{B}_{\mathrm{D}_{\hat{f}_\delta}}(\pi_\lambda; L_N) \quad \text{with} \quad \mathcal{C}(\pi_\lambda) := \mathbb{E}_{\theta\sim\pi_\lambda}\left[a(\theta)\right] - b. \tag{18}$$

The same algorithm can be used with the augmented Lagrangian as the objective function instead:

$$\mathcal{L}_A(\lambda, \eta, \tilde{\eta}) = \mathcal{B}_{\mathrm{D}_{\hat{f}_\delta}}(\pi_\lambda; L_N) + \eta\mathcal{C}(\pi_\lambda) - \frac{\tilde{\eta}}{2}\mathcal{C}(\pi_\lambda)^2. \tag{19}$$

The right value for b is obtained through the estimation of the two integrals in Equation (10), which requires an approximation of the Jeffreys prior, the details are given in [35].

3.4 Posterior sampling

In order to build *a posteriori* estimates, we propose a sampling method for θ from its posterior distribution $\pi_\lambda(\theta \,|\, \mathbf{X})$. The prior distribution π_λ has an implicit form in general and cannot be evaluated. The samples are obtained from the posterior distribution of the variable ε which has a known density function up to a multiplicative constant:

$$\pi^{\text{post}}_{\varepsilon,\lambda}(\varepsilon \,|\, \mathbf{X}) \propto p_\varepsilon(\varepsilon) L_N(\mathbf{X} \,|\, g(\lambda, \varepsilon)) \; ; \quad \text{if } \varepsilon \sim \pi^{\text{post}}_{\varepsilon,\lambda}(\varepsilon \,|\, \mathbf{X}) \text{ then } g(\lambda, \varepsilon) \sim \pi_\lambda(\theta \,|\, \mathbf{X}). \tag{20}$$

We utilize in practice an adaptive version of the Metropolis-Hastings algorithm with a multivariate Gaussian as the proposition distribution.

4 VA-RP FOR SEISMIC FRAGILITY CURVES ESTIMATION

4.1 Problem statement

A seismic fragility curve is the probability of failure, denoted by $P_f(a)$, of a mechanical structure subjected to a seism as a function of an IM, which is a scalar value denoted by a and derived from the seismic ground motion. We denote by Z the variable that equals 1 if the mechanical structure fails, and 0 otherwise. We introduce the probit-lognormal model used to estimate seismic fragility curves, it is also referred as the lognormal model in the literature. The properties of the Jeffreys prior for this model are highlighted by [31].

The model is defined by the observation of a vector of i.i.d. samples $\mathbf{X} = (X_1, \ldots, X_N)$ where for any index i, $X_i \sim (Z, a) \in \mathcal{X} = \{0, 1\} \times (0, \infty)$. The random variable (Z, a) has the following distribution that depends on the parameters $\theta = (\alpha, \beta) \in \Theta = (0, \infty)^2$:

$$\begin{cases} a \sim p_a \\ P_f(a) = \Phi\left(\dfrac{\log a - \log \alpha}{\beta}\right) \\ Z \sim \text{Bernoulli}(P_f(a)), \end{cases} \tag{21}$$

where Φ designates the cumulative distribution function of the standard Gaussian. The distribution p_a is supposed to be known, it is common to associate it to a log-normal distribution:

$\log(a) \sim \mathcal{N}(\mu_a, \sigma_a^2)$ (see [31–33] and Section 5). The likelihood writes:

$$L_N(\mathbf{X} \,|\, \theta) = \prod_{i=1}^{N} p_a(a_i) \prod_{i=1}^{N} P_f(a_i)^{Z_i}(1 - P_f(a_i))^{1-Z_i} \propto \prod_{i=1}^{N} P_f(a_i)^{Z_i}(1 - P_f(a_i))^{1-Z_i} \tag{22}$$

For simplicity, we denote : $\gamma_i = \dfrac{\log a_i - \log \alpha}{\beta} = \Phi^{-1}(P_f(a_i)) = \text{probit}(P_f(a_i))$, the gradient of the log-likelihood has the following expression:

$$\begin{cases} \dfrac{\partial \log L_N}{\partial \alpha}(\mathbf{X} \,|\, \theta) = \displaystyle\sum_{i=1}^{N} \frac{1}{\alpha\beta} \left((-Z_i) \frac{\Phi'(\gamma_i)}{\Phi(\gamma_i)} + (1 - Z_i) \frac{\Phi'(\gamma_i)}{1 - \Phi(\gamma_i)} \right) \\ \dfrac{\partial \log L_N}{\partial \beta}(\mathbf{X} \,|\, \theta) = \displaystyle\sum_{i=1}^{N} \frac{\gamma_i}{\beta} \left((-Z_i) \frac{\Phi'(\gamma_i)}{\Phi(\gamma_i)} + (1 - Z_i) \frac{\Phi'(\gamma_i)}{1 - \Phi(\gamma_i)} \right). \end{cases} \tag{23}$$

We do not dispose of an explicit expression of the MLE. Consequently, we approximate it using samples in the variational inference process.

In [31], the authors study the asymptotical behaviors of the likelihood and of the Jeffreys prior. They prove that the Jeffreys prior is improper w.r.t. β, namely, when $\alpha > 0$ is fixed,

$$\begin{cases} J(\theta) \propto 1/\beta \quad \text{as} \quad \beta \longrightarrow 0 \\ J(\theta) \propto 1/\beta^3 \quad \text{as} \quad \beta \longrightarrow +\infty. \end{cases} \tag{24}$$

If $\beta > 0$ is fixed, the prior is proper on the first component α:

$$J(\theta) \propto \frac{|\log \alpha|}{\alpha} \exp\left(-\frac{(\log \alpha - \mu_a)^2}{2\beta^2 + 2\sigma_a^2} \right) \quad \text{when} \quad |\log \alpha| \longrightarrow +\infty. \tag{25}$$

It is proven in [31] that the improper rates of the Jeffreys prior w.r.t. β can yield improper posteriors. Based on this work and taking into account additional constraints to make it proper (see Section 2), the authors in [33] suggest, in order to have a more in-depth look at the Jeffreys prior, to approximate it from its decay rates. The resulting prior π_a is then given by:

$$\pi_a(\theta) = \frac{K}{\beta + \beta^3} \frac{|\log \alpha|}{\alpha} \exp\left(-\frac{(\log \alpha - \mu_a)^2}{2\beta^2 + 2\sigma_a^2} \right) \tag{26}$$

K represents a positive normalization constant. Furthermore, it is proven in Appendix B that, $\mathbb{E}_{\theta \sim \pi_a} |\log \alpha|^2 = \infty$. This result will be very useful in the sequel to interpret the results of the VA-RP training.

4.2 Neural network architecture

A single layer neural network is implemented for this problem. It takes the following form

$$\varepsilon \mapsto \begin{pmatrix} \mathbf{h}_1(\varepsilon) \\ \mathbf{h}_2(\varepsilon) \end{pmatrix} \mapsto \begin{pmatrix} \tilde{\varepsilon}_1 = \mathbf{w}_1^T \mathbf{h}_1(\varepsilon) + b_1 \\ \tilde{\varepsilon}_2 = \mathbf{w}_2^T \mathbf{h}_2(\varepsilon) + b_2 \end{pmatrix} \mapsto \begin{pmatrix} \alpha = u_1(\tilde{\varepsilon}_1) \\ \beta = u_2(\tilde{\varepsilon}_2) \end{pmatrix}. \tag{27}$$

The coordinates of the vectors $\mathbf{w}_1, \mathbf{w}_2 \in \mathbb{R}^p$ and $\mathbf{b} = (b_1, b_2) \in \mathbb{R}^2$ constitute the weights of the neural network, they are initialized independently and randomly w.r.t. a Gaussian distribution.

$\mathbf{h}_1$, $\mathbf{h}_2$, u_1, u_2 are activation functions. They are specifically chosen in order to ensure the VA-RP's decay rates match with the ones of the target prior: $u_1 = u_2 = \exp$ and for $\mathbf{x} \in \mathbb{R}^p$, $\mathbf{h}_1(\mathbf{x}) = (h_1(x_i))_i$ and $\mathbf{h}_2(\mathbf{x}) = (h_2(x_i))_i$ with

$$h_1(x) = x; \quad h_2(x) = \log(1 - \Phi(x)), \tag{28}$$

where Φ designates the c.d.f. of a standard Gaussian. In Appendix A, we prove that under those settings the VA-RP yields marginal distributions p_α, p_β w.r.t. α, β, that take the following form:

$$\begin{aligned} p_\alpha(\alpha) &= \frac{1}{\alpha\sqrt{2\pi}\|\mathbf{w}_1\|_2} \exp\left(-\frac{(\log\alpha - b_1)^2}{2\|\mathbf{w}_1\|_2^2}\right) \\ p_\beta(\beta) &= \sum_{i,\, w_{2i}>0} K_i \beta^{-\frac{1}{w_{2i}}-1} \mathbb{1}_{\beta > e^{b_2}} + \sum_{i,\, w_{2i}<0} K_i \beta^{-\frac{1}{w_{2i}}-1} \mathbb{1}_{\beta < e^{b_2}}, \end{aligned} \tag{29}$$

where $(w_{2i})_{i=1}^p$ denote the coordinates of $\mathbf{w}_2$, and the K_i are constants that depend on $\mathbf{w}_2$ and b_2. The marginal distribution p_α is a log-normal distribution, which is similar, up to the $\log\alpha$ term, to the Jeffreys prior decay rates w.r.t. α (Equation (25)). Concerning p_β, calling $w_{2j}^{-1} = \min\{w_{2i}^{-1},\, w_{2i} > 0\}$ and $w_{2l}^{-1} = \max\{w_{2i}^{-1},\, w_{2i} < 0\}$, it admits the following decay rates:

$$p_\beta(\beta) \underset{\beta\to 0}{\sim} K_j \beta^{-\frac{1}{w_{2l}}-1}; \quad p_\beta(\beta) \underset{\beta\to\infty}{\sim} K_l \beta^{-\frac{1}{w_{2j}}-1}. \tag{30}$$

They are consistent with the ones of the Jeffreys prior.

4.3 Numerical results

Training of the unconstrained VA-RP In this section, we present the training process of the VA-RP. We have implemented the probit-lognormal modeling with $\mu_a = 0$ and $\sigma_a = 1$. These parameters, which define the distribution of a, are the only elements that have a theoretical impact on the reference prior. Their values are derived from the case study which is considered in Section 5.

As for the different parameters for the training, we take $N = 50$, $T = 50$, $p = 50$. The training consists in the maximization of $\mathcal{B}_{\mathrm{D}_{\hat{f}_\delta}}$ with $\delta = 0.5$. The optimization has been done through a number of 9000 epochs, with a learning rate of 0.005.

Figure 1 shows the evolutions of the weights of the neural network that define the marginal distributions of p_α and p_β (see Equation (29)), as a function of the number of epochs. Thus, the bias b_1 is constant and equal to 0, which is consistent with the asymptotic distribution of α. $1/w_{2l}$ and $1/w_{2j}$ both tend to 0. As a result, the marginal distribution of β tends to $1/\beta$. This is an improper distribution which does not exactly match the expected improper Jeffreys prior (see Equation (24)): it has a heavier tail as $\beta \longrightarrow \infty$ than the original Jeffreys. However, we will see in the end of this section that the distribution is not limited to its tails, and that this prior actually ascribes huge weights to small values of β (see the paragraph "Posterior evaluation"). Finally, $\|\mathbf{w}_1\|_2$ tends to $+\infty$. This means that the variance of $\log\alpha$ tends to $+\infty$ as it is expected (see Appendix B), even though this does not lead to the target distribution. This last result is nevertheless not surprising. It is the consequence of the chosen architecture of the VA-RP: p_α is a lognormal distribution (Equation (30)), which is not exactly the case for the Jeffreys prior (Equation (25)). What fundamentally distinguishes these two distributions is that (i) in the first case, that of Jeffreys, the variance of $\log\alpha$ is finite conditionally to β, and is infinite otherwise because of the heavy tail of the distribution w.r.t. β while (ii) in the second case, the variance

becomes infinite to correspond, via the training process, to the target variance of the Jeffreys prior. In doing so, the marginal distribution of α tends towards the improper distribution $1/\alpha$. Although this result is open to criticism, it appears to be an "acceptable" compromise insofar as the model's likelihood tends to substantially attenuate the prior's tails with respect to α, thereby limiting the influence of the prior's decay rates on the posterior. An alternative approach could involve modifying the neural network architecture to better capture the correlation between α and β. However, such adjustments would inevitably increase the computational complexity of the training process.

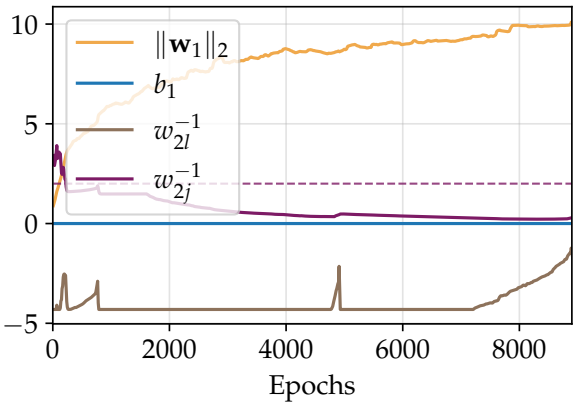


Figure 1: Weights of interest of the neural network during the unconstrained optimization. The brown and violet curves represent respectively $1/w_{2l}$ and $1/w_{2j}$ that are defined in Section 4.2. In violet dashed line is plotted the target value of $1/w_{2j}$.

Training of the constrained VA-RP As for the constrained case, we use a moment constraint of the form $a(\alpha, \beta) = \beta^\kappa$ for the second component with $\kappa \in \left(0, \frac{2}{1+1/\delta}\right)$, to ensure the proper character of the resulting constrained prior. We define $\gamma = \kappa/\delta \in (0, \frac{2}{1+\delta}) \subset (0, 2)$, the target proper prior is : $\pi^*(\theta) \propto J(\theta)\beta^\gamma$. Its asymptotic rates are identical to those of the Jeffreys prior for α. If we fix α, we have then:

$$\begin{cases} \pi^*(\theta) \propto \beta^{\gamma-1} & \text{as} \quad \beta \longrightarrow 0 \\ \pi^*(\theta) \propto \beta^{\gamma-3} & \text{as} \quad \beta \longrightarrow +\infty. \end{cases} \tag{31}$$

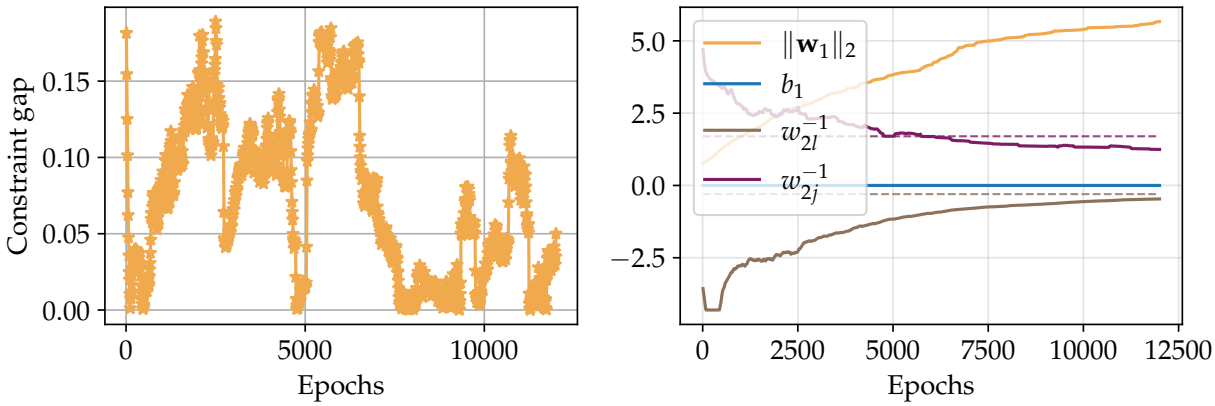
In this case, the approximation of the target prior, as suggested in [33] and expressed in Equation (26), becomes:

$$\pi_a^\gamma(\theta) = \frac{K_\gamma}{\beta^{1-\gamma} + \beta^{3-\gamma}} \frac{|\log \alpha|}{\alpha} \exp\left(-\frac{(\log \alpha - \mu)^2}{2\beta^2 + 2\sigma^2}\right), \tag{32}$$

with K_γ depending only on γ. In Appendix B, it is proven that for any $\gamma > 0$ this prior verifies $\mathbb{E}_{\theta\sim\pi_a^\gamma} |\log \alpha|^2 = \infty$.

The divergence is still set with $\delta = 0.5$, and we fix $\kappa = 0.15$, so that $\gamma = 0.3$. The training process of the constrained VA-RP is done during 12000 epochs with a learning rate of 0.001. We take $N = 50$, $T = 50$ and $p = 50$. the parameter η in the augmented Lagrangian is updated every 10 epochs.

Figure 2a suggests that the constraint seems reasonably verified during the whole optimization as the constraint gap is close to zero and appears to be decreasing. Figure 2b shows that the inverse weights $1/w_{2l}$ and $1/w_{2j}$ appear to steadily approach their theoretical values, respectively $-\gamma = -0.3$ and $2 - \gamma = 1.7$. As in the unconstrained case, we observe that $\|\mathbf{w}_1\|_2$ tends to $+\infty$. This is consistent with the theoretical results of the Appendix B which show, recall, that for all $\gamma \geq 0$, $\mathbb{E}[|\log \alpha|^2] = \infty$. So, we verify that adding constraints therefore allows, as expected, to "control" the distribution of β. Concerning α, we find the same trend as in the unconstrained case, which is theoretically expected from the point of view of the variance of the marginal distribution.



(a) Constraint value gap.

(b) Weights of interest of the neural network.

Figure 2: (a) Evolution of the constraint value gap during training. It corresponds to the difference between the target and current values for the constraint (in absolute value). (b) Weights of interest of the neural network during the constrained optimization. The yellow and blue curves represent respectively $\|\mathbf{w}_1\|_2$ and b_1, where $\|\mathbf{w}_1\|_2^2$ is the variance of the log of the first component, and b_1 its mean. The brown and violet curves represent respectively $1/w_{2l}$ and $1/w_{2j}$ that are defined in Section 4.2. In same color dashed line are plotted their target value.

Posterior evaluation We suggest to evaluate the posterior that the VA-RPs derived in this section yield. For this purpose, we take as dataset 50 samples that are generated given a probit-lognormal model with θ_{true} close to $(3.37, 0.43)$. The distribution of a considered is still a log-normal with $\mu_a = 0$ and $\sigma_a = 1$.

A posteriori samples are generated using the adaptive Metropolis-Hasting on ε as described in Section 3.4. A number of 5000 samples of θ are generated this way. For the sake of completeness, they are compared with samples generated considering an other prior: the actual reference prior derived by approximating numerically its theoretical expression, as depicted in [31]. The samples are generated by an other adaptive Metropolis-Hastings algorithm, which iteratively evaluates the prior. In the following, we refer to the latter as AJ.

In this example, we have not retained the samples for which the data are dissociated into two disjoint subsets when they are classified according to the values of a. In this case, indeed, the posterior distribution is improper because the likelihood is constant when β tends to 0 [31]. We then say that the dataset is degenerate [33]. The probability of occurrence of such datasets is not negligible if the dataset has a small size. The following plots are obtained with non-degenerate

datasets.

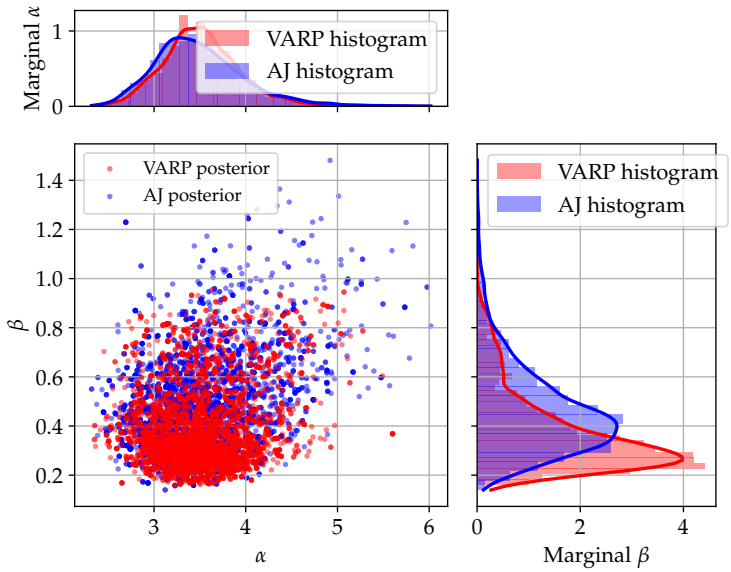


Figure 3: Scatter histogram of the unconstrained VA-RP posterior and the approximated Jeffreys (AJ) posterior distributions obtained from 5000 samples. Kernel density estimation is used on the marginal distributions in order to approximate their density functions with Gaussian kernels.

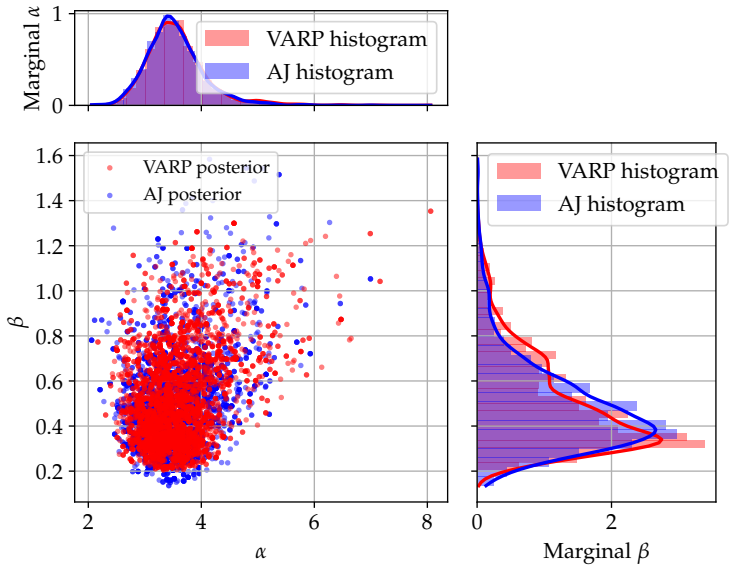


Figure 4: Scatter histogram of the constrained VA-RP posterior and the approximated Jeffreys (AJ) posterior distributions obtained from 5000 samples. Kernel density estimation is used on the marginal distributions in order to approximate their density functions with Gaussian kernels.

Figures 3 and 4 show that the posterior distribution obtained from the VA-RP is more like Jeffreys prior in the constrained case than in the unconstrained one. More specifically, w.r.t. α, the VA-RP is similar to the Jeffreys prior in the both cases. That suggests that the limitation of

our neural network architecture discussed earlier has little impact on the posterior distribution. Regarding β, in the unconstrained case, the VA-RP ascribes predominant weight to smaller values of β compared to the Jeffreys prior, with a mode that is closer to 0. In terms of fragility curve estimation that could represent an issue, as it may lead to estimates with thinner credibility intervals around a fragility curve.

5 APPLICATION OF THE METHOD ON A CASE STUDY TAKEN FROM THE NUCLEAR INDUSTRY

5.1 Artificial seismic signals

In this study, we rely on a large database of 10^5 artificial seismic signals that were generated beforehand. The generation of them is based on the stochastic generator suggested in [45]. It has been calibrated as in [46] from 97 real accelerograms taken from the European strong motion database for a magnitude that lives between 5.5 and 6.5, and a source-to-site distance that is lower than 20 km [47].

The IM that we have derived for this study is the peak ground acceleration (PGA). In Figure 5, we compare the empirical distribution of the PGA given from that database with an approximation of the one that comes from the 97 real seismic signals. In the same figure is also plotted the p.d.f. of a log-normal distribution with same median and same log deviation than the empirical distribution of the IM. This distribution corresponds to a log-normal with parameters (μ_a, σ_a) that are approximately equal to $(0, 1)$.

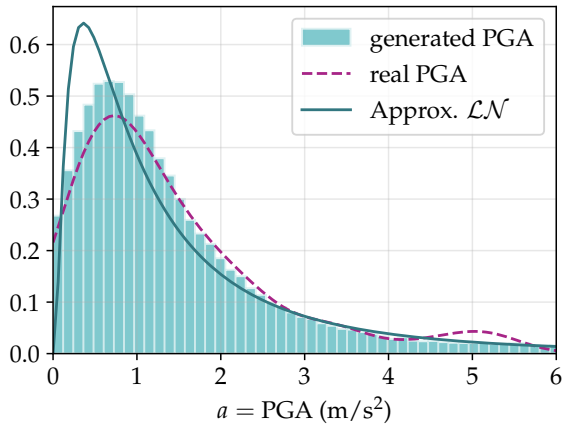


Figure 5: Histogram of the PGA of 10^5 generated signals compared with: 1. an approximation of the distribution of 97 PGA derived from real seismic signals using kernel density estimation (dashed line), 2. the p.d.f. of a log-normal distribution with same median and same log deviation than the generated signals (solid line).

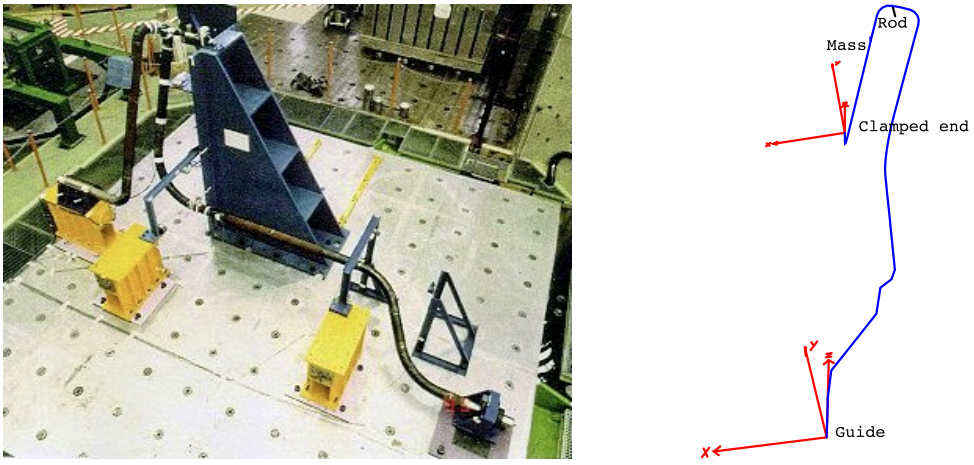
5.2 Description of the case study

This case study examines a piping system that constitutes a secondary line within a French pressurized water reactor. The system was investigated both experimentally and numerically [48]. Figure 6a presents an overview of the experimental mock-up, which was mounted on the Azalee shaking table at the EMSI laboratory of CEA. The corresponding finite element model (FEM), constructed using beam elements, is depicted in Figure 6b. This numerical model was

implemented using the in-house finite element code CAST3M [49] and validated through an extensive experimental campaign.

The mock-up is fixed at one end, while the opposite end is supported by a guide to restrict displacement in the X and Y directions. Additionally, a rod is positioned atop the specimen to constrain mass displacement in the Z direction (see Figure 6b). During the experimental tests, excitation was applied exclusively in the X direction. The artificial input signals were filtered using a fictitious 2% damped linear single-mode system at 5 Hz, which corresponds to the first eigenfrequency of the piping system with a 1% damping ratio.

The engineering demand parameter (EDP) adopted in this study is the out-of-plane rotation of the elbow located near the clamped end of the mock-up, as recommended in [50]. The failure is defined as an excessive value of that EDP. The critical rotation threshold is set at $3.8°$, representing the 90% quantile of a sample consisting of $8 \cdot 10^4$ numerical simulations (see Figure 7b).



(a) (b)

Figure 6: (a) Overiew of the mock-up of the piping system on the Azalee shaking table in CEA and (b) associated FEM.

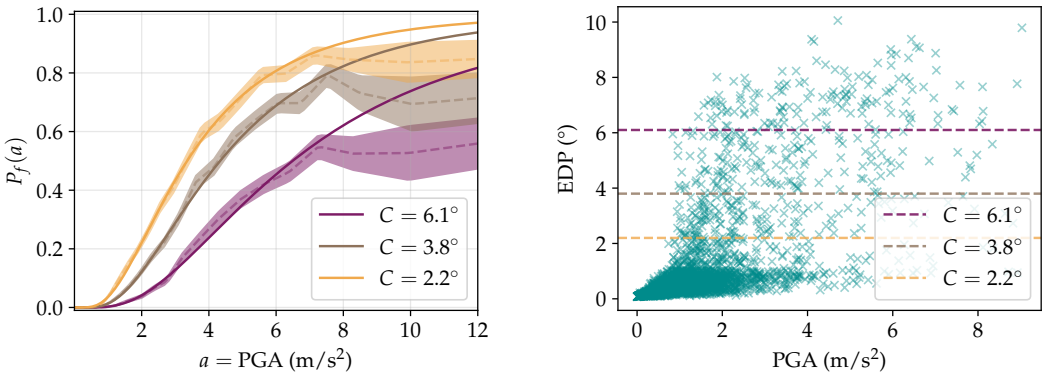
5.3 Reference fragility curve and benchmarking metrics

To evaluate our estimates, we compare them to a reference fragility curve, derived for this case study using the full batch of the $8 \cdot 10^4$ data. That reference fragility curve, denoted by P_f^{ref} consists in a non-parametric estimate based on Monte-Carlo estimations of local probability of failures w.r.t. the IM, as suggested in [12].

Different fragility curves, derived considering different excessive rotation threshold for defining the equipment's failure are elucidated in Figure 7a. Each is compared to a probit-lognormal curve P_f^{MLE} whose parameters are estimated by maximum likelihood estimation using the $8 \cdot 10^4$ results presented in Figure 7b. That comparison demonstrates the adequacy of such modeling of the fragility curve. However, it also emphasizes an existing bias that the model has compared with the non-parametric estimate. In the following, we refer to this difference between P_f^{MLE} and P_f^{ref} as the square model bias, denoted by $\mathcal{MB}$:

$$\mathcal{MB} = \|P_f^{\text{ref}} - P_f^{\text{MLE}}\|_{L^2}^2; \tag{33}$$

where the norm $\|\cdot\|_{L^2}$ is defined by $\|P\|_{L^2}^2 = \int_0^\infty P(a)^2 da$. It is derived numerically from Simpson's interpolation on a sub-division of $(0, \infty)$.



(a) (b)

Figure 7: (a) Non-parametric reference fragility curves (dashed lines) surrounded by their 95% confidence intervals for different critical rotation threshold C. The thresholds yield different proportion of failures in the database, respectively 95% (purple), 90% (brown), and 85% (yellow). For each value of C are plotted (same color, solid line) the corresponding probit-lognormal MLE. (b) Results of the $8 \cdot 10^4$ numerical simulations : each cross is an element of the database (IM, EDP).

Given an observed sample of size N: $\mathbf{X} = (X_1, \ldots, X_N)$, we denote by $P_f^N : a \mapsto \Phi(\beta^{-1} \log(a/\alpha))$ whose distribution inherits from the posterior distribution $\pi(\cdot \,|\, \mathbf{X})$ of θ. For any value of a, we note $q_r^N(a)$ the r-quantile of $P_f^N(a)$, and $m^N(a)$ its median. We propose to evaluate our estimates using the following metrics:

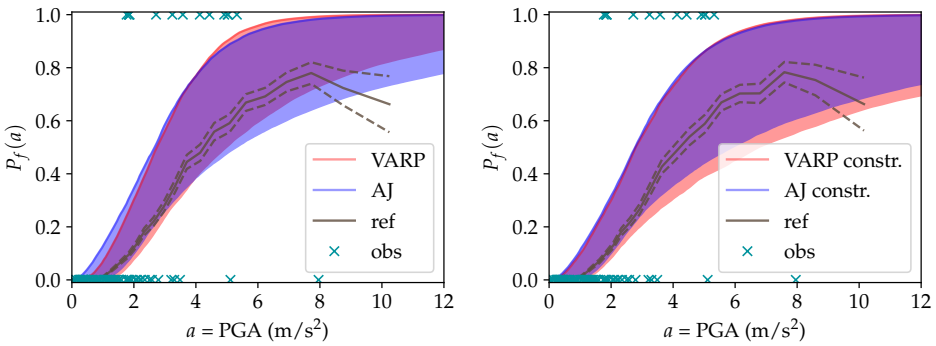
- The square bias to the median $\mathcal{B}^N = \|m^N - P_f^{\text{ref}}\|_{L^2}^2$.
- The credibility width $\mathcal{W}^N = \|q_{1-r/2}^N - q_{r/2}^N\|_{L^2}^2$, with $r = 0.05$.
- The upper quantile error $\mathcal{Q}^N = \|q_{1-r/2}^N - P_f^{\text{ref}}\|_{L^2}^2$.

The quantiles and the medians are approximated using generated samples.

5.4 Results and discussion

In this section, we compare the evaluation of our estimates that results from using the VA-RP with the evaluation of the ones that result from using an approximation via numerical derivation of the Jeffreys prior. In the following, we refer to the latter as AJ. Both are implemented with and without the incorporation of a constraint that is tailored to ensure they are proper. The considered constraint is detailed in Section 4.3.

In Figure 8 we provide examples of *a posteriori* fragility curves credibility intervals yielded by the two priors, with and without incorporating a constraint on the decay rates. These examples arise from given samples of $N = 80$ observations. They illustrate that the VA-RP is capable of providing fragility curve estimates that are equivalent of the ones provided by the AJ, which is an approximation of the exact expression of the reference prior.



(a) (b)

Figure 8: Example of *a posteriori* 95% credibility intervals yielded by the posterior distribution resulting from a sample of $N = 80$ data and from the VA-RP (in red) or the AJ (in blue) : (a) unconstrained priors and (b) constrained priors. The reference curve P_f^{ref} (solide line) is surrounded by its 95% confidence interval (dashed line) in each figure. The cyan crosses represent the observed sample.

To deepen the comparison, we implemented the methods multiple times for values of N varying in $[10, 250]$. More precisely, for each $N \in \{10, 20, \ldots, 250\}$, 10 random data samples of size N have been drawn. Each of these samples were then used to compute the metrics $\mathcal{B}^N$, $\mathcal{W}^N$ and $\mathcal{Q}^N$.

The average values of these metrics along with their 95% confidence intervals are presented in Figure 9 (for the unconstrained case) and Figure 10 (for the constrained case) as functions of N.

Figures 9 and 10 confirm previous results on a larger scale, namely that, with the current architecture, the constrained VA-RP outperforms the unconstrained VA-RP. Regarding bias, for example, it produces less erratic results. Compared to the AJ approximation, the VA-RP approximations exhibit slightly larger biases. The gap decreases as the number of data increases as expected, but only in the constrained case. Regarding the sizes of the credibility intervals (the values of $\mathcal{W}^N$), they are generally smaller in the unconstrained case. This result is consistent with the observation made in Figure 3 concerning the marginal distribution of β. This shows smaller values than with the AJ approximation of the Jeffreys prior. It is important to emphasize that it is primarily the lower bounds of the credibility intervals that are "penalized" with the constrained VA-RP approximation. The upper bound remains consistent with that of the AJ, as reflected in the values of $\mathcal{Q}^N$. This result illustrates the relevance of using the constrained VA-RP approach for safety study. Indeed, in such a context, focusing on the upper quantile is consistent with a conservative approach.

6 CONCLUSION

Estimating seismic fragility curves when the available data are scarce is a daunting task. While the performances of a Bayesian framework in such settings are appealing, its implementation cannot overlook the prior selection process. Previous works have proven the strengths and highlighted the weaknesses of the reference prior theory incorporation in this context, with one major drawback being the computational complexity of deriving the reference prior.

In this work, we introduced a novel framework for estimating fragility curves using a variational approximation of the reference prior (VA-RP). The variational approximation that we

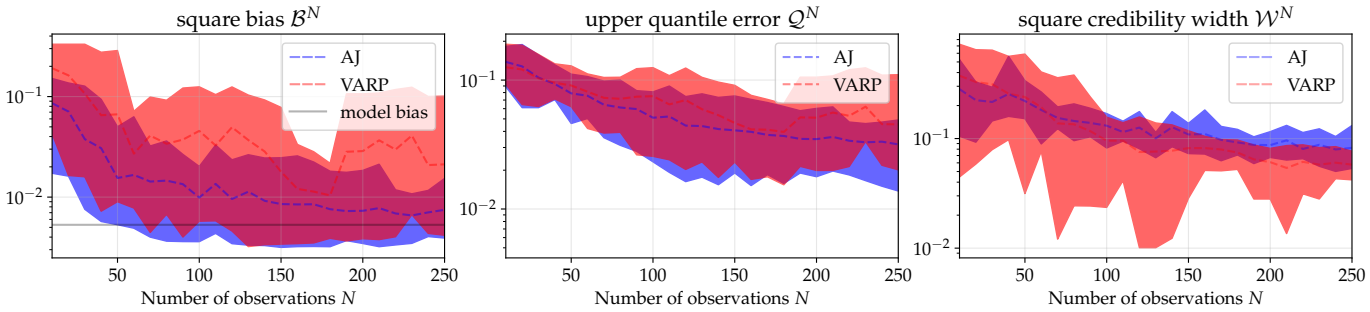


Figure 9: Average errors (dashed line) $\mathcal{B}^N$, $\mathcal{Q}^N$, $\mathcal{W}^N$ and their 95% confidence intervals, derived from numerical replication of the unconstrained methods for each value of N in $\{10, 20, \cdots, 250\}$. The values derived using the VA-RP (red) are compared with the ones using a numerical derivation to approximate the Jeffreys prior (blue). In the left figure, the square model bias $\mathcal{MB}$ is plotted (solid line).

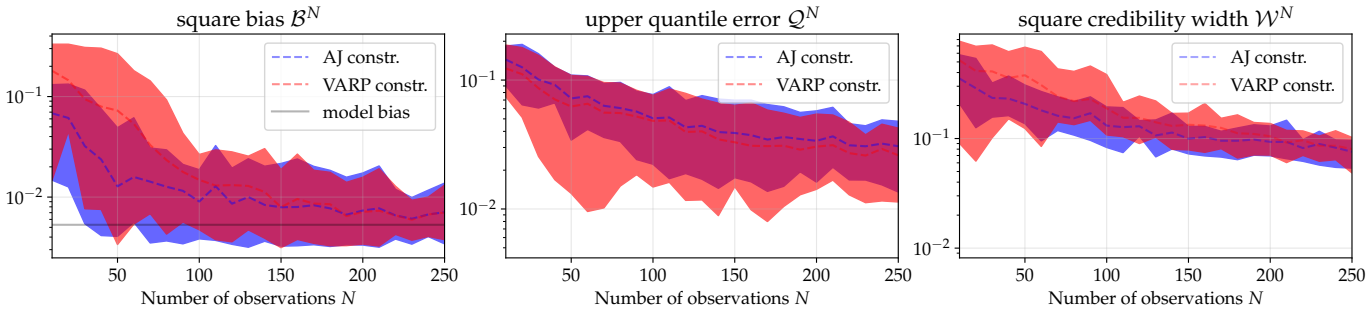


Figure 10: Same as Figure 9, but when a constraint is incorporated in the priors.

propose results from a neural network architecture specifically tailored for this application. We proved that the VA-RP accurately captures the asymptotical behavior of the reference prior through the training process. Additionally, we addressed the issue of improper posteriors by incorporating an adequate constraint into the prior. This framework facilitates an objective Bayesian estimation of seismic fragility curves, with the provision of estimates without requiring to evaluate a complex prior.

Eventually, our case study taken from the nuclear industry validated the effectiveness of the VA-RP in producing fragility curves estimates comparable to those obtained with the actual reference prior.

ACKNOWLEDGMENT

This research was supported by the CEA (French Alternative Energies and Atomic Energy Commission) and the SEISM Institute (`https://www.institut-seism.fr/en/`).

A APPENDIX: DECAY RATES OF THE VA-RP

In this appendix we elucidate the form taken by the marginal distribution of the VA-RP w.r.t. β. The VA-RP is assumed to be expressed as presented in Section 4.2 (Equation (27)): the random variable β is defined as

$$\beta = \exp\left[\sum_{i,\, w_i \neq 0} w_{2i} \log(1 - \Phi(\varepsilon_i)) + b_i\right], \tag{34}$$

where the $(\varepsilon_i)_i$ are independent standard Gaussian variables. We recall that the weights $(w_{2i})_i$ are initialized randomly w.r.t. a Gaussian distribution. Therefore, we reasonably suppose that they have distinct absolute values. We rely on the two following lemmas.

Lemma 1. *Let X be a random variable following a standard Gaussian distribution, and $\mu > 0$. Then*

- *the r.v. $Z = -\mu \log(1 - \Phi(X))$ admits the density f_Z defined by $f_Z(z) = \frac{e^{-\frac{z}{\mu}}}{\mu} \mathbb{1}_{z>0}$, it is an exponential distribution with parameter μ^{-1};*
- *the r.v $\tilde{Z} = \mu \log(1 - \Phi(X))$ admits the density $f_{\tilde{Z}}$ defined by $f_{\tilde{Z}}(z) = \frac{e^{\frac{z}{\mu}}}{\mu} \mathbb{1}_{z<0}$, we call it an opposite exponential distribution with parameter $-\mu^{-1}$.*

Lemma 2. *Let $Z_1, \dots, Z_n$ be independent exponential or opposite exponential distribution with parameters $(\mu_i)_{i=1}^n$, distinct in absolute value. Their sum $\overline{Z}$ admits a density $f_{\overline{Z}}$ that takes the form*

$$f_{\overline{Z}}(z) = \sum_{i,\, \mu_i>0} K_i e^{-\mu_i z} \mathbb{1}_{z>0} + \sum_{i,\, \mu_i<0} K_i e^{-\mu_i z} \mathbb{1}_{z<0}, \tag{35}$$

for some constants $K_1, \dots, K_n$.

The expression of β as written in Equation (34) expresses $\log(\beta - b_2)$ as a sum of independent exponential or opposite exponential distributions with parameters $(|w_{2i}|^{-1})_i$, which are distinct. Thus, the r.v. $\log(\beta - b_2)$ admits a density $\tilde{p}$ that is defined as in Equation (35):

$$\tilde{p}(z) = \sum_{i,\, w_{2i}>0} K_i e^{-\frac{z}{w_{2i}}} \mathbb{1}_{z>0} + \sum_{i,\, w_{2i}<0} K_i e^{-\frac{z}{w_{2i}}} \mathbb{1}_{z<0}. \tag{36}$$

Given that $p_\beta(\beta e^{b_2}) e^{b_2} \propto \tilde{p}(\log \beta)/\beta$ there exists some constants $(\tilde{K}_i)_{i=1}^n$ such that

$$p_\beta(\beta) = \sum_{i,\, w_{2i}>0} \tilde{K}_i \beta^{-\frac{1}{w_{2i}}-1} \mathbb{1}_{\beta>e^{b_2}} + \sum_{i,\, w_{2i}<0} \tilde{K}_i \beta^{-\frac{1}{w_{2i}}-1} \mathbb{1}_{\beta<e^{b_2}}. \tag{37}$$

Proof of Lemma 1 First, Φ being the c.d.f. of a standard Gaussian distribution, $\Phi(X)$ follows an uniform distribution in $(0, 1)$. The density $f_Z(z) = \frac{e^{-\frac{z}{\mu}}}{\mu}$ involved in the first statement of the lemma is the p.d.f. of an exponential distribution with parameter μ^{-1}. The c.d.f. of that distribution is defined by $F_Z(z) = -\mu \log(1 - z)$. Thus, $-\mu \log(1 - \Phi(X))$ is an exponential distribution with parameter μ^{-1}.
To conclude, notice that the second statement of the lemma simply results by elucidating the p.d.f. of $\tilde{Z}$ from the one of Z given that $\tilde{Z} = -Z$.

Proof of Lemma 2 We prove this result by induction over n. When $n = 1$, the form in Equation (35) is consistent with the p.d.f. of an exponential or opposite exponential distribution.

We suppose this statement true for $n - 1$ such r.v., we denote by $\tilde{f}$ the p.d.f. of $\sum_{i=1}^{n-1} Z_i$, and by f_{Z_n} the one of Z_n. As $\sum_{i=1}^{n-1} Z_i$ and Z_n are independent, $f_{\overline{Z}}$ equals the convolution between $\tilde{f}$ and f_{Z_n}: $f_{\overline{Z}} = \tilde{f} * f_{Z_n}$.

Before going further, let us derive the following integrals, for any $0 < \nu_1 < \nu_2$:

$$\begin{aligned}
&\int_{\mathbb{R}} e^{-\nu_1(y-x)} e^{-\nu_2 x} \mathbb{1}_{y-x>0} \mathbb{1}_{x>0} dx = \frac{e^{-\nu_2 y} - e^{-\nu_1 y}}{\nu_2 - \nu_1} \mathbb{1}_{y>0}, \\
&\int_{\mathbb{R}} e^{-\nu_1(y-x)} e^{\nu_2 x} \mathbb{1}_{y-x>0} \mathbb{1}_{x<0} dx = \frac{e^{\nu_2 y}}{\nu_1 + \nu_2} \mathbb{1}_{y<0} + \frac{e^{-\nu_1 y}}{\nu_1 + \nu_2} \mathbb{1}_{y>0}, \\
&\int_{\mathbb{R}} e^{\nu_1(y-x)} e^{\nu_2 x} \mathbb{1}_{y-x<0} \mathbb{1}_{x<0} dx = \frac{e^{\nu_1 y} - e^{\nu_2 y}}{\nu_2 - \nu_1} \mathbb{1}_{y<0}.
\end{aligned} \tag{38}$$

Thus, if $f_{Z_n}(x) = \mu_n e^{-\mu_n x} \mathbb{1}_{x>0}$ for $\mu_n > 0$ we get

$$\begin{aligned}
f_{\overline{Z}}(z) = &\sum_{i<n,\, \mu_i>0} \frac{K_i \mu_n}{\mu_i - \mu_n} e^{-\mu_i z} \mathbb{1}_{z>0} + \sum_{i<n,\, \mu_i<0} \frac{K_i \mu_n}{\mu_i + \mu_n} e^{-\mu_i z} \mathbb{1}_{z<0} \\
&- e^{-\mu_n z} \left[\sum_{i,\, \mu_i>0} \frac{K_i \mu_n}{\mu_i - \mu_n} + \sum_{i,\, \mu_i<0} \frac{K_i \mu_n}{\mu_i + \mu_n} \right] \mathbb{1}_{z>0};
\end{aligned} \tag{39}$$

and if $f_{Z_n}(x) = \mu_n e^{\mu_n x} \mathbb{1}_{x<0}$ for $\mu_n > 0$ then

$$\begin{aligned}
f_{\overline{Z}}(z) = &\sum_{i,\, \mu_i>0} \frac{K_i \mu_n}{\mu_i + \mu_n} e^{-\mu_i z} \mathbb{1}_{z>0} + \sum_{i,\, \mu_i<0} \frac{K_i \mu_n}{\rho - \mu_n} e^{-\mu_i z} \mathbb{1}_{z<0} \\
&+ e^{\mu_n z} \left[\sum_{i,\, \mu_i>0} \frac{K_i \mu_n}{\mu_i + \mu_n} + \sum_{i,\, \mu_i<0} \frac{K_i \mu_n}{\mu_i - \mu_n} \right] \mathbb{1}_{z<0}.
\end{aligned} \tag{40}$$

In any case, it fits the expected form.

B APPENDIX: VARIANCE OF α A PRIORI

In this section we consider the asymptotical version of the reference prior, that we denote by π_a^γ:

$$\pi_a^\gamma(\theta) = \frac{K}{\beta^{1-\gamma} + \beta^{3-\gamma}} \frac{|\log \alpha|}{\alpha} \exp\left(-\frac{(\log \alpha - \mu)^2}{2\beta^2 + 2\sigma^2} \right), \tag{41}$$

where $\mu \in \mathbb{R}$, $\sigma > 0$. The parameter γ is non null if the reference prior is not constrained, it lives in $(0, 2)$ if a constraint is incorporated as elucidated in Section 4.3. K represents a normalization constant that depends only on γ.

We fix $\beta > 0$, let us consider a random variable $X \sim \mathcal{N}(\mu, \Sigma^2)$, where $\Sigma = \Sigma(\beta) = \sqrt{\sigma^2 + \beta^2}$. We can write

$$\mathbb{E}|X| = \int_{-\infty}^{\infty} \frac{|x|}{\sqrt{2\pi\Sigma^2}} e^{-\frac{(x-\mu)^2}{2\Sigma^2}} dx = \frac{1}{\sqrt{2\pi\Sigma^2}} \int_0^{\infty} \frac{|\log y|}{y} e^{-\frac{(\log y - \mu)^2}{2\Sigma^2}} dy \tag{42}$$

$$\text{and} \quad \mathbb{E}|X|^3 = \int_{-\infty}^{\infty} \frac{|x|^3}{\sqrt{2\pi\Sigma^2}} e^{-\frac{(x-\mu)^2}{2\Sigma^2}} dx = \frac{1}{\sqrt{2\pi\Sigma^2}} \int_0^{\infty} \frac{|\log y|^3}{y} e^{-\frac{(\log y - \mu)^2}{2\Sigma^2}} dy. \tag{43}$$

Equation (42) elucidates the conditional distribution of $\alpha|\beta$ as the one whose density is

$$p_a^\gamma(\alpha|\beta) = (\mathbb{E}|X|\sqrt{2\pi\Sigma^2})^{-1} \frac{|\log \alpha|}{\alpha} \exp\left(-\frac{(\log \alpha - \mu)^2}{2\beta^2 + 2\sigma^2} \right); \tag{44}$$

and the marginal distribution of β as the one whose density is

$$p_a^\gamma(\beta) = \frac{K(\mathbb{E}|X|\sqrt{2\pi\Sigma^2})}{\beta^{1-\gamma} + \beta^{3-\gamma}}. \tag{45}$$

This way,

$$\mathbb{E}[|\log\alpha|^2 \,|\, \beta] = \frac{\mathbb{E}|X|^3}{\mathbb{E}|X|}. \tag{46}$$

The right hand term of the above equation equals the following (see [51]):

$$\begin{aligned} \mathbb{E}|X|^3 &= 2^{3/2}\Sigma^3 \frac{\Gamma(2)}{\Gamma(-\frac{3}{2})} \sum_{k=0}^{\infty} \frac{\Gamma(-\frac{3}{2}+k)}{\Gamma(\frac{1}{2}+k)} \frac{\left(-\frac{\mu}{2\Sigma^2}\right)^k}{k!}, \\ \text{and} \quad \mathbb{E}|X| &= 2^{1/2}\Sigma \frac{\Gamma(1)}{\Gamma(-\frac{1}{2})} \sum_{k=0}^{\infty} \frac{\Gamma(-\frac{1}{2}+k)}{\Gamma(\frac{1}{2}+k)} \frac{\left(-\frac{\mu}{2\Sigma^2}\right)^k}{k!}. \end{aligned} \tag{47}$$

Using the identity $z\Gamma(z) = \Gamma(z+1)$, we derive

$$\frac{\Gamma(-\frac{3}{2}+k)}{\Gamma(\frac{1}{2}+k)} = \frac{\Gamma(-\frac{1}{2}+k)}{(-\frac{3}{2}+k)\Gamma(\frac{1}{2}+k)} = \frac{1}{(-\frac{3}{2}+k)(-\frac{1}{2}+k)}. \tag{48}$$

Going back to Equation (46), we deduce:

$$\mathbb{E}|\log\alpha|^2 = \mathbb{E}\left[\mathbb{E}[|\log\alpha|^2 \,|\, \beta]\right] = K' \int_0^\infty \frac{(\beta^2+\sigma^2)^2}{\beta^{1-\gamma}+\beta^{3-\gamma}} \sum_{k=0}^{\infty} \frac{(-\mu(2\sigma^2+2\beta^2)^{-1})^k}{(2k-3)(2k-1)k!} d\beta, \tag{49}$$

with $K' > 0$ being a normalization constant. To conclude, we perform an asymptotical analysis of the integrated function in the above equation as $\beta \longrightarrow \infty$. The power series

$$\sum_{k=0}^{\infty} \frac{z^k}{(2k-3)(2k-1)k!} \tag{50}$$

has an infinite radius of convergence, so that when $z \longrightarrow 0$, it converges toward $\frac{1}{3}$. Also, while z is a negative real number, the power series is positive.

Therefore, the term in the integral of Equation (49) is positive and asymptotically equivalent to $\beta^{1+\gamma}$ as $\beta \longrightarrow \infty$. We deduce that for any $\gamma \geq 0$,

$$\mathbb{E}[|\log\alpha|^2] = \infty. \tag{51}$$

REFERENCES

[1] R. P. Kennedy, C. A. Cornell, R. D. Campbell, S. J. Kaplan, F. Harold, Probabilistic seismic safety study of an existing nuclear power plant, *Nuclear Engineering and Design,* **59**, 315–338, 1980.

[2] R. P. Kennedy, M. K. Ravindra, Seismic fragilities for nuclear power plant risk studies, *Nuclear Engineering and Design,* **79**, 47–68, 1984.

[3] B. Ellingwood, Validation studies of seismic PRAs, *Nuclear Engineering and Design,* **123**, 189–196, 1990.

[4] Y.-J. Park, C. H. Hofmayer, N. C. Chokshi, Survey of seismic fragilities used in PRA studies of nuclear power plants, *Reliability Engineering & System Safety,* **62**, 185–195, 1998.

[5] C. A. Cornell, Hazard, ground motions and probabilistic assessments for PBSD, in: *Proceedings of the International Workshop on Performance-Based Seismic Design - Concepts and Implementation, PEER Center, University of California, Berkeley, 39–52,* BLED, SLOVENIA, 2004.

[6] N. Luco, C. A. Cornell, Structure-specific scalar intensity measures for near-source and ordinary earthquake ground motions, *Earthquake Spectra,* **23**, 357–392, 2007.

[7] B. R. Ellingwood, Earthquake risk assessment of building structures, *Reliability Engineering & System Safety,* **74**, 251–262, 2001.

[8] S.-H. Kim, M. Shinozuka, Development of fragility curves of bridges retrofitted by column jacketing, *Probabilistic Engineering Mechanics,* **19**, 105–112, 2004.

[9] I. Zentner, Numerical computation of fragility curves for NPP equipment, *Nuclear Engineering and Design,* **240**, 1614–1621, 2010.

[10] P. S. Koutsourelakis, Assessing structural vulnerability against earthquakes using multi-dimensional fragility surfaces: A Bayesian framework, *Probabilistic Engineering Mechanics,* **25**, 49–60, 2010.

[11] C. Mai, K. Konakli, B. Sudret, Seismic fragility curves for structures using non-parametric representations, *Frontiers of Structural and Civil Engineering,* **11**, 169–186, 2017.

[12] K. Trevlopoulos, C. Feau, I. Zentner, Parametric models averaging for optimized non-parametric fragility curve estimation based on intensity measure data clustering, *Structural Safety,* **81**, 101865, 2019.

[13] F. Wang, C. Feau, Influence of input motion's control point location in nonlinear SSI analysis of equipment seismic fragilities: Case study on the Kashiwazaki-Kariwa NPP, *Pure and Applied Geophysics,* **177**, 2391–2409, 2020.

[14] T. K. Mandal, S. Ghosh, N. N. Pujari, Seismic fragility analysis of a typical indian PHWR containment: Comparison of fragility models, *Structural Safety,* **58**, 11–19, 2016.

[15] Z. Wang, N. Pedroni, I. Zentner, E. Zio, Seismic fragility analysis with artificial neural networks: Application to nuclear power plant equipment, *Engineering Structures,* **162**, 213–225, 2018.

[16] Z. Wang, I. Zentner, E. Zio, A Bayesian framework for estimating fragility curves based on seismic damage data and numerical simulations by adaptive neural networks, *Nuclear Engineering and Design,* **338**, 232–246, 2018.

[17] C. Zhao, N. Yu, Y. Mo, Seismic fragility analysis of AP1000 SB considering fluid-structure interaction effects, *Structures,* **23**, 103–110, 2020.

[18] Y. Katayama, Y. Ohtori, T. Sakai, H. Muta, Bayesian-estimation-based method for generating fragility curves for high-fidelity seismic probability risk assessment, *Journal of Nuclear Science and Technology,* **58**, 1220–1234, 2021.

[19] C. Gauchy, C. Feau, J. Garnier, Importance sampling based active learning for parametric seismic fragility curve estimation, 2021. URL: https://arxiv.org/abs/2109.04323, arXiv.

[20] A. Khansefid, S. M. Yadollahi, F. Taddei, G. Müller, Fragility and comfortability curves development and seismic risk assessment of a masonry building under earthquakes induced by geothermal power plants operation, *Structural Safety,* **103**, 102343, 2023.

[21] S. Lee, S. Kwag, B.-s. Ju, On efficient seismic fragility assessment using sequential bayesian inference and truncation scheme: A case study of shear wall structure, *Computers & Structures,* **289**, 107150, 2023.

[22] M. Shinozuka, M. Q. Feng, J. Lee, T. Naganuma, Statistical analysis of fragility curves, *Journal of Engineering Mechanics,* **126**, 1224–1231, 2000.

[23] D. Lallemant, A. Kiremidjian, H. Burton, Statistical procedures for developing earthquake damage fragility curves, *Earthquake Engineering & Structural Dynamics,* **44**, 1373–1389, 2015.

[24] D. Straub, A. Der Kiureghian, Improved seismic fragility modeling from empirical data, *Structural Safety,* **30**, 320–336, 2008.

[25] P. Gardoni, A. Der Kiureghian, K. M. Mosalam, Probabilistic capacity models and fragility estimates for reinforced concrete columns based on experimental observations, *Journal of Engineering Mechanics,* **128**, 1024–1038, 2002.

[26] G. Damblin, M. Keller, A. Pasanisi, P. Barbillon, É. Parent, Approche décisionnelle bayésienne pour estimer une courbe de fragilité, *Journal de la Societe Française de Statistique,* **155**, 78–103, 2014.

[27] S. K. Tadinada, A. Gupta, Structural fragility of t-joint connections in large-scale piping systems using equivalent elastic time-history simulations, *Structural Safety,* **65**, 49–59, 2017.

[28] S. Kwag, A. Gupta, Computationally efficient fragility assessment using equivalent elastic limit state and Bayesian updating, *Computers & Structures,* **197**, 1–11, 2018.

[29] J.-S. Jeon, S. Mangalathu, J. Song, R. Desroches, Parameterized seismic fragility curves for curved multi-frame concrete box-girder bridges using bayesian parameter estimation, *Journal of Earthquake Engineering,* **23**, 954–979, 2019.

[30] A. Tabandeh, P. Asem, P. Gardoni, Physics-based probabilistic models: Integrating differential equations and observational data, *Structural Safety,* **87**, 101981, 2020.

[31] A. Van Biesbroeck, C. Gauchy, C. Feau, J. Garnier, Reference prior for bayesian estimation of seismic fragility curves, *Probabilistic Engineering Mechanics,* **76**, 103622, 2024.

[32] A. Van Biesbroeck, C. Gauchy, C. Feau, J. Garnier, Influence of the choice of the seismic intensity measure on fragility curves estimation in a bayesian framework based on reference prior, in: *Proceedings of the 5th UNCECOMP Conference, 94–111,* Athens, Greece, 2023.

[33] A. V. Biesbroeck, C. Gauchy, C. Feau, J. Garnier, Robust a posteriori estimation of probit-lognormal seismic fragility curves via sequential design of experiments and constrained reference prior, 2025. URL: https://arxiv.org/abs/2503.07343. arXiv:2503.07343.

[34] J. M. Bernardo, Expected Information as Expected Utility, *The Annals of Statistics,* **7**, 686–690, 1979.

[35] N. Baillie, A. Van Biesbroeck, C. Gauchy, Variational inference for approximate reference priors using neural networks, 2025. URL: https://arxiv.org/abs/2502.02364. arXiv:2502.02364.

[36] J. O. Berger, J. M. Bernardo, D. Sun, The formal definition of reference priors, *The Annals of Statistics,* **37**, 905–938, 2009.

[37] A. Van Biesbroeck, Generalized mutual information and their reference priors under csizar f-divergence, 2024. URL: https://arxiv.org/abs/2310.10530. arXiv:2310.10530, arXiv.

[38] B. S. Clarke, A. R. Barron, Jeffreys' prior is asymptotically least favorable under entropy risk, *Journal of Statistical Planning and Inference,* **41**, 37–60, 1994.

[39] A. Van Biesbroeck, Properly constrained reference priors decay rates for efficient and robust posterior inference, 2024. URL: https://arxiv.org/abs/2409.13041. arXiv:2409.13041, arXiv.

[40] D. P. Kingma, M. Welling, Auto-Encoding Variational Bayes, in: *Proceedings of the 2nd International Conference on Learning Representations (ICLR),* Banff, Canada, 2014.

[41] C. Gauchy, A. Van Biesbroeck, C. Feau, J. Garnier, Inférence variationnelle de lois a priori de référence, in: *Proceedings des 54èmes Journées de Statistiques (JdS),* SFDS, 2023.

[42] E. Nalisnick, P. Smyth, Learning approximately objective priors, in: *Proceedings of the 33rd Conference on Uncertainty in Artificial Intelligence (UAI), Association for Uncertainty in Artificial Intelligence (AUAI),* Sydney, Australia, 2017.

[43] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, Pytorch: An imperative style, high-performance deep learning library, in: *H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems, volume 32,* Curran Associates, Inc., 2019.

[44] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: *Proceedings of the 3rd International Conference on Learning Representations (ICLR),* San Diego, USA, 2015.

[45] S. Rezaeian, A. Der Kiureghian, Simulation of synthetic ground motions for specified earthquake and site characteristics, *Earthquake Engineering & Structural Dynamics,* **39**, 1155–1180, 2010.

[46] R. Sainct, C. Feau, J.-M. Martinez, J. Garnier, Efficient methodology for seismic fragility curves estimation by active learning on support vector machines, *Structural Safety,* **86**, 101972, 2020.

[47] N. N. Ambraseys, P. Smit, R. Berardi, D. Rinaldis, F. Cotton, C. Berge, Dissemination of european strongmotion data, 2000. CD-ROM collection. European Commission, Directorate-General XII, Environmental and Climate Programme, ENV4-CT97-0397, Brussels, Belgium.

[48] F. Touboul, P. Sollogoub, N. Blay, Seismic behaviour of piping systems with and without defects: experimental and numerical evaluations, *Nuclear Engineering and Design,* **192**, 243–260, 1999.

[49] CEA, CAST3M, 2019. URL: http://www-cast3m.cea.fr/.

[50] F. Touboul, N. Blay, P. Sollogoub, S. Chapuliot, Enhanced seismic criteria for piping, *Nuclear Engineering and Design,* **236**, 1–9, 2006.

[51] A. Winkelbauer, Moments and absolute moments of the normal distribution, 2014. URL: https://arxiv.org/abs/1209.4340. arXiv:1209.4340.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

LABEL-EFFICIENT CARDIAC VOLUME ESTIMATION IN REAL-TIME MRI WITH SPATIAL CORRELATION

Felix Terhag[1,2], Philipp Knechtges[1], Anja Bach[4], Darius Gerlach[4], Jens Tank[4], Achim Basermann[1] and Raúl Tempone[2,3]

[1]Institute of Software Technology, German Aerospace Center (DLR)
Linder Hoehe, 51147, Cologne, Germany
e-mail: {felix.terhag, philipp.knechtges, achim.basermann}@dlr.de

[2]Chair of Mathematics for Uncertainty Quantification
RWTH Aachen University, Aachen, Germany
e-mail: tempone@uq.rwth-aachen.de

[3]Computer, Electrical and Mathematical Sciences and Engineering Division (CEMSE)
King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia

[4]Institute of Aerospace Medicine, German Aerospace Center (DLR)
Linder Hoehe, 51147, Cologne, Germany
e-mail: {anja.bach, darius.gerlach, jens.tank}@dlr.de

Abstract. *Real-time magnetic resonance imaging (MRI) enables imaging under free-breathing conditions, allowing researchers to study the influence of breathing on the heartbeat. However, this advantage requires handling large amounts of data, exceeding 4,500 images per heart, making manual segmentation of the ventricular volume highly labor-intensive and costly.Hence, maximizing the information gained from each labeled image is essential to improve label efficiency.*

We propose a Bayesian model for predicting ventricular volumes with minimal manual labeling. Unlike standard approaches that treat each slice independently, the proposed model incorporates spatial correlations by enforcing smoothness via a discretized one-dimensional Laplace operator, avoiding strong geometric assumptions. Additionally, this study exploits conjugacy to compute the posterior analytically, eliminating the need for sampling or variational inference and allowing for efficient hyperparameter optimization via the type-II maximum likelihood. This approach enables the proposed model to employ spatial information effectively, improving predictions for slices with few labeled images.

This work compares the proposed model to a baseline model that treats each slice independently. In the model evaluation on real patient data, the proposed method outperformed the baseline, particularly in data-scarce scenarios, where spatial correlation considerably enhances prediction accuracy.

Keywords: Bayesian Modeling, Real-time Magnetic Resonance Imaging, Empirical Bayes, Label Efficiency, Maximum Likelihood Estimation

1 INTRODUCTION

Understanding how breathing influences the heartbeat is of great interest in cardiac research. However, observing this relationship is challenging because traditional cine magnetic resonance imaging (MRI) is typically performed using a breath-holding protocol. With the emergence of real-time MRI, which can capture up to 50 frames per second (fps), breath-holding is no longer required, allowing for the direct observation of the effects of breathing on the heartbeat [14, 9, 13].

An important parameter in cardiac research is the ventricular volume over time, requiring ventricle segmentation in each image. However, real-time MRI is challenging because it generates significantly more data. For instance, the study dataset, captured at 30 fps, requires at least 10 s per slice to record about one to four breathing cycles, depending on the breathing speed. With a minimum of 15 slices per heart, this results in over 4,500 images requiring annotation, making manual segmentation highly labor-intensive.

Furthermore, the study dataset comprises patients with univentricular hearts, a rare congenital heart disease in which patients are born with a single functioning ventricle and who undergo multiple surgical interventions that significantly alter cardiac anatomy. This situation increases the difficulty of applying automatic segmentation. A more detailed description of the study that acquired the data is presented in [8].

The large volume of data and high segmentation cost require a label-efficient approach. Previous work has modeled the ventricular volume over time as a superposition of frequencies and employed sparse Bayesian learning to identify a sparse set of relevant frequencies [11]. Fewer labels are required to fit the model because only a few frequencies are necessary to describe the dynamics. Building upon this framework, we aim to incorporate spatial information while accounting for the complex and heterogeneous geometries of univentricular hearts.

Given this variability, strict geometric assumptions are minimized. Instead, this work applies the discretized one-dimensional (1D) Laplace operator to enforce smoothness in a flexible manner. This study constructs a Bayesian model incorporating these geometric correlations into the prior, ensuring that the likelihood dominates when sufficient data are available. This way, the proposed model enforces smoothness when necessary but gives more weight to the data when sufficient observations are available.

Furthermore, the proposed model utilizes conjugacy to compute the posterior analytically. This approach makes the procedure more efficient, eliminating the need for sampling or variational inference to approximate the posterior. Moreover, the proposed model enables direct computation of the marginal likelihood gradient with respect to the hyperparameters. Thus, gradient-based methods, such as BFGS , can efficiently determine the maximum of the type-II likelihood.

In [11], a similar model is applied to real-time MRI without accounting for spatial correlations. Similar models are also widely applied in remote sensing applications [4, 5]. In [15], the authors proposed a related approach for signal recovery incorporating correlation, though in the time domain. However, their method assumes all sources, analogous to slices in our case, are mutually independent. While modeling correlations along the slice dimension is less relevant in remote sensing, it is crucial for real-time MRI applications.

2 METHODS

2.1 Modeling

In contrast to cine MRI, real-time MRI allows video sequences of the beating heart to be retrieved due to a faster capture rate. The heart is captured in L slices, where every slice $\ell \in 1, \dots, L$ comprises a video of N images. The images are captured with a constant capture rate at time steps $t_1, \dots, t_N$. This work is interested in the ventricle volume over time and assumes that it can be described as a superposition of $m+1$ frequencies $\mathcal{F} = \{f_0 = 0, f_1, f_2, \dots, f_m\}$, where the main contributing frequencies are the heart and respiratory rate. Limiting the number of frequencies m benefits parameter estimation, and $M := 2m+1$ is introduced to facilitate the notation. A sparse set of frequencies that adequately describes the data is determined using sparse Bayesian learning [11]. The proposed linear model is

$$Y = AX + N, \tag{1}$$

with

- $Y \in \mathbb{R}^{N \times L}$ the real-valued measurement points $y_{k,\ell}$,
- $X \in \mathbb{R}^{M \times L}$ the real and imaginary parts of the complex amplitudes,
- $N \in \mathbb{R}^{N \times L}$ additive noise with $n_{k,\ell} \sim \mathcal{N}(0, \sigma^2)$,
- $A \in \mathbb{R}^{N \times M}$ the transfer matrix $A = [\mathbf{1}_N \; Re(B) \; Im(B)]$, and
- with $\mathbf{1}_N$ an N-dimensional vector of ones and $B \in \mathbb{C}^{N \times m}$ defined as,

$$B = \begin{bmatrix} \exp(-i2\pi f_1 t_1) & \dots & \exp(-i2\pi f_m t_1) \\ \vdots & \ddots & \vdots \\ \exp(-i2\pi f_1 t_N) & \dots & \exp(-i2\pi f_m t_N) \end{bmatrix}.$$

Prior distribution In [11], the prior for X is a zero-centered normal distribution, independent for all slices and frequencies. In contrast, this work assumes that the magnitudes of the frequencies depend on neighboring slices. Hence, the following matrix is introduced:

$$T = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix} \in \mathbb{R}^{L \times L},$$

which is a 1D equidistant discretization of the Laplace operator with Dirichlet-boundary conditions. The rationale is that without further geometric information, we prefer to select a prior that penalizes excessive curvature (i.e., the (discretized) second derivative) over the slices. The prior should be selected to resemble $X \sim ZT^{-1}$ with every column i of Z being normally distributed $Z_i \sim \mathcal{N}(0, sI_M)$ with variance $s > 0$ and I_M being the identity matrix of size M. With the vectorization operator $\text{vec}(\cdot)$, we obtain

$$p(\text{vec}(Z)|s) = \mathcal{N}(\text{vec}(Z)|0, sI_{ML}),$$

where $\mathcal{N}(x|\mu, \Sigma)$ denotes the probability density function of a multivariate normal distribution at location x for the mean μ and covariance matrix Σ. The identity

$$\begin{aligned}\text{vec}(ZT^{-1}) =&(T^{-T} \otimes I_M)\text{vec}(Z)\\ =&(T^{-1} \otimes I_M)\text{vec}(Z),\end{aligned}$$

yields the prior $p(\text{vec}(X)|s)$, fulfilling the demands, where

$$\begin{aligned}p(\text{vec}(X)|s) &= \mathcal{N}(\text{vec}(X)|0, (T^{-1} \otimes I_M)sI_{ML}(T^{-T} \otimes I_M))\\ &= \mathcal{N}(\text{vec}(X)|0, T^{-2} \otimes sI_M)\end{aligned} \tag{2}$$

using the mixed product property of the Kronecker product $\otimes$ and the fact that $(T^{-1} \otimes I_M)$ is symmetric.

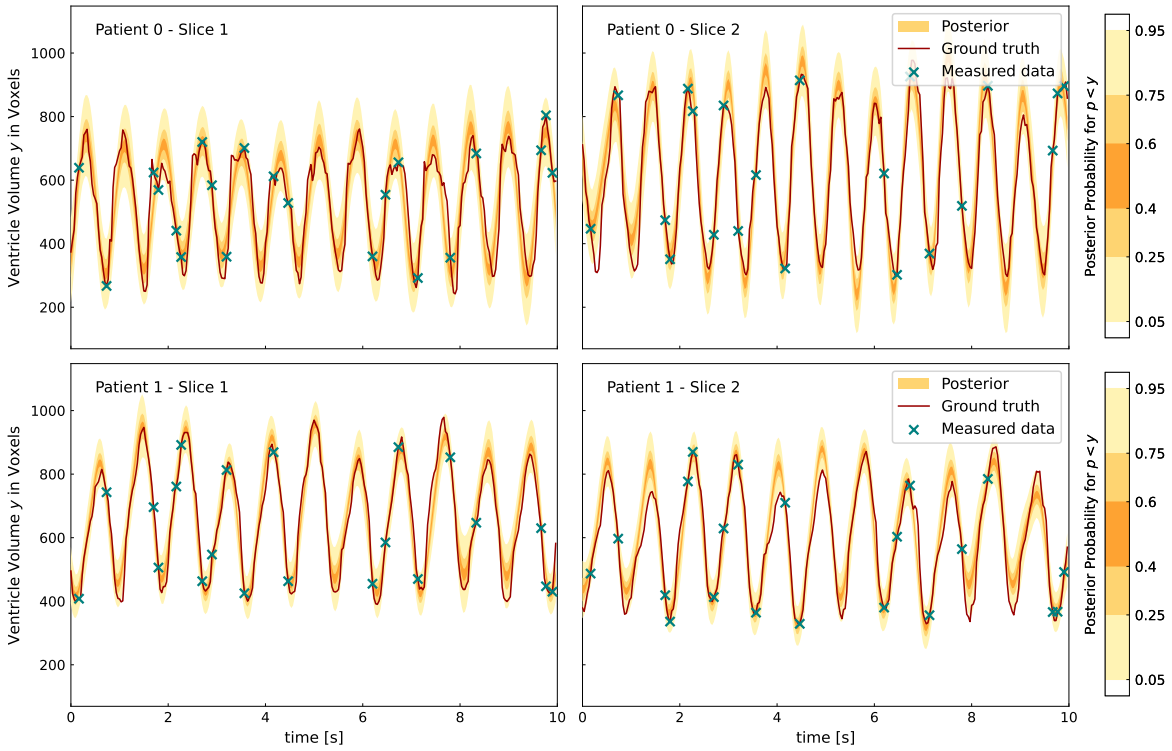


Figure 1: Ventricle volume over time for two consecutive slices in both patients. Images were captured at 30 fps, with each slice spanning 300 frames. The red curve represents the ground truth from expert-validated segmentations. The yellow curve marks the posterior predictive, based on 20 measurements per slice, indicated by a green x.

Likelihood For partially observed data Y, let $J \subset \{1, \dots, NL\}$ be the index set of the observed data in vectorized form and $P \in \mathbb{R}^{|J| \times NL}$ be the projection matrix, with entries $P_{i,J_i} = 1$ for $i \in 1, \dots, |J|$ and zero elsewhere. For the model defined in (1), the likelihood of data Y given X and σ^2 is

$$p(\text{vec}(Y)|X, \sigma^2) = \frac{\exp\left(-\frac{1}{2\sigma^2} \|P(I_L \otimes A)\text{vec}(X) - \text{vec}(Y)\|_2^2\right)}{(2\pi\sigma^2)^{|J|/2}}.$$

We define $G := P(I_L \otimes A) \in \mathbb{R}^{|J| \times ML}$ to simplify this formulation, yielding

$$p(\text{vec}(Y)|X, \sigma^2) = \frac{\exp\left(-\frac{1}{2\sigma^2} \|G\text{vec}(X) - \text{vec}(Y)\|_2^2\right)}{(2\pi\sigma^2)^{|J|/2}}. \tag{3}$$

Posterior distribution Given the likelihood (3) and prior distribution (2), assuming the parameters s and σ^2 are known the posterior of X can be calculated using the Bayes rule

$$p(X|Y, s, \sigma^2) := \frac{p(\text{vec}(Y)|X, \sigma^2)p(\text{vec}(X)|s)}{p(Y|s, \sigma^2)}. \tag{4}$$

The likelihood and prior distribution are normal distributions, allowing the analytical computation of the normal posterior distribution of X with mean μ_X and covariance Σ_X given by

$$\begin{aligned} \Sigma_X &= \left(\frac{1}{\sigma^2} G^T G + (T^{-2} \otimes sI_M)^{-1}\right)^{-1} \\ \mu_X &= \frac{1}{\sigma^2} \Sigma_X G^T \text{vec}(Y). \end{aligned} \tag{5}$$

The posterior predictive distribution of $\text{vec}(Y)$ can be directly derived from the vectorized form of (1).Thus, the distribution is normal distribution with mean μ_Y and covariance Σ_Y

$$\begin{aligned} \Sigma_Y &= (I_L \otimes A)\Sigma_X (I_L \otimes A)^T + \sigma^2 I_{NL} \\ \mu_Y &= (I_L \otimes A)\mu_X. \end{aligned} \tag{6}$$

Figure 1 presents the posterior distribution given 20 measurement points per slice. While it does not perfectly capture the ground truth, it performs well considering that only 20 out of 300 measurements per slice are used, demonstrating that the model is effective in this setting.

2.2 Empirical Bayes

The proposed model treats the parameters σ^2 and s as hyperparameters. Instead of computing the joint posterior $p(X, \sigma^2, s|Y)$, the conditional posterior (4) is used to estimate $\hat{\sigma}^2$ and $\hat{s}$. The hyperparameters are determined by maximizing the marginal likelihood

$$\hat{\sigma}^2, \hat{s} = \arg\max_{\sigma^2, s} p(Y|\sigma^2, s), \tag{7}$$

a procedure known as type-II-maximum likelihood optimization or empirical Bayes [2, 7]. In this case, the marginal likelihood is a product of the prior (2) and likelihood (3) integrated over X. Applying the data covariance $\Sigma_p := \sigma^2 I_{|J|} + G(T^{-2} \otimes sI_M)G^T$ and the Woodbury formula yields

$$p(Y|\sigma^2, s) = \int p(Y|X, \sigma^2)p(X|s)dX \propto \frac{\exp\left(-\frac{1}{2}\text{vec}(Y)^T \Sigma_p^{-1} \text{vec}(Y)\right)}{\sqrt{\det(\Sigma_p)}}. \tag{8}$$

Hence, the log-marginal likelihood for some constant value c is given by

$$\log p(Y|\sigma^2, s) = -\text{vec}(Y)^T \Sigma_p^{-1} \text{vec}(Y) - \log\det(\Sigma_p) + c.$$

Next, we can estimate $\hat{\sigma}^2$ and $\hat{s}$ in (7) by minimizing the following loss function

$$\mathcal{L}(\sigma^2, s) = \text{vec}(Y)^T \Sigma_p^{-1} \text{vec}(Y) + \log\det(\Sigma_p). \tag{9}$$

To minimize $\mathcal{L}$ using a gradient based approach, we need to calculate the derivative of $\mathcal{L}$. Using the formula for the derivative of the matrix inverse, we can calculate

$$\begin{aligned}\frac{\partial \Sigma_p^{-1}}{\partial s} &= -\Sigma_p^{-1}\frac{\partial \Sigma_p}{\partial s}\Sigma_p^{-1} \\ &= -\Sigma_p^{-1}G(T^{-2}\otimes I_M)G^T\Sigma_p^{-1}\end{aligned}$$

and

$$\frac{\partial \Sigma_p^{-1}}{\partial \sigma^2} = -\Sigma_p^{-2}.$$

Furthermore, Jacobi's formula results in

$$\frac{\partial \log\det(\Sigma_p)}{\partial s} = \text{tr}\left(\Sigma_p^{-1}\frac{\partial \Sigma_p}{\partial s}\right) = \text{tr}\left(\Sigma_p^{-1}G(T^{-2}\otimes I_M)G^T\right)$$

and analogously,

$$\frac{\partial \log\det(\Sigma_p)}{\partial \sigma^2} = \text{tr}\left(\Sigma_p^{-1}\right).$$

Combining these results leads to the following derivatives

$$\begin{aligned}\frac{\partial \mathcal{L}(\sigma^2, s)}{\partial s} &= \text{tr}\left(\Sigma_p^{-1}G(T^{-2}\otimes I_M)G^T\right) - \text{vec}(Y)^T\Sigma_p^{-1}G(T^{-2}\otimes I_M)G^T\Sigma_p^{-1}\text{vec}(Y) \\ \frac{\partial \mathcal{L}(\sigma^2, s)}{\partial \sigma^2} &= \text{tr}\left(\Sigma_p^{-1}\right) - \text{vec}(Y)^T\Sigma_p^{-2}\text{vec}(Y),\end{aligned} \tag{10}$$

allowing the use of gradient-based methods, such as the BFGS-algorithm [1, 3, 6, 10].

3 RESULTS

Real-time MRI is an emerging technology that captures images at up to 50 fps, allowing the observation of respiratory effects on the heartbeat by removing the need for the breath-holding protocol in traditional cine MRI [14, 9, 13]. The volume of the ventricle over time - ventricular filling - is of clinical relevance and is used to estimate important parameters like stroke volume. An expert must segment the ventricle to derive the ventricular filling, which is expensive and tedious. Real-time MRI produces considerable data; thus, more images must be segmented. The study dataset comprises data from two patients, with four consecutive slices of 300 images completely labeled. This work applies empirical Bayes to estimate hyperparameters s and σ^2 by maximizing the marginal likelihood (8), corresponding to minimizing the loss function (9). We employed gradient-based optimization methods using the gradient (10). The resulting optimization identifies the minimum of the loss function, as illustrated in the loss landscape in Figure 2.

To assess the stability of the proposed method, we employed a Laplace approximation, which approximates the log-marginal likelihood with a normal distribution centered at its maximum $\hat{\sigma}^2, \hat{s}$. The covariance of this normal distribution is given by the inverse Hessian H^{-1} of the log-marginal likelihood evaluated at $(\hat{\sigma}^2, \hat{s})$. This method results in the following normal approximation:

$$\mathcal{N}\left((\sigma^2, s) \mid (\hat{\sigma}^2, \hat{s}), H^{-1}\right). \tag{11}$$

We numerically approximated the Hessian and applied it to compute the 95% confidence region in Figure 2. To further evaluate the stability of the proposed method, we sampled 50 pairs

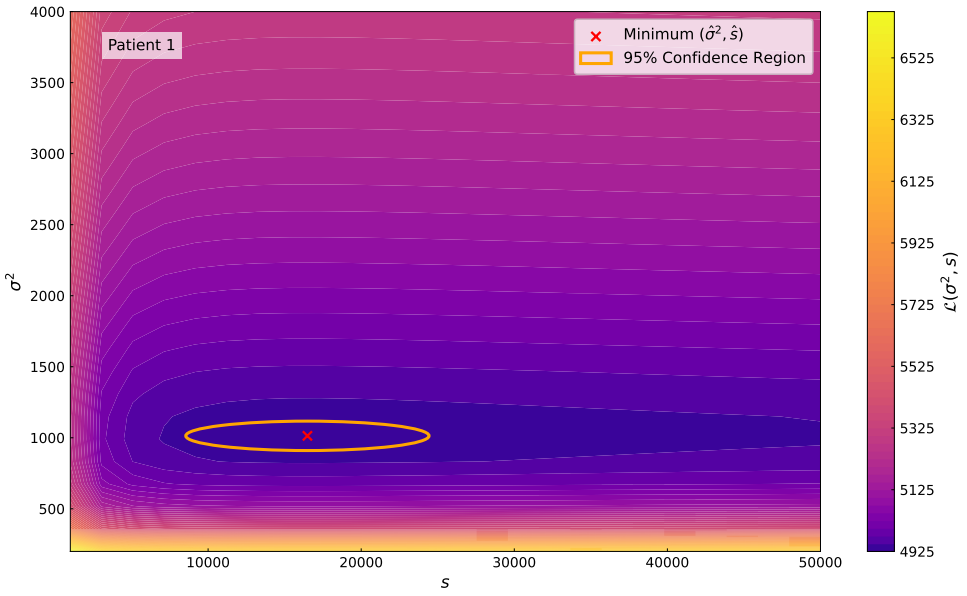


Figure 2: Loss landscape of (9) for given s and σ^2. The loss is equivalent to the negative log-marginal likelihood (8) up to a constant. The red x marks the minimum determined using the BFGS algorithm. The ellipsoid corresponds to the 95% confidence region.

(σ^2, s) from the approximated normal distribution and analyzed the information variation (12). The observed variance in information loss was consistently low, suggesting that the method remains stable in the most likely region of (σ^2, s), see Figure 3.

The proposed method employs spatial correlation between slices to improve the labeling efficiency, which is achieved by incorporating the 1D Laplace discretization T into the prior. To evaluate this approach, we compared the model to one that disregards spatial correlation, treating slices independently, corresponding to setting $T = I_L$ as the identity matrix. We call this independent model the *baseline model*. Performance is measured using the *information loss*, defined as the discrepancy in the negative log-likelihood of the ground-truth data between the posterior with full data and that with only partially observed data. Let index set $J \subset 1, ..., NL$ be the observed data and μ_{Y_L}, Σ_{Y_L} the mean and covariance of the posterior (6) given this data. If μ_Y, Σ_Y are the mean and variance of the posterior with all data, $J = 1, ..., NL$, then the information loss is defined as

$$D(J, Y) = \log(\mathcal{N}(Y|\mu_Y, \Sigma_Y)) - \log(\mathcal{N}(Y|\mu_{Y_L}, \Sigma_{Y_L})). \tag{12}$$

For better comparability, we used the same hyperparameters, s and σ^2, for both models. The difference in negative log-likelihood for the complete data is negligible for the spatial and baseline models. The difference is smaller than 0.02, a relative difference of $3 \cdot 10^{-6}$. This result suggests that when data are numerous, the influence of the different prior is negligible because the likelihood is dominant.

Two scenarios were tested to assess the performance of the proposed approach. Using neural networks for segmentation can produce reliable results; however, it could fail for single slices [11, 12], requiring experts to label these slices manually. To replicate this behavior in the first scenario, we applied jackknife resampling, omitting one slice while assuming the remaining

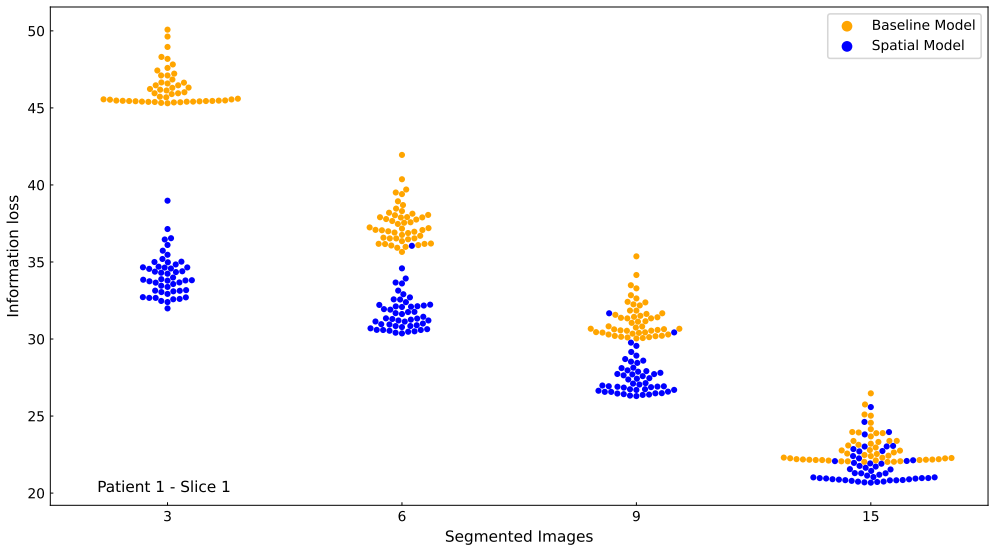


Figure 3: Information loss between the model trained on all data and each of the baseline and spatial models, evaluated for 50 sampled pairs (s, σ^2) from the normal approximation of the marginal log-likelihood around $(\hat{\sigma}^2, \hat{s})$. Both models were evaluated for varying numbers of segmented images for Patient 1 and Slice 1. The sample variance is low, indicating the stability of the proposed modeling approach with respect to σ^2 and s.

slices were fully segmented. Then, we measured the information loss on the left-out slice and iteratively added newly segmented images. The segmentation order was determined by greedily selecting indices that minimized the trace of the posterior covariance, following [11]. Figure 4 illustrates that the spatial model consistently outperforms the baseline across all slices. The spatial model retains more information than the baseline when most measurements from a slice are omitted and quickly recovers information from just a few labeled examples. This result suggests that the built-in spatial prior effectively transfers information from neighboring slices. Notably, this effect weakens after around 15 segmented images, as the likelihood (identical for both models) begins to dominate the prior.

To further illustrate our method, we present the posterior distribution for both patients using three and seven measurements to illustrate the method. In Figure 5, the advantage of the proposed approach is evident. Even with only three measurements, the posterior qualitatively aligns with the ground truth, whereas the baseline model fails to capture it. This result demonstrates how the spatial model effectively uses information from neighboring slices. This trend is consistent across all slices, with full results in the appendix in Figure 7.

In the second scenario, where this modeling approach is beneficial, no images are labeled, and a practitioner must label these images individually. A reliable prediction with as little data as possible is beneficial. We started with no labeled images for this scenario and randomly measured up to 42. Thus , in the end, 42 of the 1,200 images of the full dataset were labeled. We employed the same random seeds for the baseline and spatial models to maintain comparability. The procedure was repeated 300 times (Figure 6).

Incorporating the spatial correlation into the model is beneficial in this comparison, especially for a few labeled images. Each measurement point can influence the posterior in all

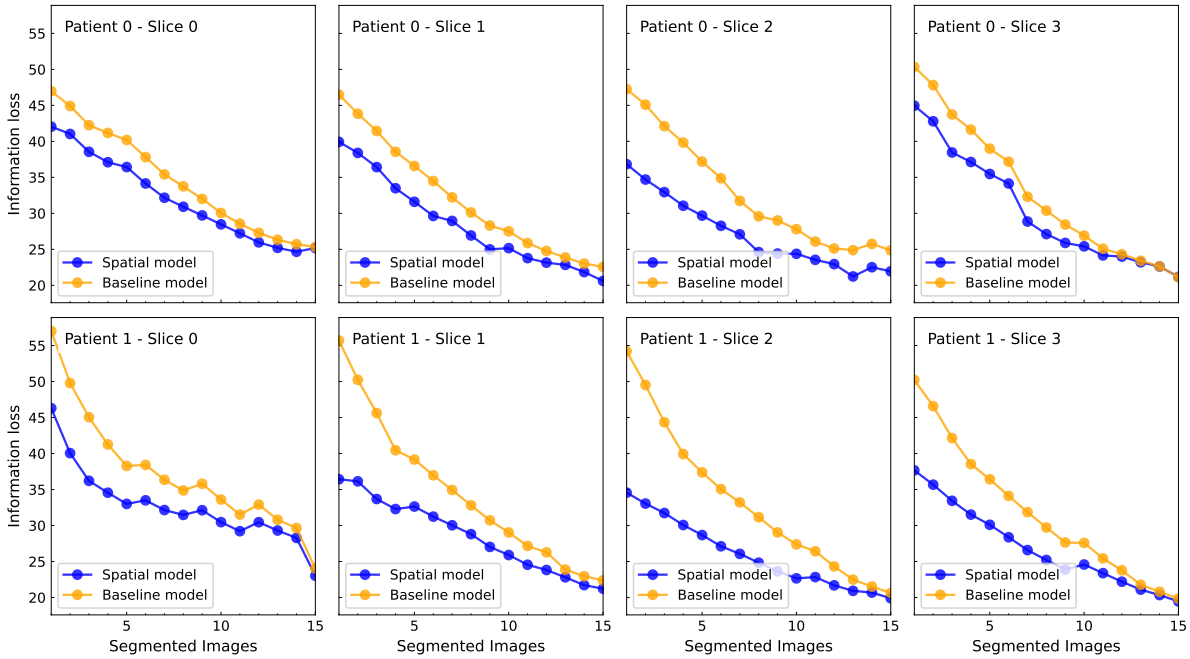


Figure 4: Information loss comparison of the baseline and spatial models on all slices and patients. In all plots, the corresponding slice is left out. Image-by-image measurements are included, up to 15 images per slice. The lower line for the spatial model indicates that it can transfer information from the other segmented slices.

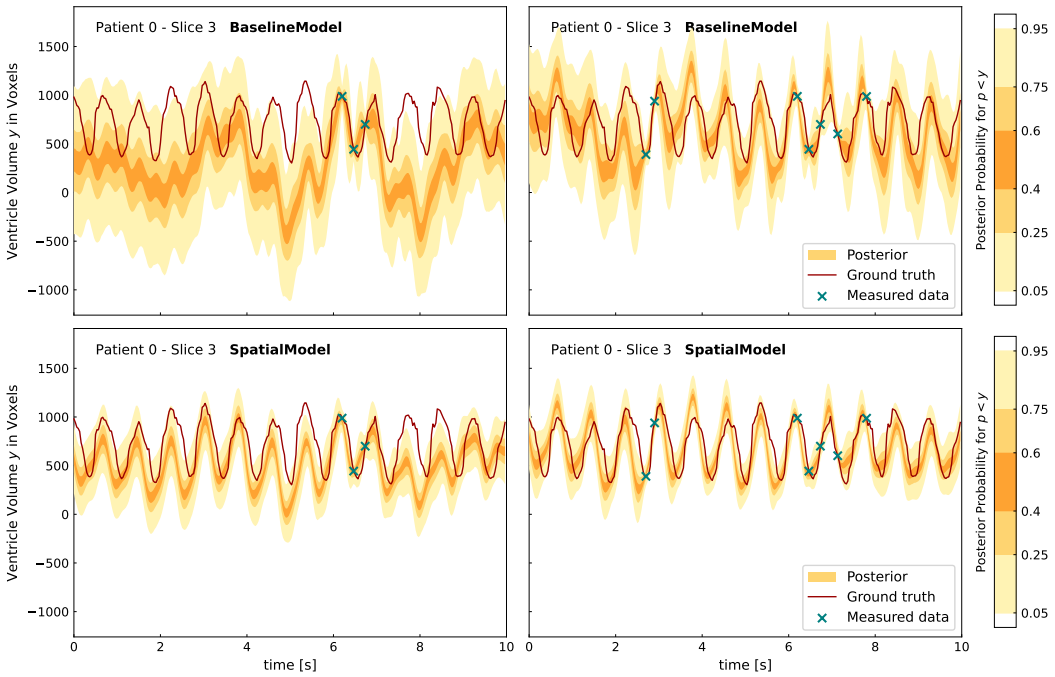
slices, which is impossible for the baseline model. Especially for the more regular heartbeat of Patient 1, the spatial model vastly outperforms the baseline model. Although the influence of spatial correlation in the prior diminishes as more data are incorporated, its effect remains observable even after 42 measurements for both patients.

4 CONCLUSION

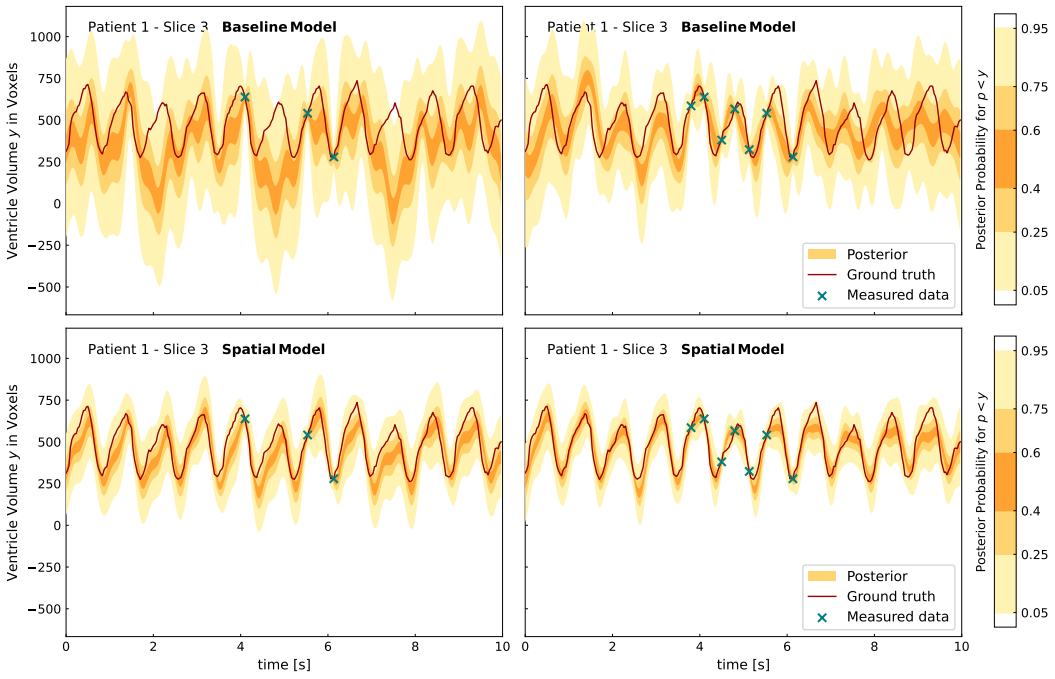
In cardiac real-time MRI, segmenting the ventricle is essential to obtaining ventricular volume information, which is used to derive critical clinical parameters. However, maximizing the information from each labeled image is crucial, given the high segmentation cost. We incorporated prior knowledge into the proposed model to achieve this maximization , representing the heartbeat over time as a superposition of frequencies. We assumed that the magnitudes of each frequency are correlated across neighboring slices and integrated this assumption into the prior using the discretized 1D Laplace operator to enforce smoothness.

A key advantage of our model is its efficiency. By applying conjugacy, we computed the posterior analytically, eliminating the need for sampling or variational inference. This outcome simplifies inference and allows the direct computation of the marginal likelihood gradient with respect to the hyperparameters. Thus, we can efficiently optimize the type-II likelihood using gradient-based methods, such as BFGS. Moreover, the proposed model is stable with respect to the choice of s and σ^2.

We evaluated the proposed approach in two scenarios: one with a missing slice but complete information from the remaining slices and one with no segmented images for any slice. In both cases, the proposed method outperformed the baseline, which analyzes each slice in-



(a) Patient 0



(b) Patient 1

Figure 5: Ventricle volume over time for Slice 3 for both patients. The red curve represents the ground truth from expert-validated segmentations; the yellow curve presents the posterior predictive based on (left) three and (right) seven measurements, assuming the remaining slices are fully segmented. The proposed model produces a posterior that closely aligns with the ground truth using information from neighboring slices. In contrast, the baseline model, which treats each slice independently, fails to capture this structure.

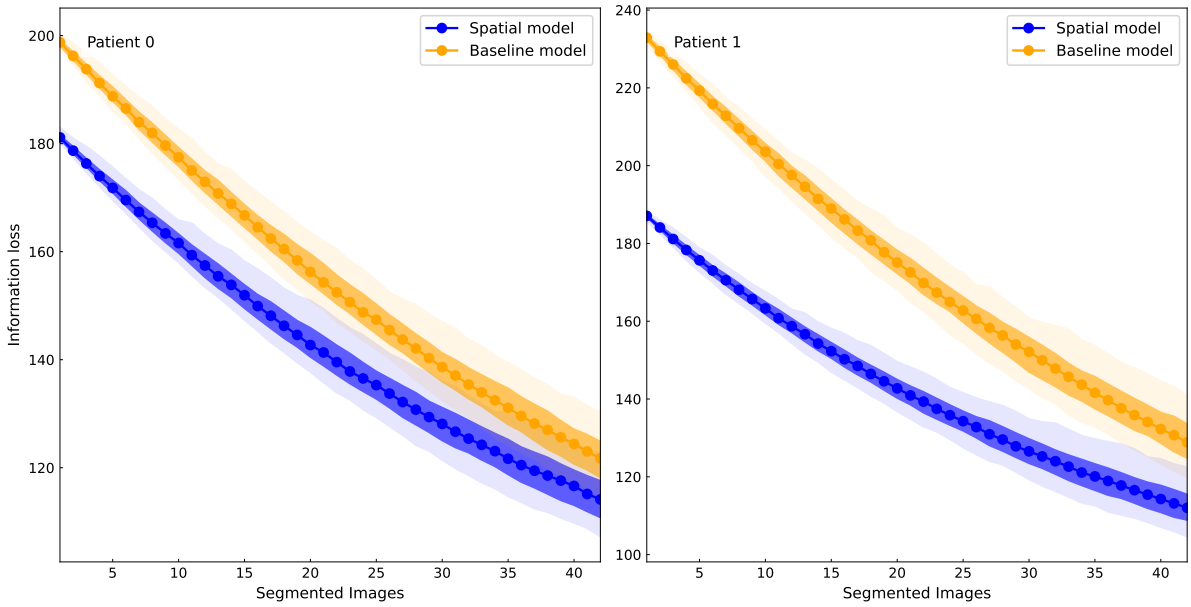


Figure 6: Information loss comparison between the baseline and spatial models for both patients. Up to 42 images were randomly selected and segmented throughout the four slices. The 300 identical random sequences were drawn for each patient and method. The solid line represents the median, whereas the shaded area represents the inner 50% and inner 90%. The lower line for the spatial model indicates that it can transfer information from other segmented slices.

dependently. The proposed method performed exceptionally well in data-scarce environments, which is critical in real-time MRI, where the high data volume limits manual segmentation, and complex cardiac geometries pose challenges for automatic methods. Furthermore, the results demonstrate that the proposed model produces a posterior that qualitatively aligns with the ground truth even with as few as three labeled frames per slice. In contrast, the baseline model fails to capture the underlying dynamics.

Acknowledgments

This work was performed as part of the Helmholtz School for Data Science in Life, Earth and Energy (HDS-LEE) and received funding from the Helmholtz Association of German Research Centres. This publication is partly supported by the Alexander von Humboldt Foundation and the King Abdullah University of Science and Technology (KAUST) Office of Sponsored Research (OSR) under Award No. OSR-2019-CRG8-4033.

We would like to thank the whole Hypofon study team: Nicole Müller, Julian Alexander Härtel, Jan Schmitz, Ute Baur, Melanie von der Wiesche, Iris Rieger, Jon von Stritzky, Christopher Hart, Janina Bros, Benedikt Seeger, Emily Zollmann, Marijke Grau, Boris Dragutinovic, Laura-Maria de Boni, Jan-Niklas Hönemann, Wilhelm Bloch, Daniel Aeschbach, Eva-Maria Elmenhorst, Ulrike Herberg, Alena Hess, Moritz Schumann, Tobias Kratz, Jens Jordan and Johannes Breuer and the participants for the unique study data and Stiftung KinderHerz for making this study possible.

Appendix

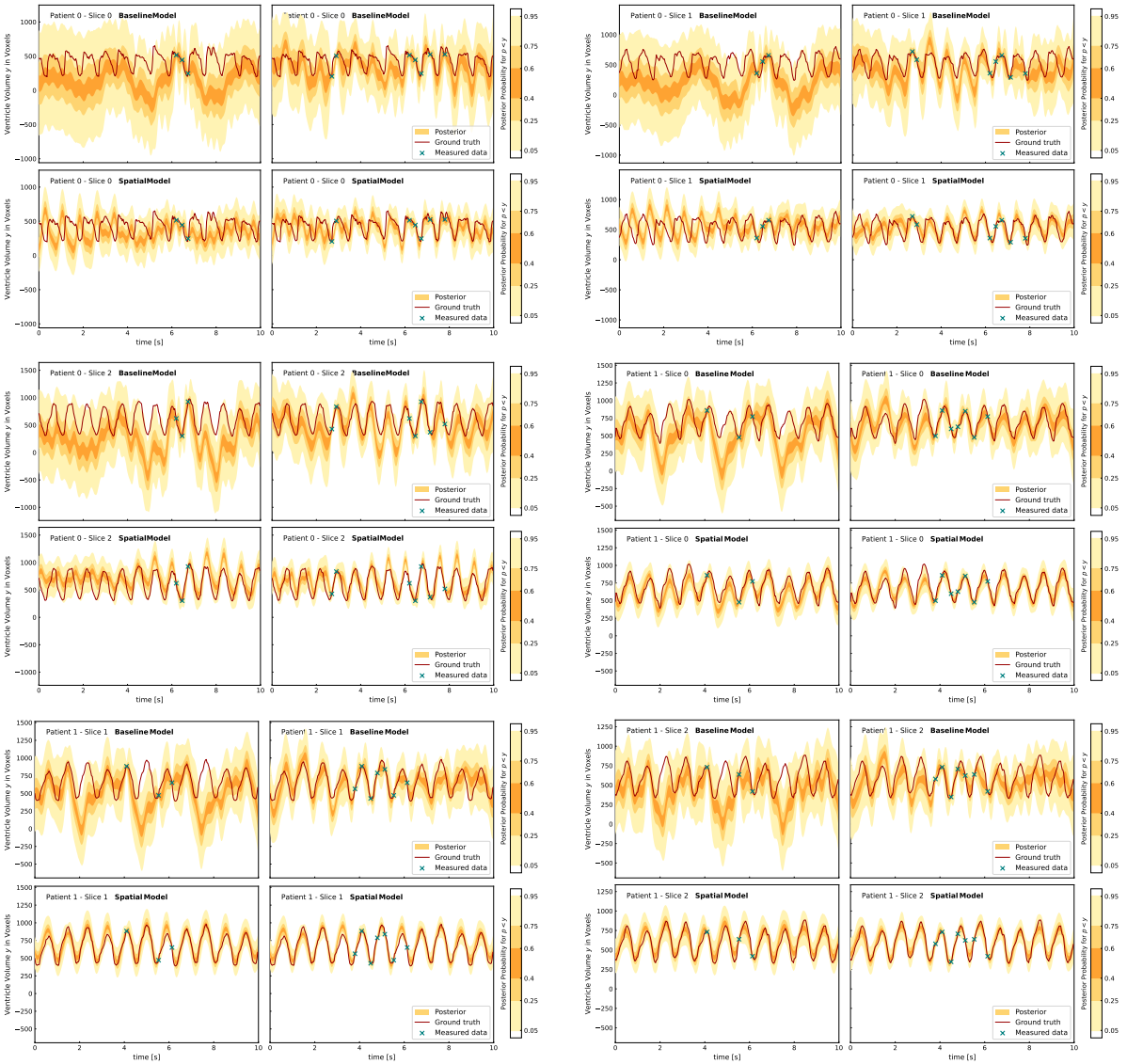


Figure 7: Ventricle volume over time for Slices 1, 2, and 3 in both patients. The red curve represents the ground truth from expert-validated segmentations; the yellow curve marks the posterior predictive, based on three and seven measurements, assuming the remaining slices are fully segmented. The proposed model produces a posterior that closely aligns with the ground truth using information from neighboring slices. In contrast, the baseline model, which treats each slice independently, fails to capture this structure.

REFERENCES

[1] C. G. Broyden. The convergence of a class of double-rank minimization algorithms 1. general considerations. *IMA Journal of Applied Mathematics*, 6(1):76–90, 03 1970.

[2] Bradley P Carlin and Thomas A Louis. *Bayes and Empirical Bayes Methods for Data Analysis*. Chapman and Hall, 1996.

[3] R. Fletcher. A new approach to variable metric algorithms. *The Computer Journal*, 13(3):317–322, 01 1970.

[4] Kay L. Gemba, Santosh Nannuru, Peter Gerstoft, and William S. Hodgkiss. Multi-frequency sparse Bayesian learning for robust matched field processing. *The Journal of the Acoustical Society of America*, 141(5):3411–3420, 05 2017.

[5] Peter Gerstoft, Christoph F. Mecklenbräuker, Angeliki Xenaki, and Santosh Nannuru. Multisnapshot Sparse Bayesian Learning for DOA. *IEEE Signal Processing Letters*, 23(10):1469–1473, 2016.

[6] Donald Goldfarb. A family of variable-metric methods derived by variational means. *Mathematics of computation*, 24(109):23–26, 1970.

[7] Kevin P. Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2023.

[8] Nicole Müller, Julian Alexander Härtel, Jan Schmitz, Ute Baur, Melanie von der Wiesche, Iris Rieger, Darius Gerlach, Jon von Stritzky, Anja Bach, Christopher Hart, Janina Bros, Benedikt Seeger, Emily Zollmann, Marijke Grau, Boris Dragutinovic, Laura-Maria de Boni, Jan-Niklas Hönemann, Wilhelm Bloch, Daniel Aeschbach, Eva-Maria Elmenhorst, Ulrike Herberg, Alena Hess, Moritz Schumann, Tobias Kratz, Jens Jordan, Johannes Breuer, and Jens Tank. Peripheral oxygenation and pulmonary hemodynamics in individuals with fontan circulation during 24-hour high-altitude exposure simulation. *Circulation*, 149(18):1466–1468, 2024.

[9] Michael Salerno, Behzad Sharif, Håkan Arheden, Andreas Kumar, Leon Axel, Debiao Li, and Stefan Neubauer. Recent advances in cardiovascular magnetic resonance. *Circulation: Cardiovascular Imaging*, 10(6):e003951, 2017.

[10] David F Shanno. Conditioning of quasi-newton methods for function minimization. *Mathematics of computation*, 24(111):647–656, 1970.

[11] Felix Terhag, Philipp Knechtges, Achim Basermann, and Raúl Tempone. Sparse bayesian learning for label efficiency in cardiac real-time MRI. 2025. (preprint).

[12] Felix Terhag, Philipp Knechtges, Achim Basermann, and Raúl Tempone. Uncertainty quantification in machine learning based segmentation: A post-hoc approach for left ventricle volume estimation in MRI. *SIAM/ASA Journal on Uncertainty Quantification*, 13(1):90–113, 2025.

[13] Shuo Zhang, Arun A. Joseph, Dirk Voit, Sebastian Schaetz, Klaus-Dietmar Merboldt, Christina Unterberg-Buchwald, Anja Hennemuth, Joachim Lotz, and Jens Frahm. Real-time magnetic resonance imaging of cardiac function and flow—recent progress. *Quantitative Imaging in Medicine and Surgery*, 4(5):313, October 2014.

[14] Shuo Zhang, Martin Uecker, Dirk Voit, Klaus-Dietmar Merboldt, and Jens Frahm. Real-time cardiovascular magnetic resonance at high temporal resolution: radial flash with non-linear inverse reconstruction. *Journal of Cardiovascular Magnetic Resonance*, 12(1):39, 2010.

[15] Zhilin Zhang and Bhaskar D. Rao. Sparse signal recovery with temporally correlated source vectors using sparse bayesian learning. *IEEE Journal of Selected Topics in Signal Processing*, 5(5):912–926, 2011.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

IMAGE-WISE UNCERTAINTY QUANTIFICATION FOR MEDICAL IMAGING

Philipp Rosauer[1], Angelina Händeler[1], Felix Terhag[1], Alexander Rüttgers[1], Michael Felderer[1,2]

[1]German Aerospace Center (DLR), Institute of Software Technology
Linder Hoehe, 51147, Cologne, Germany
e-mail: {philipp.rosauer, angelina.haendeler, felix.terhag, alexander.ruettgers, michael.felderer}@dlr.de

[2]University of Cologne, Department of Mathematics and Computer Science
Albertus-Magnus-Platz, 50923, Cologne, Germany

Abstract. *Cine cardiac magnetic resonance imaging (cineCMR) is a well-established imaging modality that is widely used in clinics and research centers to evaluate vital parameters like the exact stroke volume in the left and right ventricle or blood flow in the aortic root. To support medical professionals in the evaluation of cineCMR by machine learning, an uncertainty estimation is essential to replace trust and lack of it with probabilistic (interim) results. We present our workflow that ranks segmented images by image-wise uncertainty and supports manual enhancement by visualizing pixel-wise uncertainty, to improve the efficiency and effectiveness of human-machine collaboration. We present the performance of our nnU-Net configured U-Net with Monte Carlo (MC) dropout to predict uncertainty maps on the open Automated Cardiac Diagnosis Challenge (ACDC) dataset. Furthermore, we present multiple image-wise uncertainty scores derived from these pixel-wise uncertainty maps and compare their performance. Compared to only pixel-wise uncertainty-based workflows, we show how image-wise uncertainty scores can assist in the correction and validation of predicted segmentations through image evaluation. Due to an overwhelming amount of recorded CMR data, the proposed method could enable more reliable and faster evaluation in collaboration between medical professionals and probabilistic models.*

Keywords: Medical Image Segmentation, Cardiac Magnetic Resonance Imaging, Entropy, Monte Carlo Dropout

1 INTRODUCTION

Semantic segmentation of CMR plays a crucial role in both diagnosis, monitoring and therapy of cardiovascular diseases. Leveraging deep learning models, like convolutional neural networks (CNNs), has achieved remarkable performance and shows the potential to drastically reduce analysis times [2]. These advances in fully automated segmentation are particularly needed for new imaging techniques like real-time MRI, producing thousands of images. However, despite scoring increasingly well in challenges, these models are inherently uncertain, which poses challenges in both research and clinical applications. Quantifying this uncertainty during application is essential to provide meaningful assistance to medical staff and ensuring reliable decision making.

Uncertainty Quantification (UQ) in segmentation is an active area of research, mostly aiming to indicate uncertain regions in one segmentation. While pixel-wise uncertainty measures, such as entropy, provide insights into per-voxel confidence, clinicians must review every image, as they do not directly indicate whether the specific segmentation should be reviewed. Especially when working with hundreds and thousands of individual segmentations on a daily basis, it seems a more efficient approach to somehow aggregate uncertainty information at the image level, allowing for a ranking of segmentations by their reliability [4].

There is a lack of image-wise metrics and their practical use to guide human interaction with model generated segmentations. Similar to the work of Ng et al. [10], in this work, we analyze the approach of using an uncertainty measure for ordering segmentations based on their aggregated pixelwise uncertainty. As the application of volumetric prediction is less concerned with exact geometrical accuracy of class borders, but rather focuses on high overlap, instead of distance-based metrics we use the Dice similarity coefficient (DSC) to measure how much prediction and ground truth coincide. This score can not only assist clinicians and researchers to identify cases that may require human validation or correction, but also serve as a tool to evaluate model performance on specific data distributions. Combining both image-wise and pixel-wise uncertainty measures could help to reduce the validation and correction effort of medical staff, while increasing understanding and assessability.

To validate the approach, we use a publicly available dataset, as well as state-of-the-art open-source U-Net model configurations. We demonstrate high correlation of our implementation in terms of ordering segmentations by performance, as measured by DSC between predictions and ground truth.

2 METHODS

In the following, we present our methodology, detailing the dataset, the segmentation model, and the uncertainty measures at both the pixel and image level.

2.1 Data

The presented methods are tested using the Automated Cardiac Diagnosis Challenge (ACDC) dataset [1]. The ACDC dataset is a widely recognized benchmark dataset for cardiac image analysis. It provides annotated CMR images across a range of pathologies, allowing robust evaluation of segmentation algorithms. The full dataset with 150 patients is divided into 100 patients, with 20 patients for each of the 5 groups that can be used for training, while the test dataset consists of the remaining 50 patients, with 10 for each of the 5 groups. This results in a total of 1902 training and 1076 test images. In all images the left ventricle (LV), the myocardium (MYO) and the right ventricle (RV) are labeled by experts. The dataset contains

images from different slices and physiological timepoints. This results in wide distribution of sizes of all three classes.

2.2 Model Architecture & Training

For CMR segmentation, we employ a U-Net architecture, a widely used convolutional neural CNN designed for biomedical image segmentation [12]. Our implementation follows the parameterization of nnU-Net [5], ensuring a well-optimized configuration for medical imaging tasks.

The network consists of a five-layer encoder-decoder architecture, where each stage in the contraction and expansion paths contains a double convolutional block. Each block consists of two convolutional layers with kernel size 3, stride 1, and no padding, followed by instance normalization and Leaky ReLU activation. Dropout layers are incorporated into the convolutional blocks in the inner half of the network to estimate uncertainty using Bayesian principles [6]. Unlike standard segmentation pipelines, softmax activation is omitted, as uncertainty estimation requires access to unnormalized logits.

Training is performed using stochastic gradient descent (SGD) with a learning rate of 0.01, momentum of 0.99, and weight decay of 0.001 applied to a regularized cross-entropy loss function. The dropout rate is set to 0.2. The input images are resized to 256 × 224 pixels, and training images are randomly divided into 32 batches, each containing 58 samples.

2.3 Segmentation quality

To assess the quality of the segmentation results, we use two commonly employed and recommended metrics: the Average Symmetric Surface Distance (ASSD), representing distance-based metrics and the DSC, representing overlap-based metrics. However, segmentation quality is difficult to measure in general because it lacks a single definitive metric, varies across anatomical structures and clinical contexts, and is influenced by factors such as image resolution, label uncertainty, and interobserver variability [7].

ASSD ASSD measures the average distance between the surfaces of two segmentations. It is computed by calculating the distance from each point on the surface of one segmentation to the nearest point on the surface of the other, and then averaging these distances symmetrically:

$$\mathrm{ASSD}(A, B) = \frac{\sum_{x \in \partial S_A} d(x, \partial S_B) + \sum_{y \in \partial S_B} d(y, \partial S_A)}{|\partial S_A| + |\partial S_B|} \tag{1}$$

where:

- ∂S_A and ∂S_B are the sets of surface points of segmentations A and B, respectively.
- $d(x, \partial S_B) = \min_{y \in \partial S_B} \|x - y\|$ is the shortest Euclidean distance from point x in ∂S_A to the closest point in ∂S_B.
- Similarly, $d(y, \partial S_A) = \min_{x \in \partial S_A} \|y - x\|$.

A drawback of this metric is how it handles the case of complete segmentation failure. In this case, where a class is present only in one of the two compared segmentations, the denominator is 0. We chose to penalize these cases by setting the ASSD to the highest possible distance in the predicted image, its diagonal, as recommended in [7].

DSC DSC is a measure of overlap between the predicted segmentation and the ground truth. It ranges from 0 (no overlap) to 1 (perfect overlap), and is defined as:

$$\mathrm{DSC}(A, B) = \frac{2|A \cap B|}{|A| + |B|}, \tag{2}$$

where $|A|$ and $|B|$ denote the number of elements (e.g., pixels or voxels) in segmentations A and B, and $|A \cap B|$ represents the number of overlapping elements.

2.4 Pixel-wise Uncertainty Quantification

There are various uncertainty measures that can be applied to image segmentation. Here, we compare two methods that provide pixel-wise uncertainty.

Entropy The first is entropy, which is defined as follows [11]:

$$H(x) = -\sum_{i=1}^{C} p_i \log p_i, \tag{3}$$

where C is the number of possible classes to which pixel x can belong, and p_i is the predicted probability that the pixel belongs to the i-th class. As p_i is simply given as an interim result in $\mathrm{softmax}$, the additional cost to compute H_i is minimal.

MC Dropout Variance (MCDV) The second approach we present is to use MC dropout during inference. For this, the dropout layers in the model remain active during the inference phase to obtain probabilistic predictions. The dropout probability is set to $p = 0.2$, the same as during training. For each input image, the model is evaluated five times to obtain five probabilistic, slightly different predictions. This is equivalent to sampling the model weights five times from a Bernoulli distribution with probability p, and making predictions depending on the resulting, slightly different weights. According to the theoretical derivation by Gal et al. [3], the predictive distribution, which is also referred to as the Bayesian model average, is then approximated by taking the sample mean of these probabilistic predictions p_i, $i = 1, \ldots, 5$. It is calculated for each segment class c, $c = 1, \ldots, 4$, and each pixel x separately:

$$\hat{p}^c(x) = \frac{1}{5} \sum_{i=1}^{5} p_i^c(x) \tag{4}$$

The pixelwise uncertainty maps result from this by calculating the pixelwise entropies of the mean outputs $\hat{p}^c$ over the segment classes, $c = 1, \ldots, 4$, as defined in (3):

$$u(x) = -\sum_{c=1}^{4} \hat{p}^c(x) \log \hat{p}^c(x).$$

2.5 Image-wise Uncertainty Quantification

Mean To derive an image-wise uncertainty measure from the pixel-wise measure, we can simply compute the mean:

$$\bar{H} = \frac{1}{N} \sum_{i=1}^{N} H(x) \tag{5}$$
$$\bar{u} = \frac{1}{N} \sum_{i=1}^{N} u(x)$$

Surface Distance over Samples Using the pairwise ASSD between samples, we can define:

$$\text{ASSD}_\text{S} = \frac{2}{N(N-1)} \sum_{0 \leq i < j < N} \text{ASSD}(S_i, S_j), \tag{6}$$

where S_i and S_j are pairwise predicted labels of a probabilistic sample each, derived from the softmax outputs by assigning the segment class with the highest probability to a pixel and N the total number of Samples.

DSC over Samples To measure similarity between the softmax probabilities of the predictions we use the pairwise DSC between samples:

$$\text{DSC}_\text{S} = \frac{2}{N(N-1)} \sum_{0 \leq i < j < N} \text{DSC}(S_i, S_j), \tag{7}$$

similarly to ASSD_S.

2.6 Test setup

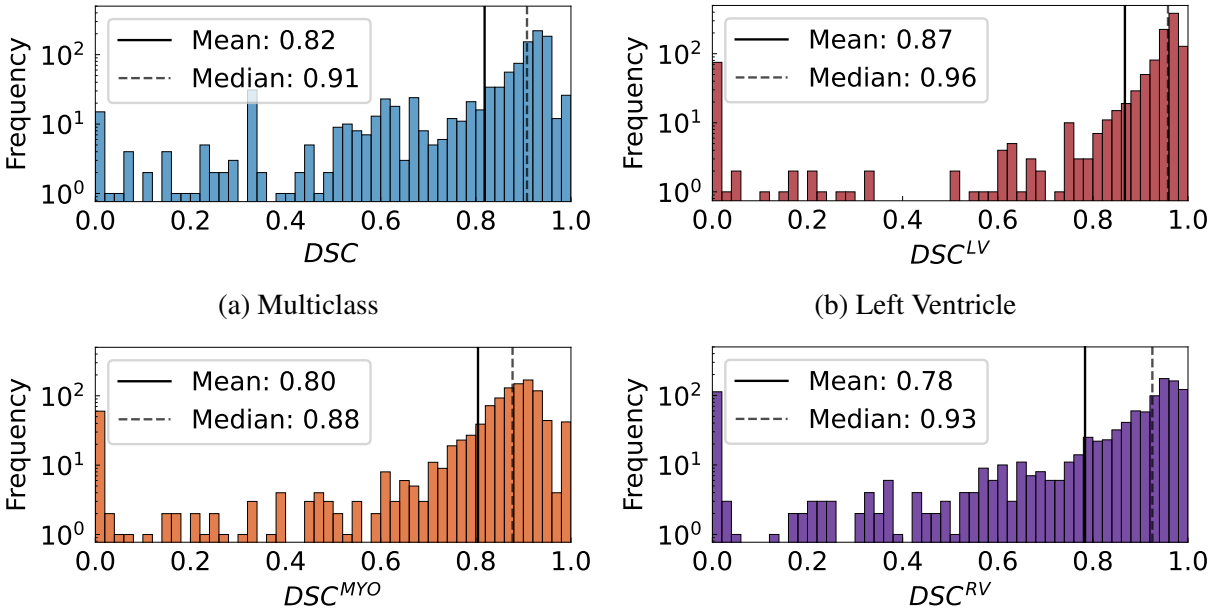
In this work, our primary focus is not on achieving the highest segmentation accuracy but rather on assessing the ability to correctly rank images by their segmentation performance. This ranking is evaluated by comparing the uncertainty quantification with the segmentation quality. To quantify the correlation between uncertainty and segmentation quality, we employ Spearman's rank correlation coefficient (ρ). Spearman's ρ is a non-parametric measure that evaluates the strength and direction of the monotonic relationship between two variables, making it well-suited for analyzing how uncertainty reflects segmentation performance. According to [13], this is defined as:

$$r_S(X, Y) = 1 - \frac{6 \sum_{i=1}^{n} (P_i - Q_i)^2}{n(n^2 - 1)},$$

where $\{(X_i, Y_i)\}_{i=1}^{n}$ are n iid data pairs, P_i the rank of X_i among the $X_1, \ldots, X_n$ and Q_i the rank of Y_i among the $Y_1, \ldots, Y_n$. In this application, X_i represent the image-wise uncertainties and Y_i the DSC of n predictions. It takes values in $[-1, 1]$, where 1 is perfect, -1 is negative, and 0 is no correlation. A strong correlation indicates that a uncertainty quantification effectively identifies cases of higher and lower segmentation performance.

3 RESULTS

In this section, we analyze the relationship between uncertainty measures and segmentation quality. We first evaluate pixel-wise uncertainty using established methods such as entropy and MC dropout. Additionally, we compute four image-wise uncertainty measures: $\bar{H}$, $\bar{u}$ (5), ASSD_S (6), and DSC_S (7). To assess their effectiveness, we compare the Spearman correlations of these metrics with segmentation performance, providing insights into how uncertainty rankings align with actual segmentation performance. The U-Net model achieved a mean DSC of 0.82 and a median DSC of 0.91 on the test dataset.



(a) Multiclass

(b) Left Ventricle

(c) Myocardium

(d) Right Ventricle

Figure 1: Histograms showing the distribution of binned $Dice$ scores for each segmentation class and all classes combined.

To analyze the segmentation performance across the dataset, we visualize the distribution of DSC using a histogram, see Figure 1. The x-axis represents the DSC values, while the y-axis indicates the frequency of occurrences on a logarithmic scale.

The histograms of the three classes show a bimodal distribution with a high frequency of samples with DSC close to 1, a very low frequency of samples in the mid-range (0.1–0.85) and a notable peak at 0, which occurs when either the prediction or the ground truth is completely missing while the other is present. The mean and median DSC scores for the LV class is highest, with a notably smaller amount of predictions with a score under 0.9. These distributions highlight that while the majority of segmentations are accurate, there are cases of inaccurate segmentations. The range of these inaccuracies spans from complete segmentation failure ($\text{DSC} = 0$) to small deviations only at the edges of classes, which may correspond to difficult or ambiguous samples, underlining the importance of identifying uncertain cases and ranking them for potential human validation.

3.1 Pixel-wise Uncertainty

We computed MCDV and entropy on a pixel basis and provide the five probabilistic inference outputs (Sample 1 - Sample 5), as well as the given GT for three examples in Figure 2. All three images are cropped to fit the ground truth segmentation for visualization, thus the resolution differs between them. Image ★ presents a typical mid-ventricular slice with a high DSC of 0.97. The entropy H_x is only raised at the very borders of the classes, which suggests high agreement between classes within a single prediction. Also, the predictions of all five samples is close to identical, represented in a very high DSC_S of 0.99. As for H_x, u_x is only raised at the borders. In case of Image ■ the DSC is significantly lower, mainly caused by errors in the RV class, but also by disagreement regarding the LV-MYO border. Mislabeled pixels do indeed show an increase in H_x, indicating lower agreement in areas that are predicted incorrectly. However, the five sampled predictions show only slight differences, lowering the DSC_S only by a margin compared to Image ★. Lastly, Image ♦ is an example for segmentation failure with a DSC of only 0.28. The overlap between the five sampled predictions ($\text{DSC}_\text{S} = 0.59$) is very low. Although H_x and also u_x visually indicate high local disagreement, computing the mean value over the entire image $\bar{H}$ and $\bar{u}$ fail to represent this information. This is likely due to the significantly smaller segmentation, compared e.g. to Image ★.

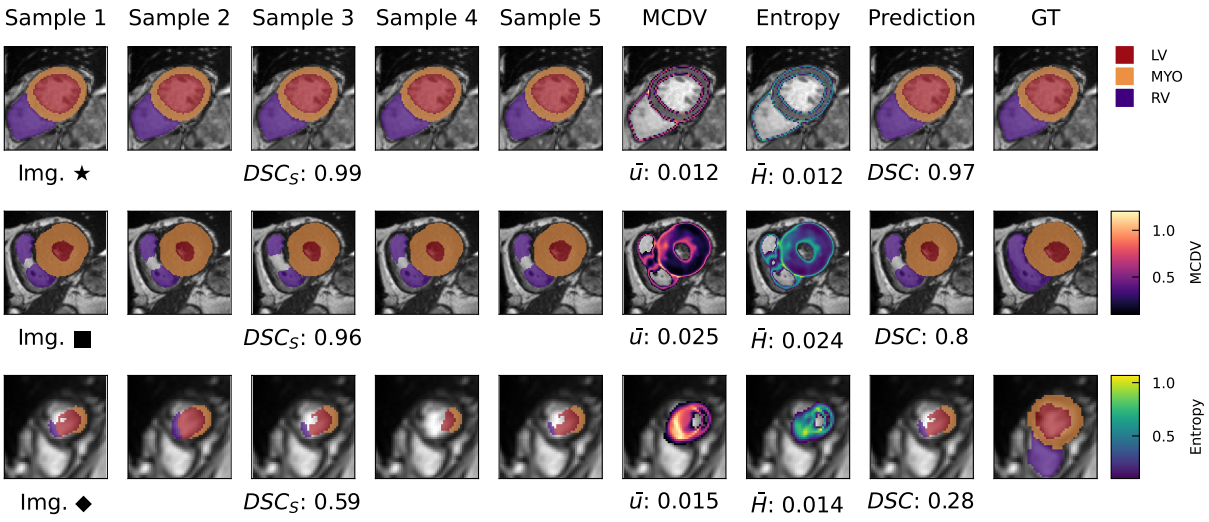


Figure 2: 3 Images from the test dataset with all 5 predictions from the probabilistic dropout inference, MC Dropout Variance (MCDV), deterministic prediction and ground truth (GT)

3.2 Correlation of image-wise uncertainty measures

We investigated the potential of all 4 uncertainty measures for evaluating segmentation performance during or after inference. To measure segmentation performance of the trained model against ground truth on the test data ,we computed both ASSD as well as DSC. As stated in [10] both metrics are well established in medical image analysis.

Table 1 shows the Spearman correlations between uncertainty measures and segmentation performance metrics for all 3 segments. We found little to no correlation between both entropy and MCDV for all classes with both quality metrics. As evident in the examples in Figure 2 this probably caused by sensitivity of the metric in regard of relative segmentation size. The correlations between ASSD_S and ASSD for all classes range between 0.61 for the LV and 0.72

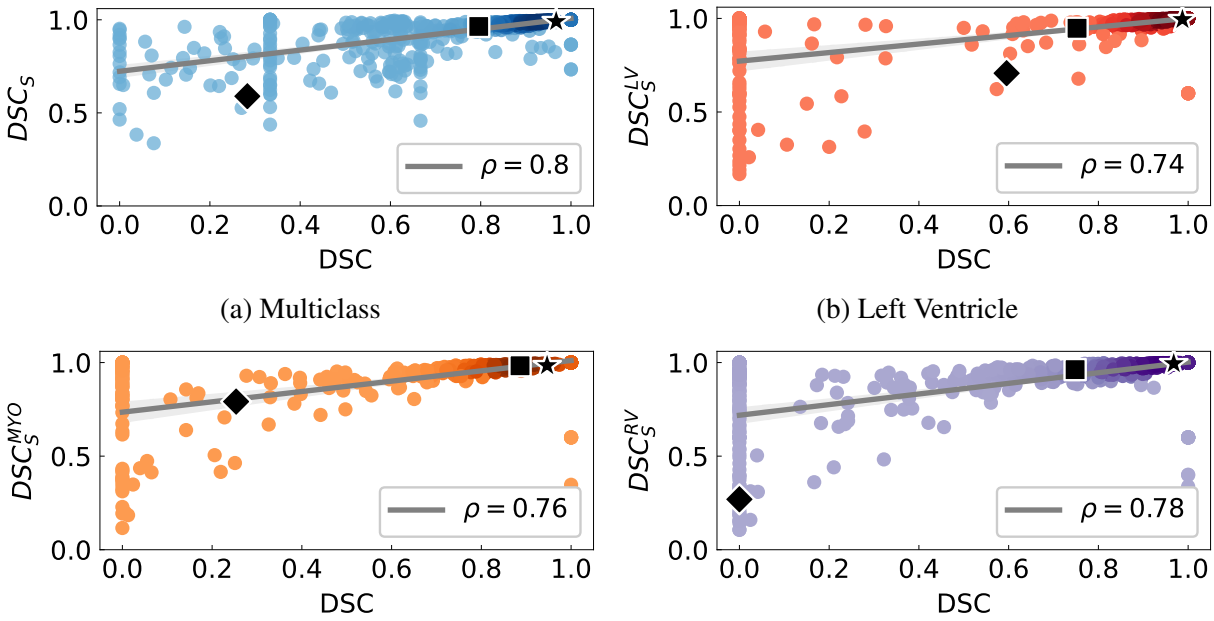
for MYO and are consistent with the findings of Ng et al. [10] and support the feasibility of using the measure for ranking predictions when a distance-based metric is needed. The correlation between $\mathrm{DSC_S}$ and DSC is even higher for all segments with lowest correlation for the class LV at 0.74 and highest correlation for RV at 0.78. These findings suggest that DSC_S is an even better predictor for segmentation quality, especially when overlap-based metrics are considered for segmentation quality.

Quality Metric	Left Ventricle				Myocardium				Right Ventricle			
	$\mathrm{DSC_S}$	$\mathrm{ASSD_S}$	$\bar{H}$	$\bar{u}$	$\mathrm{DSC_S}$	$\mathrm{ASSD_S}$	$\bar{H}$	$\bar{u}$	$\mathrm{DSC_S}$	$\mathrm{ASSD_S}$	$\bar{H}$	$\bar{u}$
DSC	**0.74**	0.57	0.12	0.22	**0.76**	0.43	0.01	0.01	**0.78**	0.64	0.12	0.17
$ASSD$	0.58	**0.61**	0.05	0.08	0.51	**0.72**	0.03	0.05	0.68	**0.69**	0.34	0.39

Table 1: Correlations between selected quality and uncertainty metrics

The plots in Figure 1 show the distributions of the computed DSC and DSC_S for every image in the test dataset for each segment as well as the combined score and measure for the multiclass segmentation (blue). It can be observed that for complete segmentation failure the uncertainty measure fails as an indicator for segmentation quality, as we measured a wide range of DSC_S for these cases, but show consistent results for the majority of data points.

Regarding computational cost, in our experiments DSC was nearly twice as fast as our implementation of ASSD, using SimpleITK for finding segmentation contours [9] and signed Maurer distance [8]. When used as a segmentation quality predictor, computation times should be considered, as they scale linearly with the number of compared samples.



(a) Multiclass

(b) Left Ventricle

(c) Myocardium

(d) Right Ventricle

Figure 3: Correlation plots showing the relationship between DSC. The x-axis represents the DSC between predictions and ground truth, while the y-axis represents the DSC between samples. The colormap in each subplot indicates point density estimated using Gaussian kernel density estimation (KDE), and a regression line illustrates the trend.

4 CONCLUSION

We presented our findings on correlations between chosen uncertainty measures and segmentation quality metrics. In particular, when the application requires a high volumetric accuracy, e.g. when determining the stroke volume, the inter-sample DSC offers high potential for ranking segmentations by predicted quality at a lower computational cost than other metrics comparing segmentation samples. Simply computing the entropy of a deterministic model or even computing MC dropout average showed little to no correlation with volumetric as well as distance based quality metrics.

As reasoned in [10], the physical meaning of distance metrices such as ASSD can provide useful insights depending on the medical application. However, we found that dealing with missing classes in one of the two compared segmentations leads to ambiguous results, as the metric itself, as stated in subsection 2.3, is not defined in these cases. Aggregating ASSD scores across segments and/or multiple samples introduces potential errors and increases ambiguity. We did not explore further variations of the ASSD that could counteract these shortcomings, but recognize the medical value a distance metric can offer.

If computation time is either not an issue, or is dominated by inference itself, we would consider using both ASSD, or another distance-based metric like the mean average surface distance (MASD), and DSC, following the recommendations in [7]. Furthermore, we do not consider the available results as merely academic, but plan to integrate them into an evaluation tool as a combination of image-wise and pixel-wise UQ measures. The calculated DSC sorts the segmented images and MCDV highlights areas that require special attention.

Acknowledgments The research was partly carried out under the project LBNP ISS4Mars by the German Aerospace Center (DLR).

REFERENCES

[1] Olivier Bernard, Alain Lalande, Clement Zotti, Frederick Cervenansky, Xin Yang, Pheng-Ann Heng, Irem Cetin, Karim Lekadir, Oscar Camara, Miguel Angel Gonzalez Ballester, et al. Deep learning techniques for automatic MRI cardiac multi-structures segmentation and diagnosis: is the problem solved? *IEEE transactions on medical imaging*, 37(11):2514–2525, 2018.

[2] Chen Chen, Chen Qin, Huaqi Qiu, Giacomo Tarroni, Jinming Duan, Wenjia Bai, and Daniel Rueckert. Deep learning for cardiac image segmentation: a review. *Frontiers in cardiovascular medicine*, 7:25, 2020.

[3] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International conference on machine learning*, pages 1050–1059. PMLR, 2016.

[4] Francesco Galati, Sébastien Ourselin, and Maria A Zuluaga. From accuracy to reliability and robustness in cardiac magnetic resonance image segmentation: a review. *Applied Sciences*, 12(8):3936, 2022.

[5] Fabian Isensee, Paul Jaeger, Simon Kohl, Jens Petersen, and Klaus Maier-Hein. nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods*, 18(2):203–211, 2021.

[6] Alex Kendall, Vijay Badrinarayanan, and Roberto Cipolla. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. *arXiv preprint arXiv:1511.02680*, 2015.

[7] Lena Maier-Hein, Annika Reinke, Patrick Godau, Minu Tizabi, Florian Buettner, Evangelia Christodoulou, Ben Glocker, Fabian Isensee, Jens Kleesiek, Michal Kozubek, et al. Metrics reloaded: recommendations for image analysis validation. *Nature Methods*, 21(2):195–212, 2024.

[8] Calvin Maurer, Rensheng Qi, and Vijay Raghavan. A linear time algorithm for computing exact Euclidean distance transforms of binary images in arbitrary dimensions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(2):265–270, 2003.

[9] Matthew McCormick, Xiaoxiao Liu, Julien Jomier, Charles Marion, and Luis Ibanez. Itk: enabling reproducible research and open science. *Frontiers in Neuroinformatics*, 8:13, 2014.

[10] Matthew Ng, Fumin Guo, Labonny Biswas, Steffen Petersen, Stefan Piechnik, Stefan Neubauer, and Graham Wright. Estimating uncertainty in neural networks for cardiac MRI segmentation: a benchmark study. *IEEE Transactions on Biomedical Engineering*, 70(6):1955–1966, 2022.

[11] Nikhil Pal and Sankar Pal. Entropy: A new definition and its applications. *IEEE transactions on systems, man, and cybernetics*, 21(5):1260–1270, 1991.

[12] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

[13] Weichao Xu, Yunhe Hou, YS Hung, and Yuexian Zou. A comparative analysis of Spearman's rho and Kendall's tau in normal and contaminated normal models. *Signal Processing*, 93(1):261–276, 2013.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

COUPLED CANN-DEM SIMULATION FOR HYPERELASTIC SOLIDS

Stefan Hildebrand[1], **Jonathan Georg Friedrich**[1], **and Sandra Klinge**[1]

[1] Department of Structural Mechanics and Analysis, Technische Universität Berlin
Straße des 17. Juni 135, 10623 Berlin, Germany
e-mail: {stefan.hildebrand, sandra.klinge}@tu-berlin.de

Abstract. *A general, unified neural network approach as replacement for the finite element method without the need for analytic expressions for material laws is suggested. The complete simulation process from the material characterization to simulations on a structural level takes place in the new neural network framework. The drawback of many conventional analytic expressions of material laws to require large numbers of experiments for parametrization is addressed by an integrated inverse approach. Specifically, an adaptation of the Deep Energy Method (DEM) is combined with a Constitutive Artificial Neural Network (CANN) and trained on measured displacement fields and prescribed boundary conditions in a coupled procedure. Tests on compressible and incompressible Neo-Hookean solids with up to twelve CANN parameters show high accuracy of the approach and very good generalization of CANNs. A small extent of data is required for robust and reliable training.*

Keywords: Neural Networks, NN, solid mechanics, Constitutive Artificial Neural Networks, CANN, Deep Energy Method, DEM, material modeling, hyperelasticity

1 INTRODUCTION

In computational engineering, machine learning (ML) approaches are gaining increasing attention as alternatives and extensions to traditional simulation techniques [1, 2, 3]. Data-driven methods are successfully used in material modeling [4, 5, 6] and characterization [7, 8], whereas a growing variety of physics-informed methods emerges for the solution of the static equilibrium condition [9, 10, 11]. The present contribution describes the general, unified neural network approach CANN-DEM [12] for elastostatic simulations. Analytic expressions for the material model [13, 14, 15] are replaced by an Artificial Neural Network (ANN) and an integrated procedure for training based on a small extent of data is suggested to automatize the material characterization. This results in a material-specific neural solver ready to be used in simulations at the structural level.

The current contribution is structured as follows: Section 2 (Methods and Materials) explains the Neo-Hookean models used in the case studies, as well as the basics of neural methods (CANN and DEM). The proposed setup is outlined in Section 3 (Integration of DEM and CANN). The validation of the proposed setup is shown in Section 4 (Case Study), which is followed by a discussion of advancements, limitations and challenges of implementation in Section 5 (Results). Finally, Section 6 summarizes conclusions and possible future directions of the research work.

2 METHODS AND MATERIALS

2.1 Constitutive modeling of hyperelastic materials

In the hyperelastic theory [16, 17, 18], the material behavior is prescribed by the strain energy density Ψ [19, 20] from which the stress tensor is determined by derivation with respect to the kinematic quantities. In the present contribution, a compressible Neo-Hookean solid is investigated, namely

$$\Psi(I_1, J) \quad = \tfrac{1}{2}\lambda(\log(J))^2 - \mu\log(J) + \tfrac{\mu}{2}(I_1 - 2), \quad \text{for } \mathbf{C} \in \mathbb{R}^{2\times 2} \quad , \tag{1}$$

with I_1 the first invariant of the right Cauchy-Green tensor $\mathbf{C} = \mathbf{F}^T\mathbf{F}$, $\mathbf{F}$ the deformation gradient and $J = \det \mathbf{F}$. λ and μ are Lamé material parameters that are determined experimentally.

2.2 Artificial Neural Networks (ANNs)

Artificial Neural Networks (ANNs) [21, 22] with parameter set Θ consisting of the weights w and biases b represent arbitrary mappings $\mathcal{R} : \mathbb{R}^n \mapsto \mathbb{R}^m$ [23] with n and m the input and output layer width, respectively. For the DEM, a Fully Connected Neural Network (FCNN) is applied, whereas the CANN requires a custom architecture. The training uses gradient descent based optimization algorithms like Adam. Data-driven loss contributions are computed by means of the Mean Square Error (MSE) function

$$\mathrm{MSE} = \frac{1}{N}\frac{1}{m}\sum_{k=1}^{N}\sum_{i=1}^{m}(R_i(P^k) - \tilde{R}_i(P^k))^2 \quad . \tag{2}$$

Here, P denotes the input to the ANN, R the target output and $\tilde{R}$ the actual ANN output of dimension m with N entries. In the following, quantities denoted with a tilde indicate ANN outputs or quantities that are calculated based on ANN outputs.

2.2.1 The Deep Energy Method (DEM)

The original PINN formulation [24] suggests a loss function based on the squared residual $\mathcal{N}$ of the PDE that is specified on the collocation points inside the body while the BCs are formulated as another, scaled contribution to the loss function, computed at the collocation points on the boundary [25, 9]. For the solution of coupled problems, several additional terms can be accounted for in the loss function [26]. This approach is adopted by the DEM, which has been introduced specifically for coupled boundary value problems including damage phase field modeling and piezoelectricity [27, 28]. Opposite to the standard PINN, the DEM only requires first derivatives of the displacement field $\tilde{\boldsymbol{u}}(\boldsymbol{X})$ to compute the total potential energy Π as the loss function, thus reducing the numerical complexity. The input of the ANN are points in the reference configuration $\boldsymbol{X} \in B \cup \Gamma$, in the domain B and on its boundary $\Gamma = \Gamma_D \cup \Gamma_N$ consisting of the Dirichlet boundary part Γ_D and the Neumann boundary part Γ_N.

A transformation combines the prescribed displacements on the boundary $\boldsymbol{u}_D(\boldsymbol{X}_D),\ \boldsymbol{X}_D \in \Gamma_D$ and the ANN output $\tilde{\boldsymbol{z}}(\boldsymbol{X})$ to calculate the displacement field $\tilde{\boldsymbol{u}}(\boldsymbol{X})$. To this end, a continuously differentiable mapping $\boldsymbol{A}(\boldsymbol{X})$ independent from the ANN parameters with $\boldsymbol{A}(\boldsymbol{X}_D) = \boldsymbol{0}$ is introduced. In a further step, the definition of $\boldsymbol{u}_D$ is extended by setting entries corresponding to free collocation points to zero: $\boldsymbol{u}_D(\boldsymbol{X}_F) = 0,\ \boldsymbol{X}_F \in B \cup \Gamma_N,\ \boldsymbol{X}_F \notin \Gamma_D$, such that the displacement field reads:

$$\tilde{\boldsymbol{u}}(\boldsymbol{X}) = \boldsymbol{u}_D(\boldsymbol{X}) + \boldsymbol{A}(\boldsymbol{X})\tilde{\boldsymbol{z}}(\boldsymbol{X}) \quad . \tag{3}$$

Accordingly, $\tilde{\boldsymbol{u}}$ fulfills the Dirichlet BCs and is differentiable w.r.t. $\boldsymbol{X}$ so that the NN parameters are are trained as the solution of an unconstrained optimization problem.

The corresponding loss function to minimize is

$$\mathcal{L}^{\mathrm{DEM}}(\tilde{\boldsymbol{u}}) = \tilde{\Pi} = \int_B \left(\Psi\left(\tilde{\boldsymbol{F}}\right) - \boldsymbol{b}\cdot\tilde{\boldsymbol{u}}\right)\,\mathrm{d}V - \int_{\Gamma_N} \overline{\boldsymbol{T}}\cdot\tilde{\boldsymbol{u}}\,\mathrm{d}A \tag{4}$$

with the deformation gradient $\tilde{\boldsymbol{F}} = \frac{\partial(\boldsymbol{X}+\tilde{\boldsymbol{u}})}{\partial \boldsymbol{X}}$, body forces $\boldsymbol{b}$ and the traction $\overline{\boldsymbol{T}} = \boldsymbol{P}\cdot\boldsymbol{N}$ defined as the 1^{st} Piola-Kirchhoff stress tensor $\boldsymbol{P}$ projection on the outward normal N of Γ. The described information flow is illustrated in Fig. 1.

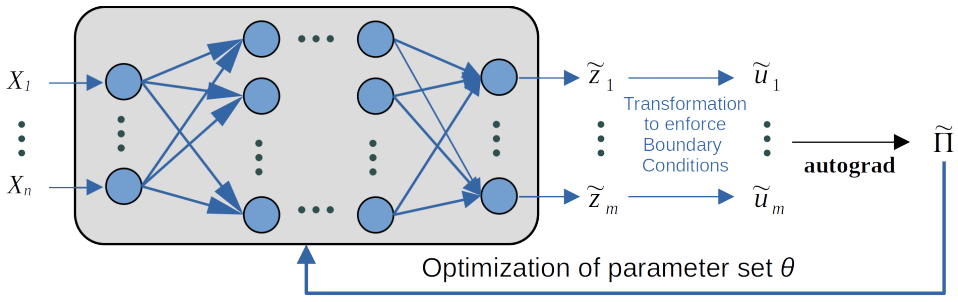


Figure 1: Information flow in DEM [12].

2.2.2 Constitutive Artificial Neural Networks (CANN)

Constitutive Artificial Neural Networks (CANNs) [29, 30, 31] replace the conventional approach of fitting the stress-strain curves of hyperelastic materials by a manually chosen set of

analytic expressions with a generalized ANN approach. The activation functions in the CANN are oriented on typical function types in conventional hyperelastic material laws and summed up in order to retain polyconvexity. Alternatively, Kolmogorov-Arnold-Networks (KANs) and more generic ANN architectures combined with automatic pruning can be applied [32, 33, 34]

CANN learn the strain energy density Ψ depending on the deformation gradient $\mathbf{F}$ which inherently provides thermodynamic consistency. This way, the first Piola-Kirchhoff stress tensor can be obtained via autograd according to $\mathbf{P} = \frac{\partial \Psi}{\partial \mathbf{F}}$. On the input side, the invariants of $\mathbf{F}$ are calculated and passed into the ANN which enforces material objectivity, frame indifference, material symmetry and isotropy.

3 A NEW GENERIC ANN-APPROACH FOR SIMULATION IN SOLID MECHANICS: INTEGRATION OF DEM AND CANN IN A NESTED SETUP

In order to replace the analytical expressions for the material model in the DEM, a generic, neural approach is suggested in the following. The aim is to set up a completely automated process which integrates the material characterization and the simulation step and requires only a small number of experiments and no explicit knowledge of an material model. The structure is moreover designed to allow the application of the obtained material model in further simulations. The completely ANN based nature of this approach finally allows to perform the computation on the same highly parallelized and optimized hardware, such as graphics or specialized ML accelerator cards.

3.1 Training stage

In the training stage, the material characterization is carried out in a nested setup of DEM and CANN depicted in Fig. 2 resulting in two nested training loops. Input to the coupled CANN-DEM training are coordinates in the undeformed configuration $\mathbf{X}$, one or more sets of BCs and the displacements $\mathbf{u}$ of the measured deformation. The inner training of the DEM is run to predict the deformations based on the given BCs and the randomly initialized state of the CANN. From the difference between the DEM output $\tilde{u}_i$ and the measured deformation state u_i caused by the prescribed BCs, the loss value for the outer training of the CANN is calculated as

$$\mathcal{L}_{CANN} = \frac{1}{N} \sum_{i=1}^{N} (u_i - \tilde{u}_i)^2 \tag{5}$$

with N the number of grid points in the geometry. In this setup, every epoch of the CANN training requires training of the DEM to compute the predicted deformations $\tilde{\mathbf{u}}$. Since the CANN values are updated adaptively, the DEM training load is reduced in later epochs of the CANN training, comparable to the process of transfer learning. Fig. 2 shows the detailed structure of the CANN-DEM training stage. During the initial training phase, the DEM network requires significantly more epochs to converge. Even without the CANN optimizer being active, the CANN loss changes significantly during the first epochs due to variations in the calculated displacement fields. This introduces considerable noise in the CANN loss at the beginning of the training, thus jeopardizing the efficiency of the coupled training. To address this issue and better separate the effects of inner and outer training processes, a starting phase is introduced, where the CANN optimizer remains inactive until the last ten CANN losses are below a threshold ($outer_loss_conv_crit$).

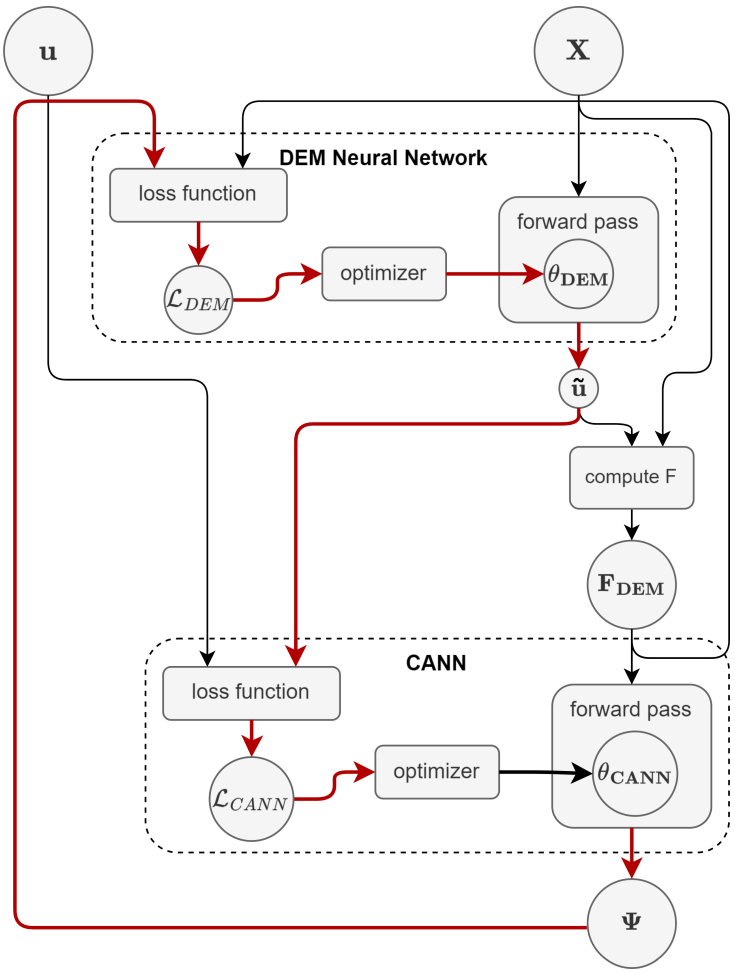


Figure 2: Nested structure of the CANN-DEM. The red path is the calculation path requiring differentiation through the DEM optimization process [12].

3.2 Application stage

After the training, the neural network approach can be used for simulations on the structural level where the inputs are sample points in the new domain as well as both Dirichlet and Neumann BCs. In this stage, the parameters of the CANN stay fixed and are not anymore subject to optimization. Only the optimizer of the DEM is applied to find a displacement field which satisfies the equilibrium condition based on the corresponding stresses calculated by the CANN. This training is the counterpart of the Newton-Raphson scheme in the FEM and visualized in Fig. 3.

4 CASE STUDY

4.1 Data generation

A multiaxial static tension-testing machine and an optical system to measure the surface displacements of the test specimen as described in [35, 36, 37] is simulated by a conventional run of the DEM in the present study in order to generate the training data. To this end, $N_x = 20, N_y = 5$ equidistant collocation points have been used. The ANN consists of three hidden layers with Tanh activation function with 8 neurons per hidden layer. The accuracy of this setup is investigated on a 2D cantilever beam from [28] that is made of a Neo-Hookean material with

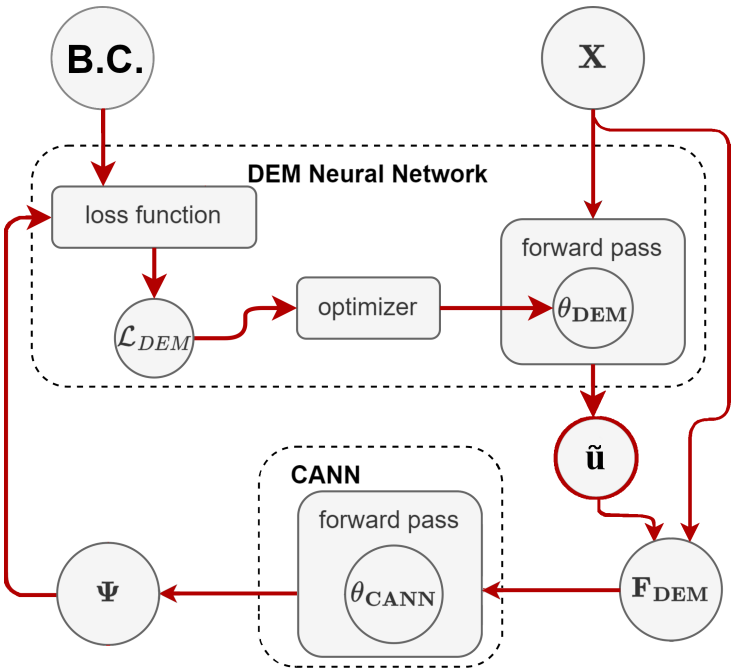


Figure 3: Structure of the CANN-DEM application phase. The CANN parameters are now fixed [12].

$E = 1000$ Pa and $\nu = 0.3$ and has length $L = 4$ m, height $H = 1$ m and width $D = 1$ m. The beam is clamped at the left side and subjected to a uniform traction load with the resulting force $R = -5.0$ N at the right end.

Note that the training data can also be generated by the traditional FEM, however the DEM simulation allows for direct analysis of the results at the same collocation points, thus facilitating their.

4.2 Hyperparameter

The model requires several hyperparameters to be set in order to run the simulation [12]. Besides conventional values such as the learning rates for the inner and outer optimizer, the layer widths for both ANNs and the numbers of layers and types of activations functions need to be defined. The nested setup requires to introduce further parameters such as a convergence criterion for the inner DEM training loop, weighing parameters of the inner against the outer loss function and adapted parameters during the starting phase and to determine when to end the starting phase.

The CANN layer width is varied in the case studies and stated for each numerical sample separately, together with the function that the CANN represents in symbolic form. Similar to [30], the CANN takes the deformation gradient as input, calculates its invariants, then passes them through a mixed layer with linear and quadratic functions, followed by another mixed layer with linear and exponential functions. A final linear layer combines all values into the scalar value for the strain energy density. The convergence criterion for each DEM training checks if the difference of the last $end_length = 4$ losses is less than $conv_crit$.

5 RESULTS

A twelve-parameter CANN with linear, quadratic, and exponential activation functions to capture complex material behaviors is trained based on the compressible Neo-Hookean model,

whereas additional case studies concerned with alternative material types are presented in [12].

The compressible Neo-Hookean model [28] follows the equation

$$\Psi(I_1, J) = \frac{1}{2}\lambda\left(\log(J)\right)^2 - \mu\log(J) + \frac{1}{2}\mu(I_1 - 2) \quad . \tag{6}$$

The prescribed values for the Lamé parameters in the simulated example are $\lambda = 16443\ \frac{\mathrm{N}}{\mathrm{m}^2}$ and $\mu = 336\ \frac{\mathrm{N}}{\mathrm{m}^2}$. The training utilizes six deformation states based on the load cases given in Table 1.

Table 1: The six different load cases used for the twelve parameter CANN-DEM training.

Load case	t_0	t_1	t_2	t_3	t_4	t_5
Load in x-direction / N	0	0	0	100	300	400
Load in y-direction / N	3	-15	50	-50	0	-100

The training process reduced the loss steadily, identifying the material law of the form

$$\begin{aligned}\Psi(I_1, J) = {} & 157.54 \cdot (I_1 - 2) + 0.00 \cdot \left(\exp\left(0.00 \cdot (I_1 - 2)\right) - 1\right) \\ & + 0.00 \cdot (J - 1) + 0.00 \cdot \left(\exp\left(0.00 \cdot (J - 1)\right) - 1\right) \\ & + 2.57 \cdot (I_1 - 2)^2 + 0.00 \cdot \left(\exp\left(0.00 \cdot (I_1 - 2)^2\right) - 1\right) \\ & + 1166.02 \cdot (J - 1)^2 + 600.47 \cdot \left[\exp(5.03 \cdot (J - 1)^2) - 1\right] \quad .\end{aligned} \tag{7}$$

The identified material model closely approximates the expected factor at the term $(I_1 - 2)$ (Eq. 6) and final deformations accurately match the numerical surrogate simulations shown in Fig. 4. The stretch-stress behavior confirms that the twelve-parameter CANN (Eq. 7) closely replicates the behavior of the underlying material model over a broad range of deformations.

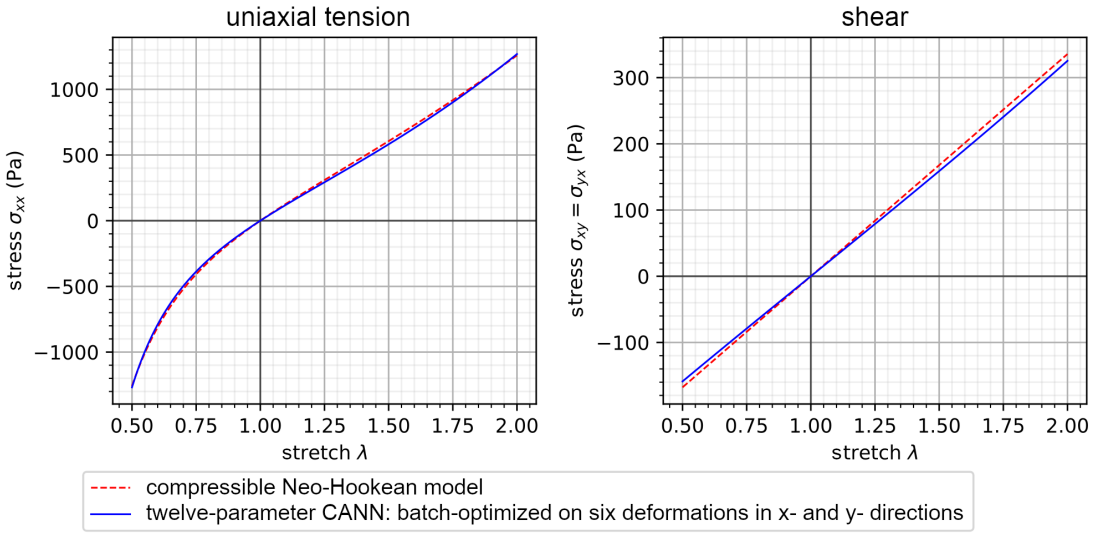


Figure 4: Comparison of stretch-stress behavior between the compressible Neo-Hookean model and the twelve-parameter CANN model trained on six deformations [12].

To assess the generalization capability of the trained twelve-parameter CANN, the approach is tested against three load conditions which are not part of the training data set, namely the

loads at the right end of the beam

$$\mathbf{t}_0 = [0\,\mathrm{N},\ 8\,\mathrm{N}]\,, \quad \mathbf{t}_1 = [100\,\mathrm{N},\ 0\,\mathrm{N}]\,, \quad \mathbf{t}_2 = [200\,\mathrm{N},\ 100\,\mathrm{N}]\,.$$

The results show that the predicted deformations closely match the numerical surrogate simulations (Fig. 5). The calculated loss values are within the same range as during the training.

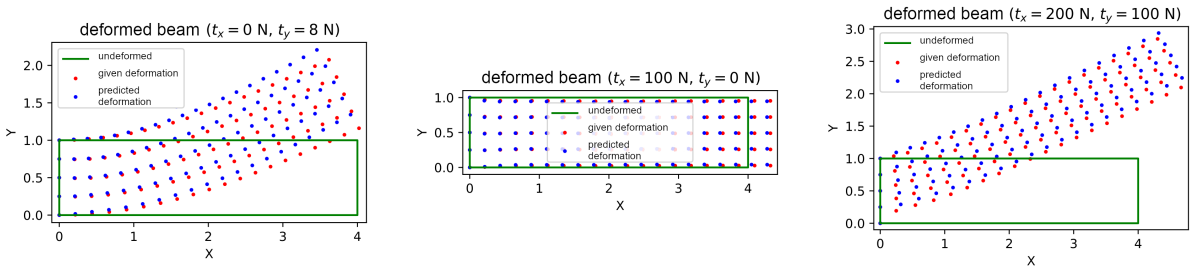


Figure 5: Comparison of the resulting deformation from given material model and predicted material model for three different load conditions that are not used in training [12].

6 CONCLUSIONS

In this study, a generic ANN-approach consisting of DEM and CANN in a nested setup has been investigated for elastostatic simulations. No explicit material model is required, as the CANN replaces analytical expressions to describe the material behavior. The model is trained based on a small number of deformation states and corresponding boundary conditions and can be used in further simulations on a structural level. The test presented in the contribution applies a twelve-parameter CANN for a compressible Neo-Hookean material. The CANN-DEM training has been conducted with six deformation states on 100 sample points.

The achieved results have shown that the coupled CANN-DEM architecture works robustly and performs comparable to a conventional method with an analytical constitutive model. The nested approach is shown to be able to identify hyperelastic material models within a space spanned by up to twelve parameters given training data that incorporates sufficiently different stress-strain states. The test results show a high accuracy in all case studies. Both the given displacement fields and the stretch-stress behavior of the given material models are approximated closely. The resulting CANN models generalize well for loading conditions which are not part of the training data. Accordingly, the presented CANN-DEM approach allows to conduct a completely ANN based simulation of structural behavior without a conventional constitutive model. Instead, a data-driven approach based on a small number of experiments is sufficient to reliably depict the material behavior and subsequently perform simulations at the structural level.

In conclusion, the integrated process is promising for application in a fully automatized lab setting, from the specimen to the final material model and simulation setup. Still, specimen of more complex shape, e.g. containing a hole or an L-shape sample, and more complex loading conditions, which lead to stress concentrations and a variety of multiaxial stress states, are expected to reduce the required number of load cases further [38, 39, 40]. To allow a complete 3D characterization of the material, a multi-camera setup could give the necessary data in a single pass. Moreover, deformation data should be included that contains larger stretches. This could help to reduce unexpected parameter combinations in the resulting CANN model.

REFERENCES

[1] Bock F E, Aydin R C, Cyron C J, Huber N, Kalidindi S R and Klusemann B 2019 *Frontiers in Materials* **6** ISSN 2296-8016 URL https://www.frontiersin.org/articles/10.3389/fmats.2019.00110

[2] Cicirello A 2024 *J. Phys.: Conf. Ser.* **2909** 012034 URL https://doi.org/10.1088/1742-6596/2909/1/012034

[3] Le Clainche et al. 2023 *Aero. Sc. Tech.* 108354 URL https://doi.org/10.1016/j.ast.2023.108354

[4] Shoghi R and Hartmaier A 2024 *Mach. Learn.: Sci. Technol.* **5** 025008 URL https://doi.org/10.1088/2632-2153/ad379e

[5] Hildebrand S and Klinge S 2024 *Mach. Learn.: Sci. Technol.* **5** 045058 URL https://iopscience.iop.org/article/10.1088/2632-2153/ad95da

[6] Rezaei S, Moeineddin A and Harandi A 2024 *Computational Mechanics* **74** 333–366 ISSN 1432-0924 URL https://doi.org/10.1007/s00466-023-02435-3

[7] Shoghi R and Morand L and Helm D and Hartmaier A 2024 *Comp. Mech.* **74** 447-466 URL https://doi.org/10.1007/s00466-023-02440-6

[8] Karabulut M 2025 *Buildings* **15**(3) 432 URL https://doi.org/10.3390/buildings15030432

[9] Chadha C, He J, Abueidda D, Koric S, Guleryuz E and Jasiuk I 2024 *Acta Mechanica* **234** 5975–5998 URL https://doi.org/10.1007/s00707-023-03691-3

[10] Liang Z and Gao H and Li T 2022 *Acta Mechanica* **233** 1739-1755 URL https://doi.org/10.1007/s00707-022-03174-x

[11] Wang Y et al. 2025 *Comp. Meth. Appl. Mech. Eng.* URL https://doi.org/10.1016/j.cma.2024.117518

[12] Hildebrand S and Friedrich J G and Mohammadkhah M and Klinge S 2025 *Mach. Learn.: Sci. Technol.* **6** 015038 URL https://doi.org/10.1088/2632-2153/adaf74

[13] Mohammadkhah M and Klinge S 2023 *Journal of the Mechanical Behavior of Biomedical Materials* **148** 106203 ISSN 1751-6161 URL https://www.sciencedirect.com/science/article/pii/S1751616123005568

[14] Graham M and Klinge S 2024 *Journal of the Mechanical Behavior of Biomedical Materials* **149** 106244 ISSN 1751-6161 URL https://www.sciencedirect.com/science/article/pii/S1751616123005970

[15] Eisenträger J and Naumenko K and Altenbach H 2018 *Acta Mech.* **229** 3051-3068 URL https://doi.org/10.1007/s00707-018-2151-1

[16] Kissas G and Mishra S and Chatzi E and De Lorenzis L 2024 *Comp. Meth. Appl. Mech. Eng.* **428** 177053 URL https://doi.org/10.1016/j.cma.2024.117053

[17] Garai A and Haldar K 2025 *Int. J. Mech. Sc.* **286** 109860 URL https://doi.org/10.1016/j.ijmecsci.2024.109860

[18] Aygün S and Klinge S 2020 *Polymers* **12** ISSN 2073-4360 URL https://www.mdpi.com/2073-4360/12/11/2575

[19] Ali A, Hosseini M and Sahari B B 2010 *American Journal of Engineering and Applied Sciences* **3** 232–239 accessed: 2024-07-05 URL https://thescipub.com/abstract/10.3844/ajeassp.2010.232.239

[20] Kalina K A 2025 *Comp. Meth. Appl. Mech. Eng.* URL https://doi.org/10.1016/j.cma.2024.117725

[21] Abueidda D W, Koric S, Abu Al-Rub R, Parrott C M, James K A and Sobh N A 2021 *European Journal of Mechanics - A/Solids* **372** 113431 URL https://doi.org/10.1016/j.euromechsol.2022.104639

[22] Hildebrand S and Klinge S 2024 *Neural Computing and Applications* **36** 16657-16682 URL https://doi.org/10.1007/s00521-024-10132-2

[23] Hornik K 1991 *Neural Networks* **4** 251–257 URL https://www.sciencedirect.com/science/article/pii/089360809190009T

[24] Raissi M, Perdikaris P and Karniadakis G 2019 *Journal of Computational Physics* **378** 686–707 ISSN 0021-9991 URL https://www.sciencedirect.com/science/article/pii/S0021999118307125

[25] Wang S, Teng Y and Perdikaris P 2021a *SIAM Journal on Scientific Computing* **43** A3055–A3081 (*Preprint* https://doi.org/10.1137/20M1318043) URL https://doi.org/10.1137/20M1318043

[26] Goswami S, Anitescu C and Rabczuk T 2020 *Theoretical and Applied Fracture Mechanics* **107** 102527 ISSN 0167-8442 URL https://www.sciencedirect.com/science/article/pii/S0167844219306858

[27] Samaniego E, Anitescu C, Goswami S, Nguyen-Thanh V, Guo H, Hamdia K, Zhuang X and Rabczuk T 2020 *Comp. Meth. Appl. Mech. Eng.* **362** 112790 ISSN 0045-7825 URL https://www.sciencedirect.com/science/article/pii/S0045782519306826

[28] Nguyen-Thanh V M, Zhuang X and Rabczuk T 2020 *European Journal of Mechanics - A/Solids* **80** 103874 ISSN 0997-7538 URL https://www.sciencedirect.com/science/article/pii/S0997753819305352

[29] Peirlinck M, Linka K, Hurtado J A and Kuhl E 2024 *Comp. Meth. Appl. Mech. Eng.* **418** 116534 ISSN 0045-7825 URL https://www.sciencedirect.com/science/article/pii/S0045782523006588

[30] Linka K and Kuhl E 2023 *Comp. Meth. Appl. Mech. Eng.* URL https://www.sciencedirect.com/science/article/pii/S0045782522006867

[31] Linka K, Hillgärtner M, Abdolazizi K P, Aydin R C, Itskov M and Cyron C J 2021 *Journal of Computational Physics* **429** 110010 URL https://www.sciencedirect.com/science/article/pii/S0021999120307841

[32] Abdolazizi K P and Aydin R C and Cyron C J and Linka K 2025 *arXiv* URL https://doi.org/10.48550/arXiv.2502.05682

[33] Fuhg J N and Jones R E and Bouklas N 2024 *Comp. Meth. Appl. Mech. Eng.* URL https://doi.org/10.1016/j.cma.2024.116973

[34] Schommartz J O and Klein D K and Alzate Cobo J C and Weeger O 2025 *Comp. Meth. Appl. Mech. Eng.* **435**117592 URL https://doi.org/10.1016/j.cma.2024.117592

[35] Cooreman S, Lecompte D, Sol H, Vantomme J and Debruyne D 2008 *Experimental Mechanics* **48**(4) 421–433 ISSN 1741-2765 URL https://doi.org/10.1007/s11340-007-9094-0

[36] Polanco-Loria M, Daiyan H and Grytten F 2012 *Polymer Engineering & Science* **52** 438–448 URL https://4spepublications.onlinelibrary.wiley.com/doi/abs/10.1002/pen.22102

[37] Zhao J, Dong J, Liu Z and Xie H 2019 *Polymer Testing* **79** 106015 ISSN 0142-9418 URL https://www.sciencedirect.com/science/article/pii/S0142941819309213

[38] Flaschel M, Kumar S and De Lorenzis L 2022 *npj Computational Materials* **8** ISSN 2057-3960 URL https://doi.org/10.1038/s41524-022-00752-4

[39] Thakolkaran P *et al.* 2022 *arXiv* (*Preprint* 2205.06664) URL https://arxiv.org/abs/2205.06664

[40] Jeong I, Cho M, Chung H and Kim D N 2024 *Comp. Meth. Appl. Mech. Eng.* **418** 116569 URL https://www.sciencedirect.com/science/article/pii/S004578252300693X

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

VIBRATIONS ANALYSIS OF STRUCTURAL PROPERTIES WITH UNCERTAINTIES USING OSCILLATORY PHYSICS-INFORMED NEURAL NETWORKS (OPINN)

Stefan Hildebrand[1], **and Sandra Klinge**[1]

[1] Department of Structural Mechanics and Analysis, Technische Universität Berlin
Straße des 17. Juni 135, 10623 Berlin, Germany
e-mail: {stefan.hildebrand, sandra.klinge}@tu-berlin.de

Abstract. *Physics-Informed Neural Networks (PINN) methods have proven to be highly versatile and have been successfully applied to a broad variety of problems, such as elastostatics, heat transfer, material modeling or fluid flow. For the investigation of vibrations, a novel oscillatory Physics-Informed Neural Network (oPINN) architecture is developed. It allows for a great variety of hybrid Physics-Informed and data-driven analyses of structures, particularly in cases of incomplete or uncertain information about the system properties. In the oPINN framework, sparse or noisy vibration measurements can be integrated with structural modeling to obtain an accurate and comprehensive system representation. Additionally, estimates of a structure's eigenfrequencies can be included to reduce the computational effort. The oPINN approach achieves accuracy comparable to or exceeding conventional Adams Predictor-Corrector methods. High stability, especially in numerically challenging cases is demonstrated. Transfer learning can significantly enhance computational efficiency for series of similar setups, which might play an important role in parameter studies or multi-frequency excitation analyses. Moreover, oPINN can be employed as a modal reduction technique.*

Keywords: Machine Learning, Physics-Informed, Neural Networks, Vibrations, Oscillations, Elastic continua

1 INTRODUCTION

Many engineering tasks require dynamic simulations of structures, specifically with respect to vibrations. Examples are earthquake engineering, acoustics, failsafe and reliable design of engines and turbines [1, 2, 3], design of bearings [4, 5], brakes [6] and in railway engineering [7]. Currently, the finite element method (FEM) is the standard computational method for transient simulations of oscillating structures. For high frequencies, high-resolution temporal and spatial discretization is required, which makes the simulation computationally challenging [8, 9, 10]. Hyperbolic equations like the wave equation bring challenges to the numerical solution as they often exhibit local nonlinear shock waves while Dahlquist's barriers describe the limitations in regard of stability and accuracy properties of the broad and important class of the linear multistep time-stepping methods [11, 12, 13, 14].

Currently, the use of machine learning (ML) with neural networks (NN) [15, 16, 17] gains increasing attentions in a variety of fields due to their ability to universally approximate functions paired with outstanding parallelizability [18, 19]. Among others, very promising results have been presented for modeling of plasticity [20, 21, 22, 23], hyperelasticity and magnetoactivity [24, 25, 26] and steel-reinforced concrete [27]. In the area of physical problems, physics-informed neural networks (PINN) are a promising technique which does not require training data. Instead, the differential equation describing the respective physical phenomena is evaluated and its residual minimized to train the neural network. This way, the network is trained to output the solution of the differential equation at the sample or collocation points given as inputs. Data sources, such as pre-training with measurement data, can be added [28, 29]. For hyperbolic equations, incorporating the method of characteristics [30] has been suggested as well as physics informed attention neural networks (PIANN), which combine recurrent neural networks and attention mechanisms [31]. Similar to conventional time-stepping methods, an artificial viscosity term can be added to the PDE to increase the solution stability [32], whereas the viscosity coefficient ν can be trained as another learnable parameter. Despite these suggestions, the literature on the application of PINNs to dynamic problems largely focuses on test cases with low frequencies [33], since PINNs tend to prioritize learning lower frequency modes, while high-frequency patterns often do not converge. This can be addressed by transfer learning from low-frequency cases towards higher frequency cases [34, 35].

The proposed oPINN architecture adresses this issue primarily by an internal architecture that facilitates the learning with a defined range of frequencies and demonstrates superior accuracy and stability compared to the predictor-corrector time-stepping scheme chosen as reference. Moreover, the oPINN architecture allows an easy integration of all known data, such as sparse or noisy measurement data or related simulation results, to improve the accuracy, for acceleration by pre-training and transfer learning. Besides, oPINN can at the same time deliver a modal analysis and with reduced layer width serve as modal reduction, where the eigenfrequencies and eigenshapes of a structure are obtained directly from the NN parameters.

2 GOVERNING EQUATIONS

The wave equation

$$\frac{\partial^2 \boldsymbol{u}}{\partial t^2} - c^2 \Delta \boldsymbol{u} = \boldsymbol{0} \tag{1}$$

with the elastic deflection $\boldsymbol{u}(\boldsymbol{x}, t)$, the sound velocity c and time t is a hyperbolic equation and applies to both the tensile bar and the string.

The wave speed for the tensile bar is

$$c^2 = \frac{E}{\rho} \tag{2}$$

where E is the Young's modulus and ρ the material density.

The general homogeneous solution of the homogeneous wave equation is [36]:

$$u_h(x,t) = \sum_{j=1}^{\infty} \left(A_j \cos\left(\frac{\omega_j}{c}x\right) + B_j \sin\left(\frac{\omega_j}{c}x\right)\right) \cdot \left(C_j \cos(\omega_j t) + D_j \sin(\omega_j t)\right), \tag{3}$$

$$A_j, B_j, C_j, D_j \in \mathbb{R}.$$

For a system with a force excitation, the particular solution needs to be added to the homogeneous solution to get the general solution. For a harmonic sine force, the particular solution reads

$$u_p(x,t) = \left(E \cos\left(\frac{\Omega}{c}x\right) + F \sin\left(\frac{\Omega}{c}x\right)\right) \sin(\Omega t) \quad . \tag{4}$$

In case of a harmonic cosine excitation, the equation is instead multiplied with $\cos(\Omega t)$. The eigenfrequencies are determined from the homogeneous solution, however, their values depend on the boundary conditions.

A one-sided clamped bar has the following eigenfrequencies [37]:

$$\omega_j = \frac{(2j+1)\pi}{2}\frac{c}{l}, \; j = 0, 1, \ldots . \tag{5}$$

The eigenfrequencies for a bar that is fixed on both ends, as well as for a bar that is free at both ends, are [37]:

$$\omega_j = (j+1)\pi\frac{c}{l}, \; j = 0, 1, \ldots . \tag{6}$$

Only the eigenmodes differ for these two cases. In the following, the kinetic and strain energy are employed as a means to evaluate the accuracy of simulation results, since the trade-off between stability and accuracy in conventional time stepping algorithms often is quantified by the loss in the calculated energy over the simulated time interval. This reduction of energy has a urely numerical character and is often considered as numerical viscosity. The kinetic energy of a tensile bar is [38]

$$E_{kin} = \frac{1}{2}\int_0^l \rho A \dot{u}^2 \mathrm{d}x \tag{7}$$

and the strain energy

$$E_{str} = \frac{1}{2}\int_0^l EAu'^2 \mathrm{d}x \quad . \tag{8}$$

The work done by an excitation force is

$$W_{force}(t) = \int_0^l F(t)\mathrm{d}u. \tag{9}$$

The balance of energy in the oscillating system is given by

$$E_{tot} = E_{kin}(t=0) + E_{str}(t=0) + W_{force}(t) = E_{kin}(t) + E_{str}(t) \quad . \tag{10}$$

3 METHODS AND MATERIALS

3.1 Physics-Informed Neural Networks (PINNs)

NNs with parameter set Θ consisting of the weights w and biases b represent an arbitrary mapping $\mathcal{R} : \mathbb{R}^n \mapsto \mathbb{R}^m$ from the input to the output side, with n and m the input and output layer width, respectively (universal approximation theorem, [39, 40]). For training, gradient descent based optimization algorithms like Adam [41] are used, based on the MSE loss which typically yields robust and accurate results

$$\mathrm{MSE} = \frac{1}{N}\frac{1}{m}\sum_{k=1}^{N}\sum_{i=1}^{m}(R_i(P^k) - \tilde{R}_i(P^k))^2 \quad . \tag{11}$$

Here, P denotes the input to the NN, R the target output and $\tilde{R}$ the actual NN output of dimension m with N entries. In the following, quantities denoted with a tilde indicate NN outputs or quantities that are calculated based on NN outputs. Thus, all quantities with the tilde symbol are implicitly dependent on the NN parameter set Θ.

Data driven methods rely on available measurements and have been, among other scenarios, applied to detect and characterize brake squeal [42], conduct system identification of nonlinear oscillators [43], performance analysis of control system structures [44] and the prediction of nonlinear dispersive waves [45]. PINNs, on the contrary, use the underlying physics of a problem formulated as the residual of a partial differential equation (PDE) which is employed as the loss function $\mathcal{L}_{PDE}$ to predict the behavior of a system [28, 46]. Hence, PINNs represent a collocation method which can be used without the need of complex meshes [30].

The boundary conditions can be included in two different ways. One way is to include the boundary term in the loss as $\mathcal{L}_{BC}$, which is called weak enforcement [29]. The total loss then consists of [47]:

$$\mathcal{L} = \mathcal{L}_{PDE} + \mathcal{L}_{BC}. \tag{12}$$

Another method is to ensure that the boundary conditions are fulfilled a priori by modifying the NN output [48, 29, 49] which converts the constrained optimization problem into an unconstrained optimization problem

Extensions of PINN include competitive PINN[50] and variational PINN [51]. A variant for solid body mechanics is the deep energy method (DEM) [48, 52] which employs the potential energy of a structure as loss value which has to be minimal in the state of mechanical equilibrium.

3.2 Numerical reference solution method

As reference solution, the widely used Adams predictor-corrector scheme is applied, which allows for a good compromise between stability and accuracy [13]. Specifically, the predictor step is calculated with the second-order Adams-Bashforth method [12]

$$\dot{\tilde{u}}_{t+1} \approx \dot{\tilde{u}}_t + \Delta t\left(\frac{3}{2}\ddot{\tilde{u}}_t - \frac{1}{2}\ddot{\tilde{u}}_{t-1}\right) \quad , \qquad \tilde{u}_{t+1} \approx \tilde{u}_t + \Delta t\left(\frac{3}{2}\dot{\tilde{u}}_t - \frac{1}{2}\dot{\tilde{u}}_{t-1}\right) \tag{13}$$

followed by the corrector step according to the second-order Adams-Moulton method

$$\dot{u}_{t+1} \approx \dot{\tilde{u}}_t + \Delta t\left(\frac{1}{2}\ddot{u}_{t+1} + \frac{1}{2}\ddot{\tilde{u}}_t\right) \quad , \qquad u_{t+1} \approx \tilde{u}_t + \Delta t\left(\frac{1}{2}\dot{\tilde{u}}_{t+1} + \frac{1}{2}\dot{\tilde{u}}_t\right) \quad . \tag{14}$$

The spatial discretization is conducted with central differences of second order and a mesh width of Δx. The temporal discretization is adapted manually until stability is reached.

4 SUGGESTED OSCILLATORY PINN (oPINN) ARCHITECTURE

The custom oPINN architecture with input being the sample coordinates x and time steps t and the output being the displacement u is shown in Fig. 1.

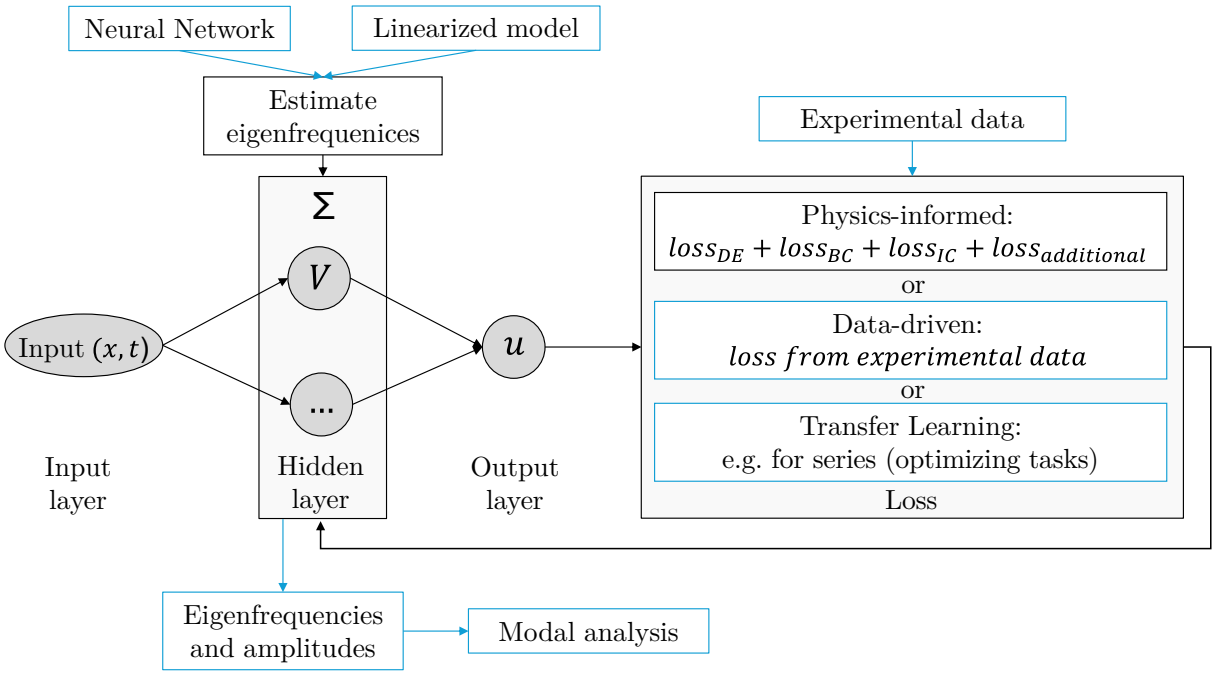


Figure 1: oPINN architecture

Only one hidden layer is used in the network. Harmonic functions are used as activation functions, so that the output of the NN corresponds to the general solution of the wave equation, including the homogeneous solution and the particular solution

$$u(x,t) = \sum_{j=1}^{d} \left(A_j \cos\left(\frac{\omega_j}{c}x\right) + B_j \sin\left(\frac{\omega_j}{c}x\right) \right) \left(C_j \cos(\omega_j t) + D_j \sin(\omega_j t) \right) + P(x,t) \quad , \tag{15}$$

where d is the number of neurons in the hidden layer. The analytical solution has an infinite number of summands. $P(x,t)$ represents the particular solution which in case of a sine excitation is:

$$P(x,t) = \left(E \cos\left(\frac{\Omega}{c}x\right) + F \sin\left(\frac{\Omega}{c}x\right) \right) \sin(\Omega t) \quad . \tag{16}$$

The variables A_j, B_j, C_j, D_j, E, F, ω_j and Ω in Eq. (15) are in this setup the trainable parameters of the oPINN.

These parameters should be initialized in a similar order of magnitude as the corresponding values in the solution. This information can e.g. stem from experience with similar systems, other simplified simulations, measurements or another data-driven ML application that serves as an estimator. For testing purposes, the initialization values for the eigenfrequencies are drawn from a uniform distribution between 0.5 and 1.5 times the analytical value.

The loss is calculated based on the residual of the wave equation $\mathcal{L}_{PDE}$ and additional terms for the boundary conditions $\mathcal{L}_{BC}$ and initial conditions $\mathcal{L}_{IC}$

$$\mathcal{L} = \mathcal{L}_{PDE} + \lambda_1 \mathcal{L}_{BC} + \lambda_2 \mathcal{L}_{IC} \quad , \tag{17}$$

where the scaling factors λ_i are introduced to weigh the contributions of the loss terms against each other following the considerations from [**?**].

The Dirichlet boundary conditions can be enforced by applying a transformation on the NN output [53, 48]. For the oscillating, left-clamped bar, the boundary condition $u(x = 0, t) = 0$ can be introduced by fixing the constant $A_j = 0$. Then, the general homogeneous solution simplifies to

$$u_h(x,t) = \sum_{j=1}^{hd} B_j \sin\left(\frac{\omega_j}{c} x\right) \left(C_j \cos(\omega_j t) + D_j \sin(\omega_j t)\right) \quad . \tag{18}$$

For the training, the Adam optimizer is employed with learning rate decay halving the learning rate every 100 epochs to refine the solution[54, 55]. As default values, 3000 time step samples in an interval of $1.5\,\mathrm{s}$, an equidistant spatial discretization with 50 sample points and 20 neurons are used. Changes on these default values are indicated explicitly in the following.

5 CASE STUDY – Sawtooth displacement of a tensile bar

The case study presented here is chosen to be maximally challenging for a conventional numerical timestepper, as it deals with the sawtooth displacement movement which in the analytic solution leads to infinitely high frequencies.

It can be produced, when a tensile bar clamped on the left hand side is deformed linearly in its initial state as shown in Fig. 2. The bar has the length $l = 1\,\mathrm{m}$ and is made out of a material with Young's modulus $E = 10\frac{N}{m^2}$ and density $\rho = 1\frac{kg}{m^3}$. The initial deformation at the right end is specified as $u(x = l, t = 0) = 0.1\,\mathrm{m}$ and the initial displacement is linearly distributed over the length of the bar. After the application of the deformation at $t = 0\,\mathrm{s}$, the bar oscillates freely. This is equivalent to a system that is deformed by a force applied at time $t = 0\,\mathrm{s}$ and removed immediately afterwards.

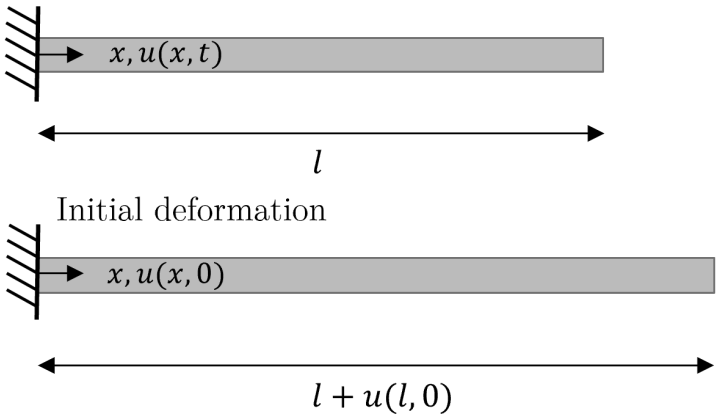


Figure 2: Case study – a tensile bar with an initial linear deformation.

The analytical solution for this case reads

$$u(x,t) = \frac{2F}{EAl} \sum_{j=0}^{\infty} \frac{(-1)^j}{\lambda_j^2} \cos(\omega_j t) \sin(\lambda_j x) \quad . \tag{19}$$

Approximations to this solution with different numbers of summands are visualized in Fig. 3.

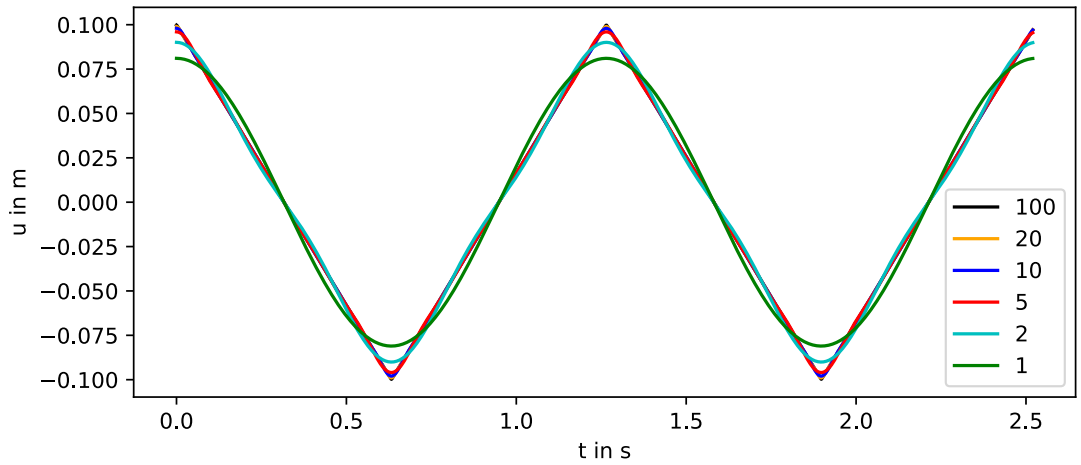


Figure 3: Approximations of the analytical solution with different numbers of summands.

The reference solution is set up with 50 spatial discretization points and 474 time steps which were chosen to ensure stability. The movement of the right end of the bar follows a sawtooth pattern which can be attributed to the initial condition of the system. The initial wave packet moves with constant shape back and forth through the bar. The oPINN is trained for 2000 epochs on collocation points with 50 spatial and 3000 time steps. The number of neurons is varied between one and 200.

The results obtained by oPINN with 20 neurons are compared to the analytical solution and the numerical reference solution in Fig. 4. Fig. 4 (left) shows that the reference solution achieves acceptable agreement with the analytical solution, but over the time of 10 seconds, a significant drift in the frequency builds up and a reduction of the amplitude corresponding to a significant numerical dissipation can be identified. In opposition to that, the relative error of the amplitudes of the oPINN solution remains constant and amounts to 7.3%, while the shape of the displacement is in very good accordance with the expected sawtooth pattern. Moreover, the amplitude of the oPINN solution is constant and long-time stable, which has been additionally confirmed by further studies on extended time intervals. The comparison of the calculated total energies is shown in Fig. 4 (right). The energies calculated from the oPINN and the analytic solution oscillate around a constant and stable mean value. These oscillations are comparatively small and most likely caused by numerical limitations during the postprocessing. The numerical reference solution, however, shows a significant, monotonic decrease in the calculated total energy and thus exhibits significant numerical viscosity [12, 13, 11].

A comparison of results achieved with different numbers of neurons is shown in Fig. 5. The lowest frequency is always met with high accuracy and acceptable accuracy of amplitudes, while the increase of parameters allows the improvement of the fit at the peaks of the displacement curve. This indicates the applicability of oPINN with a reduced number of neurons directly as a tool for modal reduction.

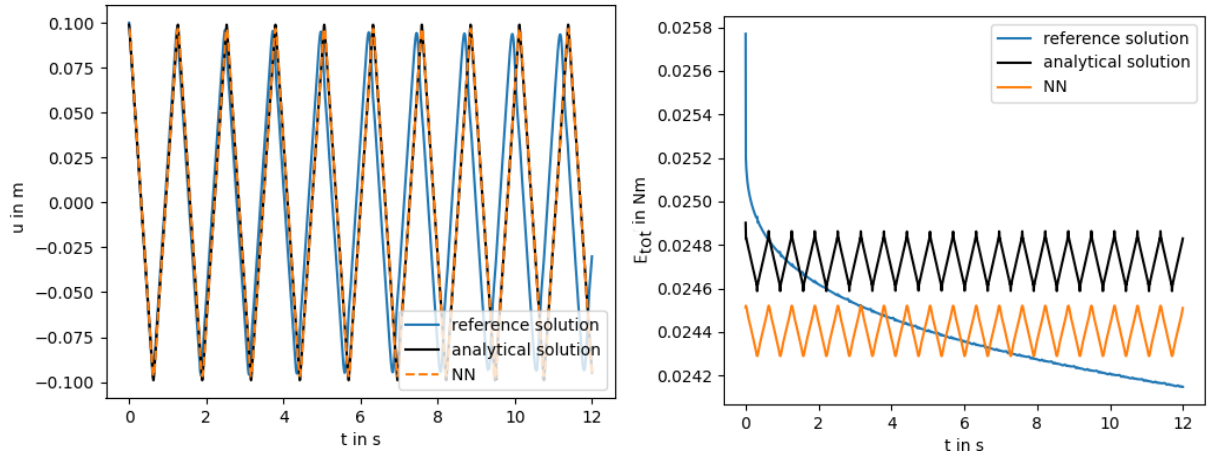


Figure 4: Comparison of the oPINN solution (20 neurons) with the numerical reference solution and the analytical solution. Comparison for the displacement of the point at the right end of the clamped bar (left). Comparison of the total energy (total sum of kinetic and potential energy) (right).

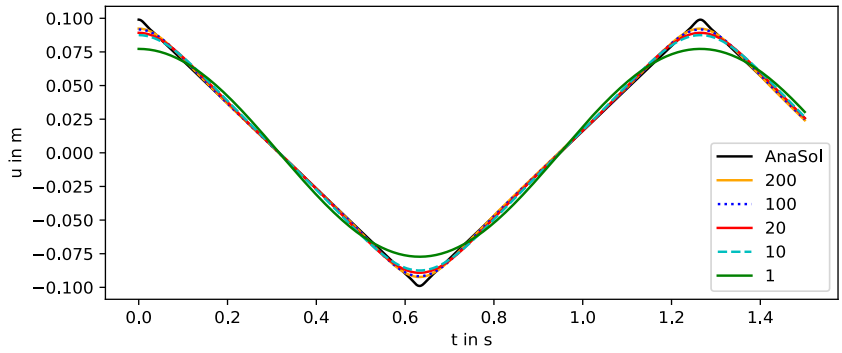


Figure 5: Comparison of the oPINN solution with different numbers of neurons and the analytical solution for the displacement of the point at the right end of the clamped bar.

6 CONCLUSIONS and OUTLOOK

In the current contribution, the new oPINN architecture for the analysis of vibrating structures with physics-informed neural networks has been presented. The architecture allows to combine the physical modeling of the system subject to analysis with several additional data sources, such as noisy or sparse measurement data, data-driven estimations and prior simulation results. At the same time, oPINN can be applied for modal analysis and modal reduction. The case study demonstrates the superiority of oPINN over conventional time-stepping algorithms. This is made possible by the two-sided temporal difference stencils and the search for the solution at the whole time interval of interest at once so that Dahlquist's barriers do not apply. The robustness of the method is underlined by the investigation on a case study that includes a sawtooth displacement curve with infinitely high frequencies.

In a future work, a more extensive investigation of the numerical properties of oPINN will be presented. Similarly, a combination of transfer learning with a rigorous hyperparameter optimization promises to significantly reduce the computational effort of the oPINN setup, especially for parameter studies and design optimization tasks and will be carried in future research. Finally, the application on two-dimensional and three-dimensional systems and adaptations for dissipative and nonlinear systems will be presented in future work.

REFERENCES

[1] Smetankina N, Pak A, Mandrazhy O, Usatova O and Vasiliev A 2023 Modelling of free axisymmetric vibrations of the fluid-filled shells with non-classical boundary interface conditions *Smart Technologies in Urban Engineering* ed Arsenyeva O, Romanova T, Sukhonos M, Biletskyi I and Tsegelnyk Y (Cham: Springer Nature Switzerland) pp 185–196 ISBN 978-3-031-46874-2

[2] Nicoletti R and Liebich R 2018 *Journal of Intelligent Material Systems and Structures* **29** 3108–3123 URL https://doi.org/10.1177/1045389X18783078

[3] Cicirello A 2024 *J. Phys.: Conf. Ser.* **2909** 012034 URL https://doi.org/10.1088/1742-6596/2909/1/012034

[4] Schilling G and Bäuerlein K and Liebich R 2019 *SIRM 2019 – 13th International Conference on Dynamics of Rotating Machines, Copenhagen, Denmark, 13th – 15th February 2019* URL https://www.static.tu.berlin/fileadmin/www/10002284/KuP/Forschen/Publikationen/SIRM2019_Numerical_Description_of_a_Rotor_supported_by_Gas_Polymer_Bearings.....pdf

[5] Pfeil S and Gravenkamp H and Duvigneau F and Woschke E 2023 *Int. J. Mech. Sc.* **247** 108164 URL https://doi.org/10.1016/j.ijmecsci.2023.108164

[6] Gräbner N and Tiedemann M and von Wagner Utz and Hoffmann N 2014 *SAE Technical Paper* URL https://doi.org/10.4271/2014-01-2511

[7] Stojanović V, Petković M D and Deng J 2018 *International Journal of Mechanical Sciences* **136** 155–168 ISSN 0020-7403 URL https://www.sciencedirect.com/science/article/pii/S0020740317317514

[8] Kranjc T, Slavič J and Boltežar M 2016 *Journal of Vibration and Control* **22** 371–381 URL https://doi.org/10.1177/1077546314533137

[9] Strzalka C 2023 *Untersuchungen zur modalen Superposition für die a priori Detektion versagenskritischer Regionen dynamisch beanspruchter Strukturen* Ph.D. thesis Technische Univesität Berlin

[10] Capuano G, Ruzzene M and Rimoli J J 2018 *Finite Elements in Analysis and Design* **145** 10–19 ISSN 0168874X

[11] Givoli D 2023 *J. Comp. Phys.* **475** 111836 URL https://doi.org/10.1016/j.jcp.2022.111836

[12] Hairer E and Wanner G 1996 *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems* vol 14 (Berlin, Heidelberg: Scholars Portal) ISBN 978-3-642-05220-0

[13] Ghrist M L, Fornberg B and Reeger J A 2015 *BIT Numerical Mathematics* 1572–9125

[14] Tattersall P and McGuirk J 1994 *Computers & Fluids* **23** 177–209 ISSN 0045-7930 URL https://www.sciencedirect.com/science/article/pii/0045793094900345

[15] Karniadakis G E and Keyredikis I G and Lu L et al. 2021 *Nat. Rev. Phys.* **3** 422-440 URL https://doi.org/10.1038/s42254-021-00314-5

[16] Hildebrand S and Friedrich J G and Mohammadkhah M and Klinge S 2025 *Mach. Learn.: Sci. Technol.* **6** 015038 URL https://doi.org/10.1088/2632-2153/adaf74

[17] Le Clainche et al. 2023 *Aero. Sc. Tech.* 108354 URL https://doi.org/10.1016/j.ast.2023.108354

[18] Arabshahi F and Singh S and Anandkumar A 2018 *ICML workshop Neural Abstract Machines & Program Induction v2 (NAMPI)* URL https://forougha.github.io/paperPDF/neuralODE_paper.pdf

[19] Bock F E, Aydin R C, Cyron C J, Huber N, Kalidindi S R and Klusemann B 2019 *Frontiers in Materials* **6** ISSN 2296-8016 URL https://www.frontiersin.org/articles/10.3389/fmats.2019.00110

[20] Hildebrand S and Klinge S 2024 *Mach. Learn.: Sci. Technol.* **5** 045058 URL https://iopscience.iop.org/article/10.1088/2632-2153/ad95da

[21] Shoghi R and Morand L and Helm D and Hartmaier A 2024 *Comp. Mech.* **74** 447-466 URL https://doi.org/10.1007/s00466-023-02440-6

[22] Shoghi R and Hartmaier A 2024 *Mach. Learn.: Sci. Technol.* **5** 025008 URL https://doi.org/10.1088/2632-2153/ad379e

[23] Rezaei S, Moeineddin A and Harandi A 2024 *Computational Mechanics* **74** 333–366 ISSN 1432-0924

[24] Garai A and Haldar K 2025 *Int. J. Mech. Sc.* **286** 109860 URL https://doi.org/10.1016/j.ijmecsci.2024.109860

[25] Kalina K A et al. 2024 *Comp. Meth. App. Mech. Eng.* **421** 116739 URL https://doi.org/10.1016/j.cma.2023.116739

[26] Abdolazizi K P and Aydin R C and Cyron C J and Linka K 2025 *arXiv* URL https://doi.org/10.48550/arXiv.2502.05682

[27] Karabulut M 2025 *Buildings* **15**(3) 432 URL https://doi.org/10.3390/buildings15030432

[28] Raissi M, Perdikaris P and Karniadakis G 2019 *Journal of Computational Physics* **378** 686–707 ISSN 0021-9991 URL https://www.sciencedirect.com/science/article/pii/S0021999118307125

[29] Kollmannsberger S, D'Angella D, Jokeit M and Herrmann L 2021 *Deep Learning in Computational Mechanics* vol 977 (Cham: Springer International Publishing) ISBN 978-3-030-76586-6

[30] Braga-Neto U 2022 Characteristics-informed neural networks for forward and inverse hyperbolic problems URL http://arxiv.org/pdf/2212.14012

[31] Rodriguez-Torrado R, Ruiz P, Cueto-Felgueroso L, Green M C, Friesen T, Matringe S and Togelius J 2022 *Scientific reports* **12** 7557

[32] Coutinho E J R, Dall'Aqua M, McClenny L, Zhong M, Braga-Neto U and Gildin E 2023 *Journal of Computational Physics* **489** 112265 ISSN 00219991

[33] Samaniego E, Anitescu C, Goswami S, Nguyen-Thanh V M, Guo H, Hamdia K, Zhuang X and Rabczuk T 2020 *Computer Methods in Applied Mechanics and Engineering* **362** 112790 ISSN 00457825

[34] Mustajab A H, Lyu H, Rizvi Z and Wuttke F 2024 Physics-informed neural networks for high-frequency and multi-scale problems using transfer learning URL http://arxiv.org/pdf/2401.02810v2

[35] Rahaman N, Baratin A, Arpit D, Draxler F, Lin M, Hamprecht F A, Bengio Y and Courville A 2019 On the spectral bias of neural networks URL http://arxiv.org/pdf/1806.08734v3

[36] Rao S S 2019 *Vibration of Continuous Systems* (Wiley) ISBN 9781119424147

[37] Ramesh A and Utku S 1990 *Journal of Sound and Vibration* **139** 407–424 ISSN 0022-460X URL https://www.sciencedirect.com/science/article/pii/0022460X9090673N

[38] Hagedorn P and DasGupta A 2007 *Vibrations and Waves in Continuous Mechanical Systems* (Wiley) ISBN 9780470517383

[39] Abueidda D W, Lu Q and Koric S 2021 *International Journal for Numerical Methods in Engineering* **122** 7182–7201

[40] Hornik K 1991 *Neural Networks* **4** 251–257 ISSN 0893-6080 URL https://www.sciencedirect.com/science/article/pii/089360809190009T

[41] Kingma D P and Ba J 2014 Adam: A method for stochastic optimization URL http://arxiv.org/pdf/1412.6980v9

[42] Stender M, Tiedemann M, Spieler D, Schoepflin D, Hoffmann N and Oberst S 2021 *Mechanical Systems and Signal Processing* **149** 107181 ISSN 0888-3270 URL https://www.sciencedirect.com/science/article/pii/S0888327020305677

[43] Wulff P, Lentz L and von Wagner U 2023 *Acta Mechanica* **234**(5) 1973–1989

[44] Zamfirache I A, Precup R E and Petriu E M 2023 *Facta Universitatis* **21**(4) 615–630

[45] Mohaghegh F, Murthy J and Alam M R 2021 *Applied Ocean Research* **117** 102920 ISSN 0141-1187 URL https://www.sciencedirect.com/science/article/pii/S0141118721003886

[46] Hildebrand S and Klinge S 2024 *Neural Computing and Applications* **36** 16657-16682 URL https://doi.org/10.1007/s00521-024-10132-2

[47] Herrmann L and Kollmannsberger S 2024 *Computational Mechanics* ISSN 0178-7675

[48] Nguyen-Thanh V M, Zhuang X and Rabczuk T 2020 *European Journal of Mechanics - A/Solids* **80** 103874 ISSN 09977538

[49] He J, Chadha C, Kushwaha S, Koric S, Abueidda D and Jasiuk I 2023 *Acta Mechanica* **234**(4) 1365–1379 URL https://doi.org/10.1007/s00707-022-03449-3

[50] Zeng Q, Kothari Y, Bryngelson S H and Schäfer F 2022 Competitive physics informed networks URL https://arxiv.org/abs/2204.11144

[51] Radin N, Klinkel S and Altay O 2023 *PAMM* **23**

[52] Chadha C, He J, Abueidda D, Koric S, Guleryuz E and Jasiuk I 2023 *Acta Mechanica* **234** 5975–5998 ISSN 0001-5970

[53] Berrone S, Canuto C, Pintore M and Sukumar N 2023 *Heliyon* **9** e18820 ISSN 2405-8440

[54] Kleinberg R, Li Y and Yuan Y 2018 An alternative view: When does sgd escape local minima? URL http://arxiv.org/pdf/1802.06175v2

[55] You K, Long M, Wang J and Jordan M I 2019 How does learning rate decay help modern neural networks? URL http://arxiv.org/pdf/1908.01878

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

TIERED MULTI-SCALE DAMAGE ASSESSMENT OF CRITICAL INFRASTRUCTURE USING COHERENCE CHANGE DETECTION AND SAM-BASED DEEP LEARNING

Nadiia Kopiika[1,3,4], Sotirios Argyroudis[2,3,4], Jelena Ninic[1], Zehao Ye[1], and Stergios-Aristoteles Mitoulis[1,3,4]

[1] University of Birmingham, Edgbaston, Birmingham, B15 2TT, UK
e-mail: n.kopiika@bham.ac.uk; j.ninic@bham.ac.uk; zxy239@student.bham.ac.uk; s.a.mitoulis@bham.ac.uk

[2] Brunel University London, Kingston Lane, Uxbridge Middlesex, UB8 3PH, UK
e-mail: sotirios.argyroudis@brunel.ac.uk

[3]MetaInfrastructure.org, London, UK

[4]bridgeUkraine.org, London, UK

Abstract

Critical infrastructure like bridges supports transportation, accessibility, and economic flow by enabling movement. However, during conflicts and natural disasters, these vital assets are often targeted due to their strategic importance, causing major disruptions and costly recovery efforts. Traditional damage assessment methods, rely on manual inspection and on-site surveys, facing limitations in accessibility, especially in conflict zones, where safety risks and restricted access complicate evaluations. In these contexts, rapid, automated damage assessments are crucial to enable remote restoration planning and decision-making, ultimately fostering resilience. This research presents a multi-scale framework for the rapid assessment of damaged infrastructure assets, addressing the challenges of accessibility and timeliness in high-risk or remote areas. Leveraging diverse digital tools, including satellite imagery, crowdsourced data, and geospatial datasets, the framework enables scalable damage characterization by integrating Sentinel-1 SAR data, coherence change detection (CCD), and deep learning-based image segmentation for automated infrastructure damage assessment. A case study in Ukraine, involving 17 bridges damaged during conflict, demonstrates the framework's adaptability to challenging environments. This approach enhances the speed and reliability of the initial damage assessments, supporting data-driven decision making for infrastructure restoration prioritization and investment, which are essential to resilience-building efforts in post-disaster recovery.

Keywords: critical infrastructure, automated damage assessment, remote sensing, data fusion, deep learning, coherence change detection

1 INTRODUCTION

Critical infrastructure, such as bridges, is vital for connectivity, economic flow, and accessibility, enabling the movement of people and goods, promoting social cohesion, and ensuring access to essential services, including emergency facilities [1],[2]. Maintaining and monitoring their condition is crucial for uninterrupted operation. In conflict situations, bridges are often targeted due to their strategic importance, and damage can severely affect functionality, leading to high restoration costs and traffic disruptions [3],[4].

Preliminary damage assessment is necessary to facilitate informed decision-making and recovery planning [5]. Traditional infrastructure inspections rely on costly, time-consuming, and often risky manual surveys, which become impractical in inaccessible areas after major disasters (e.g. in conflict- affected regions) [6]. Moreover, large-scale damage requires regional assessments, but conventional methods fail to account for asset interdependencies, delaying recovery [7],[8]. To address this, integrating digital tools, such as GPS, InSAR, IoT, UaV, DIC, and others, enables multi-scale assessments, bridging gaps between macro-level regional evaluations and micro-level asset inspections [9],[10],[11],[12]. Satellite imagery has emerged as a key tool for remote damage assessment, but its integration across different scales remains limited. In the case of bridges, remote sensing and computer vision-based methods aid in assessing structural deflections, soil settlements, cracks, and corrosion, enhancing decision-making for restoration [13],[14].

Earth Observation (EO) technologies, particularly Synthetic-Aperture Radar (SAR), offer non-invasive monitoring for post-disaster damage characterization [15]. InSAR data, combined with Coherent Change Detection (CCD), detects structural changes by comparing pre- and post-event images [16],[17],[19]. While existing studies focus on natural disasters, conflict-related damage presents unique challenges due to scattered destruction in dense urban environments, making automated damage identification difficult [18]. High-resolution satellite data, often unavailable in conflict zones due to security concerns, further complicates assessments. Low-resolution imagery and open data sources, such as crowdsourcing and OpenStreetMap, can help mitigate these limitations [19],[20].

Deep learning and computer vision enable automated infrastructure assessments, reducing reliance on manual inspections [21]. Convolutional neural networks (CNNs) facilitate damage classification and detection, while techniques like Grad-CAM improve localization [22]. Social media and crowdsourced images provide real-time insights, complementing satellite and UAV-based assessments [23]. Advanced semantic segmentation models, such as Segment Anything Model (SAM), enhance image preprocessing for damage identification. Deep learning-based approaches, including Bidirectional Feature Pyramid Networks and Mask R-CNN, achieve high accuracy in structural health monitoring, improving efficiency in post-disaster decision-making [24].

Despite advancements in remote sensing and AI-driven damage assessment, a unified framework integrating multi-scale damage characterization remains absent. Traditional methods struggle to assess extensive bridge damage, particularly in conflict zones where accessibility is severely limited [6]. Effective assessment must consider multiple factors, including damage type, extent, accessibility, interdependencies with other assets, and resource availability, all of which influence recovery strategies. However, current research is often fragmented, focusing on either regional (macro-) or asset-level (meso-) assessments without integrating these perspectives into a cohesive engineering framework. This gap hinders a comprehensive understanding of how infrastructure damage impacts regional operability and vice versa.

In order to fill this gap, we propose a multi-scale framework for rapid, automatic damage characterization of critical infrastructure in conflict-prone regions, which is in more detail de-

scribed in recent publication [16]. The approach combines InSAR CCD to evaluate damage levels with open-source satellite imagery, crowdsourced images, and records from open access databases for validation. Demonstrated in a post-conflict Ukrainian case study, this framework enhances accuracy and efficiency in bridge damage assessment, supporting timely, data-driven restoration efforts.

2 TIERED APPROACH FOR DAMAGE ASSESSMENT

2.1 General multi-scale framework and data

Figure 1 presents a tiered framework for multi-scale damage characterisation, integrating disparate open-access data sources to enable rapid and informed decisions for infrastructure restoration. This approach combines **regional (R)**, **asset (A)**, and **component (C)** scale assessments to address the challenges of damage detection in conflict-affected regions. By leveraging multiple data sources, including remote sensing technologies, satellite imagery, records from open access databases, and crowdsourced information, the framework offers a comprehensive understanding of damage across different scales.

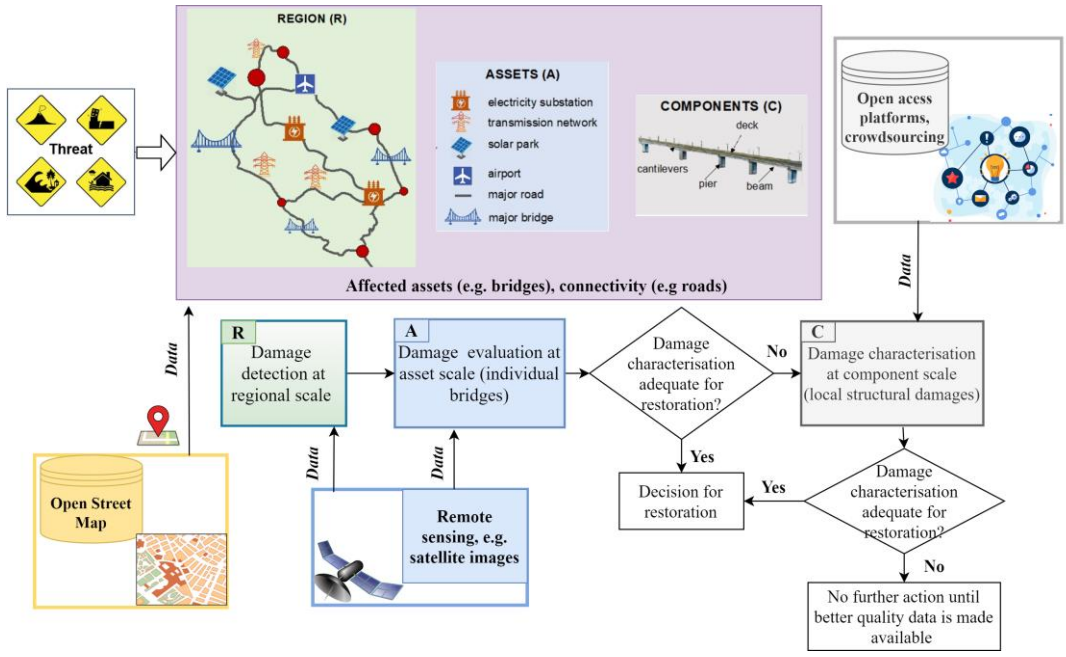


Figure 1: Framework for damage assessment at different scales: regional (R), asset (A) and component (C) toward decisions for restoration.

Each data source plays a distinct but complementary role in the damage characterisation process. Regional-scale assessments provide a broad overview of affected areas, which is refined and validated using asset-level data to capture more detailed information. When necessary, component-scale data offers high-resolution insights into specific structural elements, filling critical gaps left by broader-scale assessments. In this way, the proposed multi-scale integration enhances the accuracy and reliability of damage detection, which is of crucial importance to facilitate decision-making for restoration strategies. By integrating regional, asset, and component-level information this research ensures that both broad network-level vulnerabilities and detailed structural damage are captured, enabling a robust and scalable approach for post-disaster recovery planning [16].

2.2 Damage assessment at regional and asset scales

Damage characterisation at the regional and asset scale involves a multi-phase approach using open-access data and remote sensing technologies (See Figure 2). The primary data sources include Sentinel-1 Single Look Complex (SLC) imagery [25],[26] and crowdsourced data, such as OpenStreetMap (OSM) [27], to indicate the particular timeframe of the disaster, which led to infrastructure damage in the area.

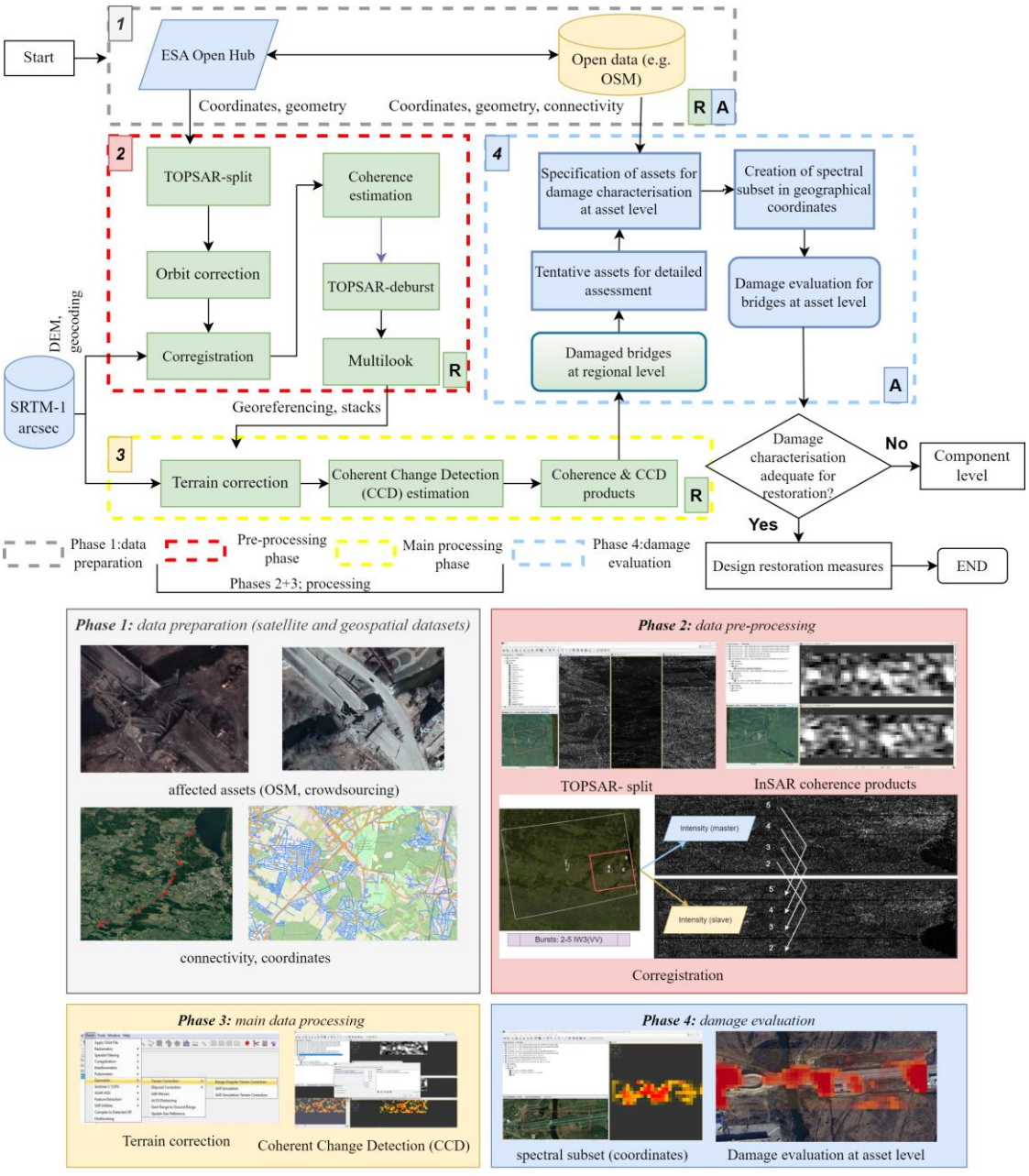


Figure 2: Workflow for damage evaluation at regional (R) and asset (A) scale based on four phases

The process begins with **Phase 1**, where Sentinel-1 SAR images are acquired in both ascending and descending geometries from open-access platforms like ESA Open Hub [25]. Concurrently, preliminary geographic coordinates of affected assets are identified using OSM [27] and crowdsourcing platforms [28],[29], which report disruptions in transport routes and

asset damage. Crowdsourced data address potential limitations of OSM by providing up-to-date information during hazard events. In **Phase 2**, Sentinel-1 SAR images undergo preprocessing using the Sentinel Application Platform (SNAP). This involves splitting images into sub-swaths, applying orbit correction using precise orbit files, and co-registering the images with the Digital Elevation Model (DEM) [30] of the Shuttle Radar Topography Mission (SRTM-1 arcsec)[31]. Interferometric pairs are generated from images taken before and after the hazard, followed by coherence (γ) estimation to assess changes in radar reflections, which serve as an indicator of structural stability (see Equation 1).

$$\gamma = \frac{E\left[u_1 u_2^*\right]}{\sqrt{E\left[\left|u_1\right|^2\right]}\sqrt{E\left[\left|u_2\right|^2\right]}}, \tag{1}$$

where u_1 and u_2 indicate the complex correlation coefficient between two SAR scenes, $E\{\}$ represents the mathematical expectation and * is the complex conjugate operator.

Phase 3 focuses on Coherent Change Detection (CCD) to identify damage. CCD involves calculating coherence values for two pairs of SAR images, —one pair taken before the event,— $\gamma(pre)$ and another before and after the event,—$\gamma(post)$ (Equation 2).

$$CCD = \gamma(pre) - \gamma(post). \tag{2}$$

Coherence values range from 0 (low stability) to 1 (high stability). A decrease in coherence between pre- and post-event images indicates significant changes in the built environment, suggesting damage [34]. CCD values, which range from -1 to 1, are used to quantify these changes. Positive CCD values indicate substantial differences and potential damage, while values near zero represent stable areas.

Finally, in **Phase 4**, coherence and CCD products are integrated into Geographic Information Systems (GIS) such as ArcGIS or QGIS. This integration visualises areas with potential damage, providing a semi-automatic means of identifying and prioritising locations for further inspection and repair. Assets of interest are further cross-validated using open data sources like Google Maps and Sentinel-1 imagery. The results are exported in Well-Known Text (WKT) format for GIS-based illustration and damage classification. Different CCD ranges correspond to varying damage levels, which are classified according to engineering criteria to support restoration planning.

2.3 Damage assessment at component scale

When regional or asset-scale assessments are insufficient for making restoration decisions, a more detailed component-scale assessment becomes essential. This involves detecting, localising, and classifying damage at the structural component level using advanced semantic segmentation methods (See Figure 3).

For selected assets additional visual information is gathered from open platforms such as The Eyes on Russia [28] and UADamage [29]. These images, showing visible structural damage, are processed using selected computer vision (CV) techniques for automatic damage detection. The process consists of two main steps:

- Component Segmentation – Identifying specific structural components (e.g., bridge deck, pier).
- Instance Segmentation – Classifying and localising the type of damage (e.g., cracks or deformations).

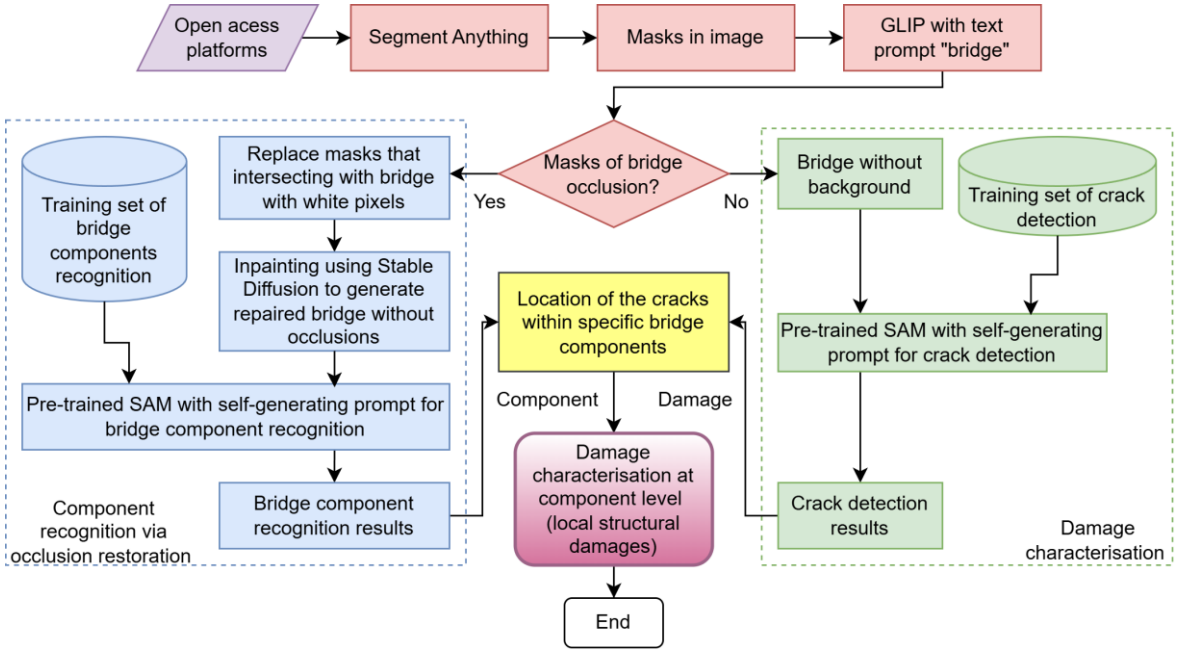


Figure 3: Flowchart, illustrating the damage characterisation method at component scale

Since some images may have limited resolution or obstructions, image pre-processing is employed to enhance quality and remove occlusions. Large pre-trained models, such as Grounded Language-Image Pre-training (GLIP) [35],[36] and Contrastive Language-Image Pre-Training (CLIP)[37], improve image clarity and facilitate more accurate detection. For instance segmentation, the Segment Anything Model (SAM)[38] is used to generate precise masks for the detected components.

In this automated workflow, SAM's "anything mode" detects all potential masks in an image, with labels assigned to each mask. Masks labeled as "bridge" by GLIP are isolated for further analysis. If occlusions are detected, the stable diffusion model is applied for image inpainting to restore hidden sections of the structure. This process generates context-consistent repair content and reconstructs the missing parts of the bridge or other components.

Damage Characterisation follows once occlusions are resolved. SAM, enhanced with a self-generating prompt mechanism, automatically detects damage such as cracks or deformations. The final output integrates component recognition and damage characterisation to pinpoint the exact location and extent of the damage within each structural component.

3 APPLICATION: CONFLICT CASE STUDY

Assessing infrastructure damage in inaccessible regions like Ukraine presents significant challenges, particularly due to ongoing hostilities. Missile strikes and artillery fire have caused widespread destruction, severely damaging over 345 bridges across the region, with the Kyiv region being one of the most affected [39]. Bridges along the Irpin River, crucial for logistics, supply routes, and evacuation corridors, have sustained substantial damage, disrupting critical connections such as Bucha-Kyiv, Hostomel-Kyiv, and Irpin-Kyiv routes [40]. This case study focuses on validating the proposed framework by identifying and characterising damage to selected bridges along the Irpin River. Given the inaccessibility of these areas, the framework enables remote damage detection and classification, accelerating decision-making for restoration efforts and enhancing the region's resilience.

The assessment is conducted at both regional, asset and component scales, following the methodology outlined in Section 2. At the regional scale, open geospatial data from OpenStreetMap (OSM) is used to identify critical river-crossing bridges essential for maintaining connectivity. These bridges are geometrically validated using Google Satellite Imagery and high-resolution optical images, resulting in 17 bridges selected for further evaluation.

At the asset scale, damage detection is performed using Sentinel-1 SAR Single Look Complex (SLC) images during the period of extensive destruction between February and March 2022. The Coherent Change Detection (CCD) method is employed to assess damage, with coherence values from image pairs analyzed to determine damage severity. Each asset is classified according to the Level of Knowledge (LoK), which reflects the reliability of results based on coherence values (see Figure 4). Assets with coherence in the first image pair below 0.5 ($\gamma(pre)<0.5$), indicated low-resolution data and unreliable results, are excluded from the evaluation. High coherence values (above 0.7) indicate high reliability and allow for a more accurate damage assessment.

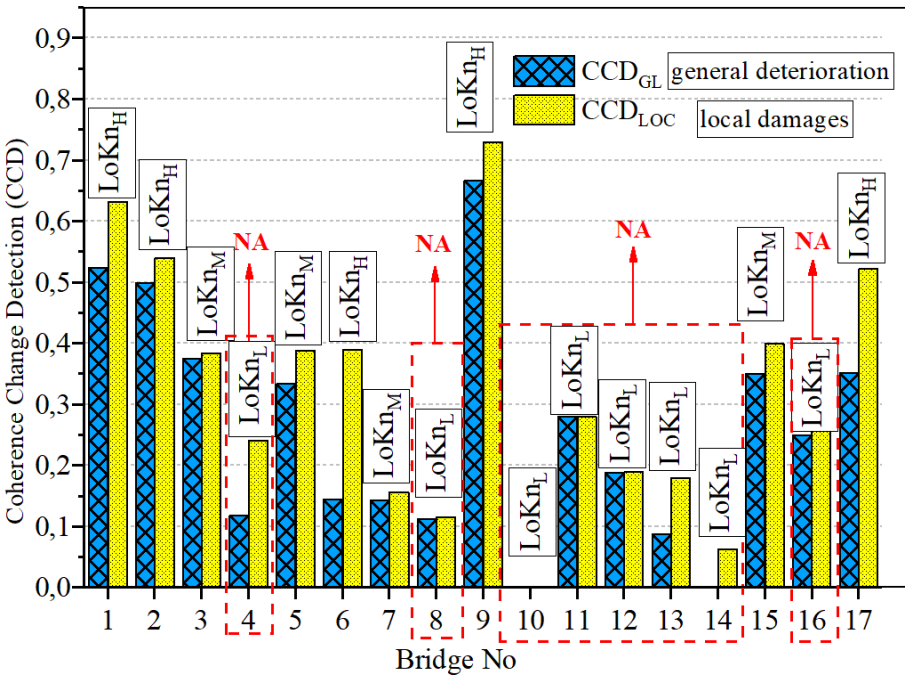


Figure 4: Coherent Change Detection (CCD) and Level of Knowledge (LoK) for the analyzed bridges. CCD_{LOC} indicates local damage within the asset area (maximum CCD); CCD_{GL} indicates mean deterioration (average CCD).

Damage levels are categorized as low (DL_L), moderate (DL_M), or high (DL_H) based on CCD values at both local (CCD_{LOC}) and global (CCD_{GL}) scales (see Table 1).

DL	CCD_{LOC} (Max)	CCD_{GL}	Description
DL_H (High)	0.5-1.0	0.4-1.0	Severe/Complete damage: Complete destruction of the structure or some of its components.
DL_M (Moderate)	0.35-0.5	0.3-0.4	Moderate/Extensive damage: Considerable damage in some of the components.
DL_L (Low)	0.0-0.35	0.0-0.3	No/Minor damage: General deterioration, signs of slight damage.

Table 1: Damage characterisation of infrastructure assets (bridges) based on CCD.

Assets are assigned an index corresponding to their damage level, based on approximate CCD value ranges. Despite limitations related to satellite spatial resolution, line of sight, and event sequence, the methodology proves effective for damage evaluation when access to the affected area is restricted.

Figure 5 illustrates an example of damage assessment at asset scale for Bridge No 1 (high reliability of data, -LoK_H and high level of damage, -DL_H). Alignment of the CCD results integrated into a GIS environment with high-resolution imagery from Google Earth Pro further validates damage levels (Figure 5, b). This multi-source approach supports reliable damage assessment and restoration planning, minimizing downtime, optimizing traffic flow, and facilitating stakeholder coordination for rapid recovery.

(a)

(b)

Figure 5: Damage characterisation at asset scale for B1(LoK_H; DL_H): (a) assessment with CCD (CCD_{GL}=0.523; CCD_{LOC}=0.623); (b) high-resolution imagery from Google Earth Pro (on the right a temporary diversion route was constructed to bypass the bridge)

For bridge B1, classified with high reliability (LoK_H) and severe damage (DL_H), an automated component-scale damage assessment was performed using crowdsourced images from open-access platforms [28],[29]. Following the methodology outlined in Section 2.3, two computer vision tasks were conducted: instance segmentation for component detection and classification, and instance segmentation for identifying, locating, and classifying defects such

as cracks. The results, shown in Figure 6, were obtained using the Segment Anything Model (SAM) and Grounded-SAM.

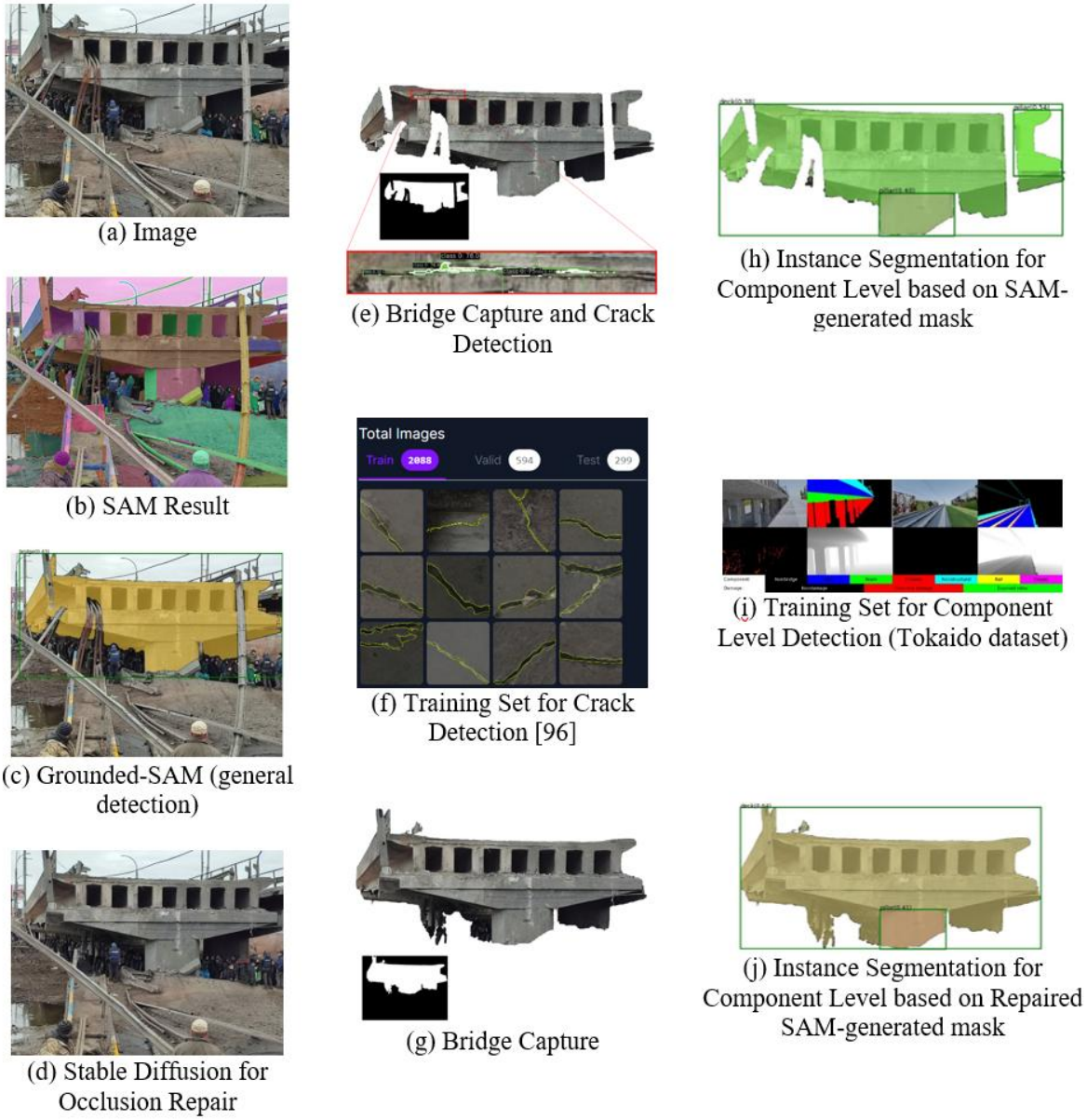


Figure 6: Combined component and damage detection for B1 bridge

SAM first extracted masks in "everything mode," generating visual segmentations for the bridge components. Grounded-SAM then matched labels to the masks and filtered out low-recognition results, providing more refined segmentation. The process faced two primary challenges: the complexity of the background and occlusion that partially obscured the bridge. Grounded-SAM addressed these issues by conducting an initial general detection to isolate the bridge mask, followed by background removal and occlusion repair using Stable Diffusion. This process removed non-relevant details and restored the visibility of the bridge structure. Once the occlusions were repaired, a crack detection task was performed using a dedicated model trained on approximately 3,000 images, leveraging the query-based learnable prompt SAM algorithm. Bridge component detection was then carried out using a self-prompting SAM model trained on the Tokaido dataset, resulting in significantly improved de-

tection accuracy [41],[42]. After completing crack detection and component recognition, a matching process was used to identify the precise location of cracks within specific bridge components. For instance, a critical crack was detected on a pier, a key structural element, providing clear evidence of severe deterioration. This component-scale approach, supported by advanced computer vision technologies, demonstrated exceptional capability in enhancing automatic damage detection. It aligns with the multi-scale framework presented in this study and offers a valuable tool for improving damage characterization and restoration planning.

4 CONCLUSIONS

This research presents a multi-scale framework for automated infrastructure damage assessment at regional, asset, and component scales using openly available data. The framework addresses the lack of automated multi-scale damage characterization methods and aims to improve potential for use of these cutting-edge digital technologies for post-disaster recovery for regions affected by conflicts or natural disasters. By applying this framework to damaged infrastructure, we demonstrate its ability to automatically and accurately assess damage at various scales. Damage at the regional and asset levels is assessed using a new proxy metric based on interferometric coherence change detection (CCD). Depending on the Level of Knowledge (LoK), which reflects data reliability, CCD values represent local (CCD_{LOC}) or global (CCD_{GL}) damage and are categorized into three distinct damage levels, validated with high-resolution imagery. At the component scale, advanced computer vision techniques are employed to detect and localize defects in individual bridge components by extracting and repairing occluded areas.

This approach supports rapid assessment and decision-making for critical infrastructure reconstruction by providing crucial information on the extent of damage. The framework is especially valuable for inaccessible areas, such as war zones or regions affected by floods and other climate hazards. Limitations of the methodology include spatial resolution constraints of Sentinel-1 images, challenges related to event timing, and line-of-sight issues. However, integrating multiple data sources significantly enhances the accuracy and feasibility of the approach. Future work will incorporate higher-resolution optical satellite data, such as Sentinel-2 and PlanetScope, for more detailed inspections of surface conditions. Integrating automated damage assessments with open-access data enables a more informed response to hazards, accelerating decision-making and optimizing recovery planning and restoration strategies.

ACKNOWLEDGEMENTS

Stergios-Aristoteles and Sotirios Argyroudis received funding by the UK Research and Innovation (UKRI) under the UK government's Horizon Europe funding guarantee [Ref: EP/Y003586/1, EP/X037665/1]. This is the funding guarantee for the European Union HORIZON-MSCA-2021-SE-01 [grant agreement No: 101086413] Re-Charged - Climate-aware Resilience for Sustainable Critical and interdependent Infrastructure Systems enhanced by emerging Digital Technologies. Nadiia Kopiika would like to acknowledge of support received from the British Academy for this research (Award Reference: RaR\100770).

REFERENCES

[1] R. Sacks, A. Kedar, A. Borrmann, L. Ma, I. Brilakis, P. Hüthwohl, S. Daum, U. Kattel, R. Yosef, T. Liebich, B.E. Burcu Barutcu, S. Muhic. SeeBridge as next generation bridge inspection: Overview, information delivery manual and model view definition. *Automation in Construction*, **90**, 134-145, 2018. https://doi.org/10.1016/j.autcon.2018.02.033

[2] S.-A. Mitoulis, M. Domaneschi, J. R. Casas. Bridges are more than simple connections in transport networks and they must consider their impact on the resilience of networks, systems and regions. *Proceedings of the Institution of Civil Engineers-Bridge Engineering*, **177(3)**, 137-138, 2024. https://doi.org/10.1680/jbren.2024.177.3.137

[3] S. A. Argyroudis, G. Nasiopoulos, N. Mantadakis, S A. Mitoulis. Cost-based resilience assessment of bridges subjected to earthquakes, *International Journal of Disaster Resilience in the Built Environment*, **12(2)**, 209-222, 2020. https://doi.org/10.1108/IJDRBE-02-2020-0014

[4] N. Kopiika, R. Di Bari, S. Argyroudis, J. Ninic, S.A. Mitoulis. Sustainability and resilience-driven prioritisation for restoring critical infrastructure after major disasters and conflict. *Transportation Research Part D: Transport and Environment*, **139**, 104592, 2025. http://dx.doi.org/10.1016/j.trd.2025.104592

[5] V. Macchiarulo, G. Giardina, P. Milillo, Y. D. Aktas, M. R. Z. Whitworth. Integrating post-event very high-resolution SAR imagery and machine learning for building-level earthquake damage assessment. *Bull Earthquake Eng*, 2024. https://doi.org/10.1007/s10518-024-01877-1

[6] S.A. Mitoulis, S.A. Argyroudis, M. Panteli, C. Fuggini, S. Valkaniotis, W. Hynes, I. Linkov. Conflict resilience framework for critical infrastructure peacebuilding. *Sustainable Cities and Society*, **91**, 104405, 2023. https://doi.org/10.1016/j.scs.2023.104405

[7] Y. P. Fang, G. Sansavini. Optimum post-disruption restoration under uncertainty for enhancing infrastructure resilience. *Reliab Eng Syst Saf*, **185**, 1-11, 2019. https://doi.org/10.1016/j.ress.2018.12.002

[8] D. V. Achillopoulou, S. A. Mitoulis, S. A. Argyroudis, Y. Wang. Monitoring of transport infrastructure exposed to multiple hazards: A roadmap for building resilience. *Science of the Total Environment*, **746**, 141001, 2020. https://doi.org/10.1016/j.scitotenv.2020.141001

[9] M. Koohmishi, S. Kaewunruen, L. Chang, Y. Guo. Advancing railway track health monitoring: Integrating GPR, InSAR and machine learning for enhanced asset management. *Automation in Construction*, **162**, 105378, 2024. https://doi.org/10.1016/j.autcon.2024.105378

[10] V. Macchiarulo, P. Milillo, C. Blenkinsopp, G. Giardina. Monitoring deformations of infrastructure networks: A fully automated GIS integration and analysis of InSAR time-series. *Structural Health Monitoring*, **21(4)**, 1849-1878, 2022. https://doi.org/10.1177/14759217211045912

[11] N. Kopiika, Y. Blikharskyy. Digital Image Correlation for Assessment of Bridges' Technical State and Remaining Resource. *Structural Control and Health Monitoring*, **1763285**, 1545-2255, 2024. https://doi.org/10.1155/2024/1763285

[12] S.A. Argyroudis, S.A. Mitoulis, E. Chatzi, J. W. Baker, D. Frangopol, I. Linkov. Digital technologies can enhance climate resilience of critical infrastructure. *Climate Risk Management*, **35**, 100387, 2022. https://doi.org/10.1016/j.crm.2021.100387

[13] G.A. Papadopoulos, A. Agalos, A. Karavias, I. Triantafyllou, I. Parcharidis, E. Lekkas. Seismic and Geodetic Imaging (DInSAR) Investigation of the March 2021 Strong Earthquake Sequence in Thessaly, Central Greece. *Geosciences*, **11(8)**:311, 2021. https://doi.org/10.3390/geosciences11080311

[14] S. Mavroulis, I. Triantafyllou, A. Karavias, M. Gogou, K-N. Katsetsiadou, E. Lekkas, G. A. Papadopoulos, I. Parcharidis. Primary and Secondary Environmental Effects Triggered by the 30 October 2020, Mw = 7.0, Samos (Eastern Aegean Sea, Greece) Earthquake Based on Post-Event Field Surveys and InSAR Analysis. *Applied Sciences*; **11(7):**3281, 2021. https://doi.org/10.3390/app11073281

[15] T. Smail, M. Abed, A. Mebarki, M. Lazecky. Earthquake-induced landslide monitoring and survey by means of InSAR. *Natural Hazards and Earth System Sciences*, **22(5)**, 1609-1625, 2022. https://doi.org/10.5194/nhess-22-1609-2022

[16] N. Kopiika, A. Karavias, P. Krassakis, Z. Ye, J. Ninic, N. Shakhovska, S. Argyroudis, S.A. Mitoulis. Rapid post-disaster infrastructure damage characterisation using remote sensing and deep learning technologies: A tiered approach. *Automation in Construction*, **170**, 105955, 2025. https://doi.org/10.1016/j.autcon.2024.105955

[17] N. Kopiika, J. Ninic, S.A. Mitoulis, S. Damage characterisation using Sentinel-1 images: Case study of bridges in Ukraine. *In IABSE Symposium Manchester 2024: Construction's Role for a World in Emergency*, Manchester, UK, 10-12 March, 2024, 367-375. https://doi.org/10.2749/manchester.2024.0367

[18] Y. Wang, A. W. Z. Chew, L. Zhang. Building damage detection from satellite images after natural disasters on extremely imbalanced datasets. *Automation in construction*, **140**, 104328, 2022. https://doi.org/10.1016/j.autcon.2022.104328

[19] S.A. Mitoulis, N. Kopiika, J. Ninic, A. Karavias, P. Krassakis, N. Koukouzas, N. Shakhovska, S. Argyroudis. Damage characterisation using stand-off observations to enable recovery: the case of infrastructure affected by targeted attacks, *Presented at IABSE Congress: Beyond Structural Engineering in a Changing World, San José, Cost Rica*, 25-27 September 2024, published in IABSE Congress San José 2024, 785-791, https://doi.org/10.2749/sanjose.2024.0785

[20] D. Wen, X. Huang, F. Bovolo, J. Li, X. Ke, A. Zhang, J.A. Benediktsson. Change detection from very-high-spatial-resolution optical remote sensing images: Methods, applications, and future directions. *IEEE Geoscience and Remote Sensing Magazine*, **9(4)**, 68-101, 2021. https://doi.org/10.1109/MGRS.2021.3063465

[21] Jr. B. F. Spencer, V. Hoskere, Y. Narazaki. Advances in computer vision-based civil infrastructure inspection and monitoring. *Engineering*, **5(2)**, 199-222, 2019. https://doi.org/10.1016/j.eng.2018.11.030

[22] P. Gong, W. Zhang, L. Yu, C. Li, J. Wang, L. Liang, X. Li, L. Ji, Y. Bai. New research paradigm for global land cover mapping. *National Remote Sensing Bulletin*, **20(5)**, 1002-1016, 2016. https://doi.org/10.11834/jrs.20166138

[23] W.K. Chong, H. Naganathan, H. Liu, S. Ariaratnam, J. Kim. Understanding infrastructure resiliency in Chennai, India using Twitter's Geotags and texts: a preliminary study. *Engineering*, **4(2)**, 218-223, 2018. https://doi.org/10.1016/j.eng.2018.03.010

[24] F. Du, S. Jiao, K. Chu. Application research of bridge damage detection based on the improved lightweight convolutional neural network model. *Applied Sciences*, **12**, 6225, 2022. https://doi.org/10.3390/app12126225

[25] Mission ends for Copernicus Sentinel-1B satellite. Available at: https://surl.li/djkjmv [accessed 3 February 2025].

[26] R. Torres, P. Snoeij, D. Geudtner, D. Bibby, M. Davidson, E. Attema, P. Potin, B. O. Rommen, N. Floury, M. Brown, I. N. Traver, P. Deghaye, B. Duesmann, B. Rosich, N. Miranda, C. Bruno, M. L'Abbate, R. Croci, A. Pietropaolo, M. Huchler, F. Rostan. GMES Sentinel-1 mission. *Remote Sensing of Environment*, 120, 9–24, 2012. https://doi.org/10.1016/j.rse.2011.05.028 .

[27] OpenStreetMap. Available online: https://surl.li/apjanc [accessed 3 February 2025].

[28] Online source: The Eyes on Russia. The Centre for Information Resilience (CIR). https://surl.li/qvkngs [accessed 3 October 2023]

[29] Online source: UADamage https://surl.li/althjo [accessed 3 February 2025]

[30] Science Toolbox Exploitation Platform: https://surl.li/gqyeyx [accessed 3 February 2025]

[31] USGS EROS Archive—Digital Elevation—Shuttle Radar Topography Mission (SRTM) 1 Arc-Second Global. U.S. Geological Survey. Available online: https://surl.li/aquhxs [accessed 3 February 2025].

[32] D. Closson, N. Milisavljevic. InSAR Coherence and Intensity Changes Detection. *Mine Action-The Research Experience of the Royal Military Academy of Belgium*, 292, 2017. https://doi.org/10.5772/65779

[33] S. Stramondo, C. Bignami, M. Chini, N. Pierdicca, A. Tertulliani. Satellite radar and optical remote sensing for earthquake damage detection: results from different case studies. *International Journal of Remote Sensing*, 27(20), 4433-4447, 2006. https://doi.org/10.1080/01431160600675895

[34] T. ElGharbawi, F. Zarzoura. Damage detection using SAR coherence statistical analysis, application to Beirut, Lebanon. *ISPRS Journal of Photogrammetry and Remote Sensing*, **173**, 1-9, 2021. https://doi.org/10.1016/j.isprsjprs.2021.01.001

[35] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, Y. Jie, C. Li, J. Yang, H. Su, J. Zhu, L. Zhang. Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. arXiv:2303.05499, 2023; https://doi.org/10.48550/arXiv.2303.05499

[36] L. Li, P. Zhang, H. Zhang, J. Yang, C. Li, Y. Zhong, L. Wang, L. Yuan, L. Zhang, J. Hwang, K. W. Chang, J. Gao. Grounded Language-Image Pre-training. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR*), 10965-10975, 2022. https://doi.org/10.48550/arXiv.2112.03857

[37] Bach Khoa Ho Chi Minh University. Crack detection using instance segmentation in yolov8 dataset: https://surl.li/ztehsr [accessed 3 February 2025]

[38] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W. Y. Lo, P. Dollár, R. Girshick. Segment anything. In Proceedings of the IEEE/CVF International Conference on Computer Vision, 4015-4026, 2023. https://doi.org/10.1109/ICCV51070.2023.00371

[39] Television news service. http://surl.li/lzmkl [accessed 3 February 2025]

[40] Daniel Rice. Ukrainian Bridges are Playing a Vital Role in Both the Defense and the Offense in the Ukrainian-Russian War https://surl.li/lfddaj [accessed 3 February 2025]

[41] K. Chen, C. Liu, H. Chen, H. Zhang, W. Li, Z. Zou, Z. Shi. RSPrompter: Learning to prompt for remote sensing instance segmentation based on visual foundation model. *IEEE Transactions on Geoscience and Remote Sensing*, 62 (4701117), 1-17, 2024. https://doi.org/10.1109/TGRS.2024.3356074

[42] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, B. Ommer. High-resolution image synthesis with latent diffusion models. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022; 10684-10695. https://doi.org/10.1109/CVPR52688.2022.01042

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

SENSITIVITY ANALYSIS OF IMAGE CLASSIFICATION MODELS USING GENERALIZED POLYNOMIAL CHAOS

Lukas Bahr[1,2], Lucas Poßner[2,5], Konstantin Weise[3,4], Stephan Matzka[5], Sophie Gröger[6] and Rüdiger Daub[7,8]

[1] TUM School of Engineering and Design, Technical University of Munich, Munich, Germany.
e-mail: lukas.bahr@tum.de

[2] Digitization Department for Production, BMW Group, Germany.

[3] Institute for Electrical Power Engineering, Leipzig University of Applied Sciences, Leipzig, Germany.

[4] Methods and Development Group "Brain Networks", Max Planck Institute for Human Cognitive and Brain Sciences, Leipzig, Germany.

[5] Professorship for Mechatronics and Mechatronic Systems, HTW Berlin, Berlin, Germany.

[6] Professorship of Production Measurement Technology, Technical University of Chemnitz, Chemnitz, Germany.

[7] Institute for Machine Tools and Industrial Management, Technical University of Munich, Munich, Germany.

[8] Institute for Casting, Composite and Processing Technology (IGCV), Fraunhofer, Augsburg, Germany.

Abstract. *Integrating advanced communication protocols in production has accelerated the adoption of data-driven predictive quality methods, notably machine learning (ML) models. However, ML models in image classification often face significant uncertainties arising from model, data, and domain shifts. These uncertainties lead to overconfidence in the classification model's output. To better understand these models, sensitivity analysis can help to analyze the relative influence of input parameters on the output. This work investigates the sensitivity of image classification models used for predictive quality. We propose modeling the distributional domain shifts of inputs with random variables and quantifying their impact on the model's outputs using Sobol indices computed via generalized polynomial chaos (GPC). This approach is validated through a case study involving a welding defect classification problem, utilizing a fine-tuned ResNet18 model and an emblem classification model used in BMW Group production facilities.*

Keywords: Sensitivity Analysis, Generalized Polynomial Chaos, Image Classification, Data-Driven Predictive Quality.

1 INTRODUCTION

Predictive quality has prevailed in the manufacturing industry thanks to the approaches defined in ISO / IEC 31010 [1] and include approaches such as failure mode and effects analysis (FMEA) [2], or fault tree analysis [3]. Recent advancements in large language models have enhanced their ability to reason with symbolic data, thereby maintaining the importance of these methods for risk assessment [4]. The rise of the industrial internet of things has enhanced operational efficiency by integrating advanced communication protocols. This development fosters a focus on data-driven predictive quality methods in production processes, such as the application of machine learning (ML) [5, 6]. Methods such as anomaly detection or classification algorithms can help improve product quality and reliability by finding patterns in data that are not immediately obvious to humans [7, 8]. In [9], the authors explore common approaches to ML that address predictive quality. Particular image classification models for predicting defects (nOK/OK), such as convolutional neural networks, are dominant in the manufacturing domain.

Understanding the sensitivity of image classification models is of great interest. Sensitivity analysis (SA) studies how the uncertainty of a model's output can be attributed to the uncertainty of its inputs [10]. This understanding is crucial because many models, especially those provided by external suppliers, function as black boxes, making it difficult to understand the inner workings. For instance, consider an image classification model that predicts the quality of a weld seam. Image classification models are often poorly calibrated and overconfident on the training data [11]. Thus, evaluating the sensitivity is essential to determine how changes in input, such as varying lighting conditions in the production environment, affect the model's reliability.

The challenges for analyzing the sensitivity can be summarized as follows [12, 13]:

1. Many ML models are black-box models, so we cannot explain the predictions given the inputs (model uncertainty).
2. Errors in the inputs that arise from, for example, inadequate measurements introduce uncertainty in the output (data uncertainty).
3. Distributional domain shifts in the inputs that arise from, e.g., changes in the production environment, introduce uncertainty in the output (domain uncertainty).

This paper investigates the sensitivity of image classification models to perturbations in the input data. We propose applying geometric transformations to the input images, such as changes in rotation or tilt, or changes on the image channel, such as brightness, treating these variables as uncertain inputs. We utilize generalized polynomial chaos (GPC) to approximate a surrogate model of the image classification model's input variables on the output [14]. The surrogate model is then employed to compute the Sobol indices, which quantify the sensitivity with respect to the perturbations and their joint interactions.

In Section 2, we recall the main ideas behind SA. In Section 3, we introduce a method to quantify the sensitivity of perturbations on the input image with respect to the output of the classification model. Section 4 validates the approach through two case studies: (i) a welding seam defect classification problem, utilizing a fine-tuned ResNet18 model and (ii) an actual image classification model for emblem detection employed in BMW Group's production facilities. We conclude with the discussion in Section 5.

2 SENSITIVITY ANALYSIS

SA aims to quantify the influence of input variables on a model's output [15, 10]. Let $\boldsymbol{y} = f(\boldsymbol{x})$ be a mathematical model with an input vector $\boldsymbol{x} = (x_1, \cdots, x_n)$ of size n and an output vector $\boldsymbol{y} = (y_1, \cdots, y_m)$ of size m. We are interested in quantifying the relative importance of how changes in each input x_i influence y_j, expressed through relative sensitivity indices [16]. Various methods have been developed to analyze a model's sensitivity. The most prominent methods are clustered into local and global SA.

Local SA methods aim to quantify the influence of individual input features on the model's output by changing specific input values while keeping other inputs constant [17]. Local SA approaches are characterized by their low computational cost and ease of application. However, they fail to fully represent nonlinear models and interactions between inputs [18]. In contrast, global SA methods aim to simultaneously assess the influence of all input variables on the model's output, taking parameter interactions into account [15]. The oldest global SA methods include regression-based techniques, wherein the regression coefficients are learned from the input variable's influence on the output. The sensitivity can be quantified using measures such as the Pearson correlation coefficient [10].

However, ML models for image classification are typically nonlinear, allowing the network to learn more complex representations of the input data. For these nonlinear models, polynomial chaos methods, such as GPC, can be used to approximate a surrogate model to the input-output data, enabling the computation of sensitivity coefficients through easier-to-interpret models [19]. The sensitivity coefficients can then be computed using variance-based methods. Variance-based methods, such as Sobol indices, decompose the variance of the output into the contributions of each input variable. This approach acknowledges that different input variables and their interactions contribute variably to the model's output variations [20].

2.1 Related work

Utilizing GPC and Sobol indices for SA has found broad adaptation in various domains such as mechanical, civil, biomedical, and chemical engineering [21, 22, 23, 24]. In the fields of material engineering and nondestructive testing, GPC has been applied to assess uncertainty in Lorentz force eddy current testing, providing valuable insights into the reliability of defect detection through Sobol indices [25]. A similar approach has been used to identify optimal stimulation sites for transcranial magnetic stimulation by evaluating sensitivities to uncertain model parameters, such as the conductivity of grey and white matter [26, 27]. For analyzing the sensitivity of image models, the authors in [28] propose using Sobol indices to quantify higher-order interactions between input features and the model's output. This approach attributes the neural network's predictions to the most important image regions, outperforming other black-box methods such as RISE (randomized input sampling for explanation) [29] and even white-box methods like Grad-CAM (gradient-weighted class activation mapping) [30].

Building on this literature, the following briefly overviews the theoretical background of GPC and Sobol indices.

2.2 Generalized polynomial chaos

GPC is a method used to approximate a surrogate model of $f(\boldsymbol{\xi})$ with random variables $\boldsymbol{\xi}$. The model is approximated by expanding it in terms of multivariate orthogonal polynomial basis functions Ψ. This can be expressed as [14, 31]

$$f(\boldsymbol{\xi}) \approx \sum_{\boldsymbol{\alpha} \in A} c_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}). \tag{1}$$

The expansion coefficients $c_{\boldsymbol{\alpha}}$ depend on the multi-index $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d) \in A$ where $A \subseteq \mathbb{N}_0^d$. The multivariate orthogonal polynomial basis functions are given by $\Psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) = \prod_{i=1}^{d} \psi_{\alpha_i}^i(\xi_i)$. These basis functions are products of univariate orthogonal polynomials $\psi_{\alpha_i}(\xi_i)$ of order α_i, which are orthogonal with respect to the probability density functions $p_{\Xi_i}(\xi_i)$ [32]. In this context, orthogonality is defined by the integral of the inner product of two basis functions, $\psi_{\alpha_i}(\xi_i)$ and $\psi_{\beta_i}(\xi_i)$, weighted by the probability measure $p_{\Xi_i}(\xi_i)$. This integral is zero unless $\alpha_i = \beta_i$. When $\alpha_i = \beta_i$, the integral equals the L^2-norm of the basis function. Therefore, the orthogonality condition can be expressed by

$$\int \psi_{\alpha_i}(\xi_i) \psi_{\beta_i}(\xi_i) p_{\Xi_i}(\xi_i) \, d\xi_i = \|\psi_{\alpha_i}(\xi_i)\|^2 \delta_{\alpha_i, \beta_i}, \tag{2}$$

where $\delta_{\alpha_i \beta_i}$ is the Kronecker delta function [19]. Using the Kronecker delta, which equals one when $\alpha_i = \beta_i$, and applying Equations (1) and (2), it can be demonstrated that the coefficients $c_{\boldsymbol{\alpha}}$ are expressed as

$$c_{\boldsymbol{\alpha}} = \frac{1}{\prod_i \|\psi_i(\xi_i)\|^2} \int f(\boldsymbol{\xi}) \Psi(\boldsymbol{\xi}) p_{\Xi}(\boldsymbol{\xi}) \, d\boldsymbol{\xi}. \tag{3}$$

The term can be approximated by regression or quadrature methods [31, 14].

By leveraging a surrogate model to approximate a model's output probability distribution using Equation (1), Sobol indices can be computed efficiently.

2.3 Sobol indices

Sobol indices are a variance-based sensitivity analysis method that quantifies the influence of a subset $\boldsymbol{\xi}_\tau$ of a model's inputs on its outputs. The indices $\tau \in \mathcal{P}(\{1, \ldots, d\})$ include all interactions of a model's input variables. Thus, Sobol indices can quantify the influence of individual input variables and the combined effect of different interactions of input variables [33]. We distinguish between total Sobol indices D_τ and relative Sobol indices S_τ. The total Sobol indices are given by

$$D_\tau = \mathbb{V}_{\boldsymbol{\xi}_\tau}[f(\boldsymbol{\xi})], \tag{4}$$

where $\mathbb{V}[f(\boldsymbol{\xi})]$ is the variance of $f(\boldsymbol{\xi})$ explained by the input variables $\boldsymbol{\xi}_\tau$. For the relative Sobol indices, we normalize Equation (4) by the sum over all interactions of input variables [15]

$$S_\tau = \frac{D_\tau}{\sum_\tau D_\tau} = \frac{\mathbb{V}_{\boldsymbol{\xi}_\tau}[f(\boldsymbol{\xi})]}{\sum_\tau \mathbb{V}_{\boldsymbol{\xi}_\tau}[f(\boldsymbol{\xi})]} = \frac{\mathbb{V}_{\boldsymbol{\xi}_\tau}[f(\boldsymbol{\xi})]}{\mathbb{V}[f(\boldsymbol{\xi})]}. \tag{5}$$

Thus, S_τ can be seen as a measure of the relative influence of $\boldsymbol{\xi}_\tau$ on $f(\boldsymbol{\xi})$. Global and relative Sobol indices can be computed using the coefficients in Equation (1) by

$$S_\tau = \frac{\sum_{\boldsymbol{\alpha} \in A_\tau} c_{\boldsymbol{\alpha}}^2 \|\Psi_{\boldsymbol{\alpha}}\|^2}{\sum_{\boldsymbol{\alpha} \in A} c_{\boldsymbol{\alpha}}^2 \|\Psi_{\boldsymbol{\alpha}}\|^2}, \tag{6}$$

with $A_\tau = \{\boldsymbol{\alpha} \in A : \alpha_i > 0 \iff i \in \tau\}$ representing the set of multi-indices $\boldsymbol{\alpha}$ for which $\alpha_i > 0$ if and only if i is an element of the subset τ [15].

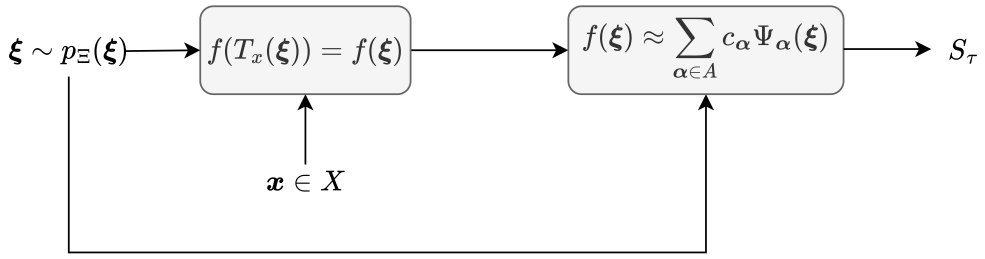


Figure 1: The fixed input $\boldsymbol{x}$, given the perturbation $\boldsymbol{\xi}$, is transformed using the transformation function $T(\boldsymbol{x}, \boldsymbol{\xi})$ and passed through the black-box model f to obtain the probability distribution $p_Y(\boldsymbol{y} \mid \boldsymbol{\xi})$. GPC approximates a surrogate model of $f(T(\boldsymbol{x}, \boldsymbol{\xi}))$ in terms of a series expansion of multivariate orthogonal polynomials. The surrogate model is then used to compute the Sobol indices S_τ.

3 GPC-SA FOR IMAGE CLASSIFICATION MODELS

Perturbations on images, such as noise, lighting variations (e.g., brightness), and geometric distortions (e.g., rotation), can significantly impact the performance of image classification models. These perturbations introduce variations in the input data distribution that the model may not be adequately trained on. As a result, the model must account for these variations to maintain robust performance.

Let $f : \mathbb{R}^{m \times n \times c} \to \{1, \ldots, K\}$ define the image classification model with input images $\boldsymbol{x} \in \mathbb{R}^{m \times n \times c}$, where m and n depict the width and height of the image, and c represents the number of image channels. The set $y = \{1, \ldots, K\}$ represents the output labels with an associated probability distribution $p_Y(\boldsymbol{y})$ for K classes. Now, let $p_Y(\boldsymbol{y} \mid \boldsymbol{x}) = f(\boldsymbol{\theta}, \boldsymbol{x})$ be a probabilistic classification model with learnable parameters $\boldsymbol{\theta}$ that maps $\boldsymbol{x}$ to $p_Y(\boldsymbol{y} \mid \boldsymbol{x})$ [34]. We aim to quantify the influence of perturbations on the image as transformations of the model's input variables $\boldsymbol{x}$ on its output $p_Y(\boldsymbol{y} \mid \boldsymbol{x})$ using Sobol indices of different interaction orders.

To quantify the relative influence of perturbations on the model's output, we consider a set of random variables $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_d)$ drawn from a probability distribution $p_\Xi(\boldsymbol{\xi})$. We define the transformation function $T(\boldsymbol{x}, \boldsymbol{\xi})$ to analyze these perturbations. We then compute the transformed inputs $\boldsymbol{x'}^{(i)} = T(\boldsymbol{x}, \boldsymbol{\xi}^{(i)})$ for each sample $\boldsymbol{\xi}^{(i)}$. The model then predicts the probability $f(\boldsymbol{x'}^{(i)}) = f(T(\boldsymbol{x}, \boldsymbol{\xi}^{(i)})) = p_Y(\boldsymbol{y} \mid \boldsymbol{x'}^{(i)})$ of $\boldsymbol{x'}$ belonging to class $\boldsymbol{y}$. By fixing $\boldsymbol{x}$, we define $T_x(\boldsymbol{\xi})$ as a function that depends only on the transformations $\boldsymbol{\xi}$ and serves as an uncertain input of the model. Thus, for a fixed $\boldsymbol{x} \in X$, it follows

$$f(\boldsymbol{x'}) = f(T_x(\boldsymbol{\xi})) = f(\boldsymbol{\xi}) = p_Y(\boldsymbol{y} \mid \boldsymbol{\xi}). \tag{7}$$

This approach allows us to isolate and analyze the influence of the perturbations $\boldsymbol{\xi}$ on the model's predictions, facilitated by the computation of Sobol indices to quantify the influence of $\boldsymbol{\xi}$ on the model's output. However, the range of polynomials is $(-\infty, +\infty)$, while the range of a discrete probability distribution is $[0, 1]$. To account for this, we apply the logit function in order to map the probability to $(-\infty, +\infty)$. This can be written as

$$f(\boldsymbol{\xi}) = \operatorname{logit}(p_Y(\boldsymbol{y} \mid \boldsymbol{\xi})) = \log\left(\frac{p_Y(\boldsymbol{y} \mid \boldsymbol{\xi})}{1 - p_Y(\boldsymbol{y} \mid \boldsymbol{\xi})}\right). \tag{8}$$

Using Equation (1), we approximate the surrogate model by

$$f(\boldsymbol{\xi}) = \text{logit}(p_Y(\boldsymbol{y} \mid \boldsymbol{\xi})) \approx \sum_{\boldsymbol{\alpha} \in A} c_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}). \tag{9}$$

Next, we compute the Sobol indices S_τ for various interaction orders τ for each input $\boldsymbol{x}'^{(i)}$ using Equation (6). To interpret the probabilities, we apply the inverse of Equation (8), which is the logistic function. This can be written as

$$\text{logistic}\left(\sum_{\boldsymbol{\alpha} \in A} c_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}\right) = \frac{1}{1 + \exp\left(-\sum_{\boldsymbol{\alpha} \in A} c_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}\right)}. \tag{10}$$

A schema of the complete approach is provided in Figure 1.

4 RESULTS

In the following, we validate our proposed approach using image classification models that have been fine-tuned on two different datasets. We demonstrate that approximating $f(T(\boldsymbol{x}, \boldsymbol{\xi}))$ using GPC and computing the Sobol indices can be used to quantify the sensitivity of perturbations in the input image on the model output. In Section 4.1, we analyze the sensitivity of a classification model trained on a benchmark welding seam dataset. Section 4.2 analyzes an image classification model used in BMW Group's vehicle manufacturing plants to detect the correct assembly of brand emblems.

4.1 TIG welding seam classification

We validate our approach on an image classification model using a pre-trained and fine-tuned ResNet18 model [35]. The model comprises 11.7 million parameters, ReLU activations, and a single 7x7 convolution layer with pooling. The pre-trained ResNet18 model is fine-tuned on benchmark Aluminum 5083 TIG welding seam images recorded with an HDR camera, down-sampled to 224x224 pixels with a single channel and without applying any transformations [36]. The training dataset contains 26,666 images, and the test dataset contains 6,588 images. The images show six defect types: good weld, burn-through, contamination, lack of fusion, misalignment, and lack of penetration.

Table 1: Sobol indices from the GPC surrogate model for *brightness*, *rotation*, and *tilt*. The results for all three transformations have a relative influence of 0.061. *Tilt* exhibits the highest influence.

	brightness	rotation	tilt
brightness	0.171	0.03	0.162
rotation	-	0.11	0.096
tilt	-	-	**0.37**

We are interested in evaluating the sensitivity of the fine-tuned image classification model to variations in brightness, rotation, and tilt, as well as the interactions among these transformations. To achieve this, we compute the Sobol indices, which quantify the contribution of each input variable and their interactions with the output variance. By analyzing the Sobol indices, we can determine the sensitivity of the classification model to each transformation and understand how these transformations interact to affect the model's performance.

For this purpose, we draw $n = 1000$ samples using Latin hypercube sampling (LHS) from a Beta distribution with specified limits for each transformation parameter. The

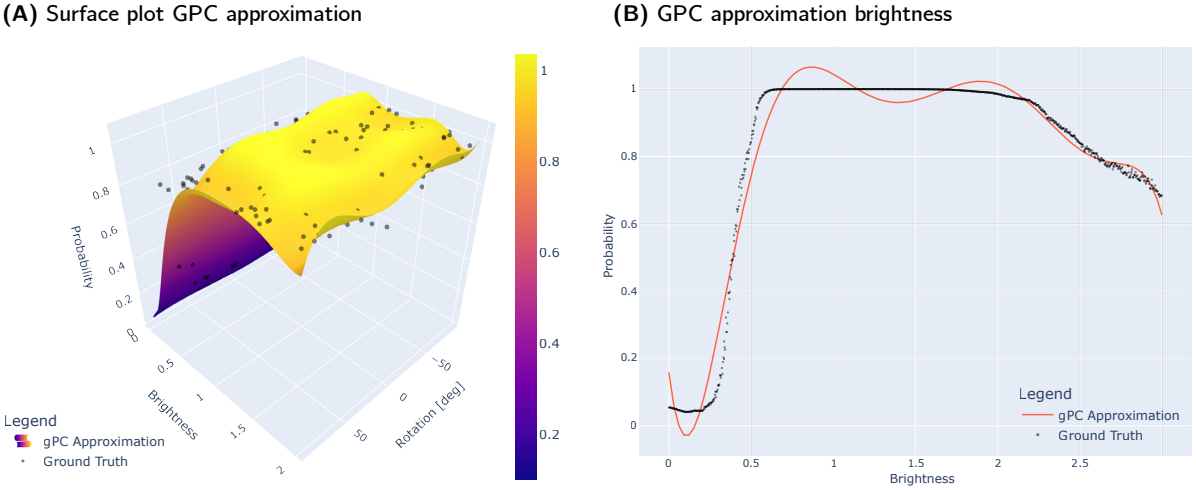


Figure 2: Figure (A) presents a surface plot of the GPC approximation, illustrating the relationship between *brightness* and *rotation* for the prediction probability, with *tilt* held constant at 0. Figure (B) displays the approximation of *brightness* while both *rotation* and *tilt* are fixed at 0.

limits are $[0, 2]$ for *brightness* (representing the scaling factor of brightness change, where 1 represents the original brightness), $[-30, 30]$ for *rotation* (the angle of rotation in degrees), and $[-20, 20]$ for *tilt* (the angle of tilt in degrees).

We apply these transformations to a fixed input image and compute the output probabilities of the image classification model using Equation (7). The surrogate model is approximated using GPC with Jacobi polynomials. The relative error of the approximation is 0.057. We set the maximum total order of polynomials to four and use the Moore-Penrose pseudo inverse to compute the polynomial coefficients. We then determined the Sobol indices with the computed coefficients, as given by Equation (6).

Table 1 presents the resulting Sobol indices. These indices indicate that *tilt* has the highest influence on the model, with a value of 0.37. In contrast, *brightness* and *rotation* exert the lowest influence on the model, each with a value of 0.03. The combined relative influence of *brightness*, *rotation*, and *tilt* is 0.061[1].

The surrogate models illustrating the GPC approximation for *rotation* and *brightness* are depicted in Figure 2. Figure 2 (A) presents a surface plot of the surrogate model with respect to *rotation* and *brightness*, where *tilt* is fixed at a value of 0. Figure 2 (B) shows a plot of the surrogate model with respect to *brightness*, where *rotation* and *tilt* are fixed at a value of 0.

4.2 Vehicle emblem classification

In this experiment, we validate our SA approach on an image classification model fine-tuned to detect the correct assembly of brand emblems in BMW's Group vehicle production plants. The model classifies images into three distinct emblem types and includes an additional class for unclassified instances. Operating as a black box, the image classification model provides output class probabilities for given inputs. We aim

[1]For those interested in reproducing the results or exploring the SA approach further, the code including the fine-tuned ResNet18 model is available on GitHub at `https://github.com/lpossner/GPC_Image_Classification`.

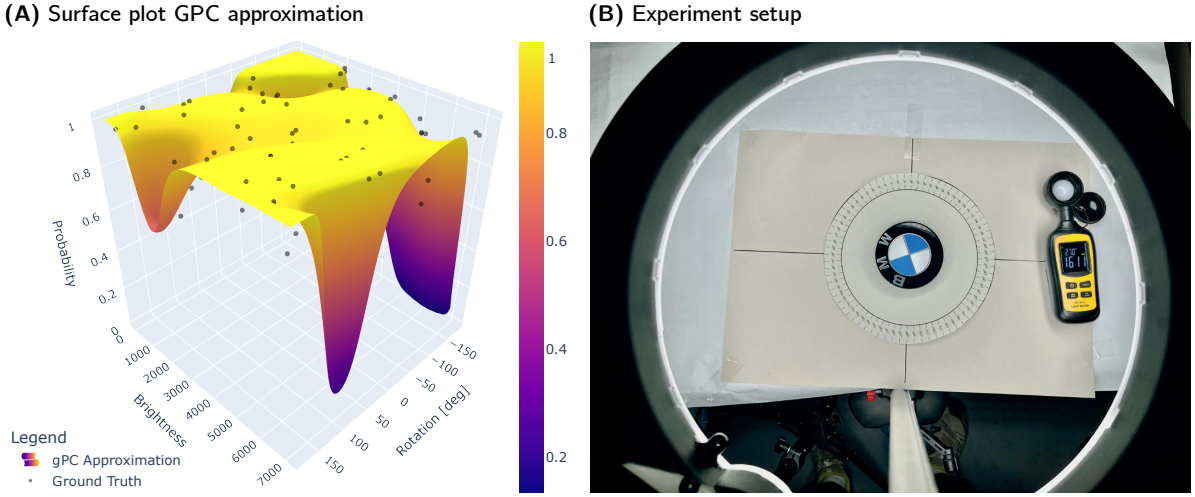


Figure 3: Figure (A) presents a surface plot of the GPC approximation, illustrating the relationship between *brightness* and *rotation* for the prediction probability. Figure (B) shows the experimental setup used to obtain the validation images.

to evaluate its sensitivity to variations in *rotation* and *brightness* and the interactions between these transformations, using Sobol indices.

To conduct our experiment, we set up an apparatus comprising a Keyence camera, an adjustable light source, a luminance meter, and an angle scale. We then adjusted the *rotation* and *brightness* for $n = 100$ samples using LHS drawn from a Beta distribution. The limits for the *rotation* transformation (the angle of rotation in degrees) are set between $[-180, 180]$, and for the *brightness* transformation (representing brightness in Lux), the range is $[0, 7000]$. With this configuration, we adjust the *rotation* angle and *brightness* and shot the images (cf. Figure 3 (B)) to compute the output class probabilities using Equation (7). To approximate the surrogate model, we employ GPC with Jacobian polynomials and subsequently compute the Sobol indices. The relative error obtained from this method is 0.23, given a maximum total polynomial order of $[6, 5]$.

Table 2: Result of the Sobol indices for *brightness* and *rotation*. The joint interaction exhibits the highest influence on the emblem classification model.

	brightness	rotation
brightness	0.229	**0.403**
rotation	-	0.368

Table 2 presents the Sobol indices obtained from our experiment. The results indicate that the joint interaction of *brightness* and *rotation* has the highest influence, with a Sobol index of 0.403. Figure 3 (A) displays a surface plot of the GPC approximation. In the following chapter, we contextualize the results.

5 DISCUSSION AND CONCLUSION

This paper presents preliminary results on the effectiveness of sensitivity analysis of image classification models with perturbed images using GPC. We introduce a method where a fixed input image is transformed by samples from a known distribution that is associated with a family of orthogonal polynomial functions. Using the output of the image classification model for the transformed input, we employ GPC to approximate a

surrogate model [14]. This surrogate model enables us to determine the relative influence of each transformation applied to the input on the model's output, quantified as Sobol indices.

In Section 4, we demonstrate the approach on an Aluminum 5083 TIG welding seam images dataset for which we fine-tuned a ResNet18 model and on an emblem image classification model utilized in BMW Group's production facilities [36]. The experiment provides initial results on evaluating the models' sensitivity to images that are exposed to perturbations, which can potentially lower the model's prediction performance. Our findings show we can approximate the nonlinear models' input-output relationships using GPC. The findings further suggest that geometric transformations and brightness adjustments alter the visual structure in a way that negatively affects the prediction accuracy of the classification model. This can be attributed to distributional domain shifts that differ from the models' training dataset. Quantifying the relative influence of input transformations using Sobol indices provides valuable insights into how variations in the input affect the model's predictions. This information can assist practitioners in enhancing the robustness of models, for example, making the emblem classification model more resilient to different lighting conditions.

However, several questions remain unanswered. It remains to be validated whether the Sobol indices generalize across the entire distribution of the dataset space for a fixed input, suggesting a potential avenue for future research. Moreover, in this study, we test only one family of polynomials, specifically the Jacobian polynomials. This leaves potential research opportunities open to exploring other polynomial families and examining their impact on the accuracy and efficiency of the surrogate model. Further investigation is required to determine the robustness of our method across different datasets and GPC configurations.

REFERENCES

[1] ISO/IEC, "Risk management — Risk assessment techniques," standard, International Organization for Standardization, Geneva, CH, June 2019.

[2] IEC, "IEC 60812:2018 - Failure modes and effects analysis (FMEA and FMECA)," tech. Rep., International Electrotechnical Commission (IEC), Geneva, Switzerland, 2018.

[3] E. Ruijters and M. Stoelinga, "Fault tree analysis: A survey of the state-of-the-art in modeling, analysis and tools," *Computer Science Review*, vol. 15-16, pp. 29–62, Feb. 2015.

[4] L. Bahr, C. Wehner, J. Wewerka, J. Bittencourt, U. Schmid, and R. Daub, "Knowledge Graph Enhanced Retrieval-Augmented Generation for Failure Mode and Effects Analysis," *arXiv preprint arXiv:2406.18114*, July 2024.

[5] C. A. Escobar, M. E. McGovern, and R. Morales-Menendez, "Quality 4.0: a review of big data challenges in manufacturing," *Journal of Intelligent Manufacturing*, vol. 32, pp. 2319–2334, Dec. 2021.

[6] A. Mayr, D. Kißkalt, M. Meiners, B. Lutz, F. Schäfer, R. Seidel, A. Selmaier, J. Fuchs, M. Metzner, A. Blank, and J. Franke, "Machine Learning in Production – Potentials,

Challenges and Exemplary Applications," *Procedia CIRP*, vol. 86, pp. 49–54, Jan. 2019.

[7] A. Zonnenshain and R. S. Kenett, "Quality 4.0—the challenging future of quality engineering," *Quality Engineering*, vol. 32, pp. 614–626, Oct. 2020.

[8] L. Possner, L. Bahr, L. Roehl, C. Wehner, and S. Groeger, "Anwendung von Causal-Discovery-Algorithmen zur Root-Cause-Analyse in der Fahrzeugmontage," *arXiv preprint arXiv:2407.16388*, July 2024.

[9] H. Tercan and T. Meisen, "Machine learning and deep learning based predictive quality in manufacturing: a systematic review," *Journal of Intelligent Manufacturing*, vol. 33, pp. 1879–1905, Oct. 2022.

[10] A. Saltelli, S. Tarantola, F. Campolongo, and M. Ratto, *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*. Wiley, 1 ed., Feb. 2002.

[11] V. T. Vasudevan, A. Sethy, and A. R. Ghias, "Towards Better Confidence Estimation for Neural Models," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7335–7339, 2019.

[12] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. V. Dillon, B. Lakshminarayanan, and J. Snoek, "Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift," in *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA: Curran Associates Inc., 2019.

[13] J. Gawlikowski, C. R. N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher, M. Shahzad, W. Yang, R. Bamler, and X. X. Zhu, "A survey of uncertainty in deep neural networks," *Artificial Intelligence Review*, vol. 56, pp. 1513–1589, Oct. 2023.

[14] K. Weise, L. Poßner, E. Müller, R. Gast, and T. R. Knösche, "Pygpc: A sensitivity and uncertainty analysis toolbox for Python," *SoftwareX*, vol. 11, p. 100450, Jan. 2020.

[15] B. Sudret, "Global sensitivity analysis using polynomial chaos expansions," *Reliability Engineering & System Safety*, vol. 93, pp. 964–979, July 2008.

[16] E. Borgonovo and E. Plischke, "Sensitivity analysis: A review of recent advances," *European Journal of Operational Research*, vol. 248, pp. 869–887, Feb. 2016.

[17] A. Saltelli and P. Annoni, "How to avoid a perfunctory sensitivity analysis," *Environmental Modelling & Software*, vol. 25, pp. 1508–1517, Dec. 2010.

[18] Y. Tang, P. Reed, T. Wagener, and K. van Werkhoven, "Comparing sensitivity analysis methods to advance lumped watershed model identification and evaluation," *Hydrology and Earth System Sciences*, vol. 11, pp. 793–817, Feb. 2007.

[19] D. Xiu and G. Em Karniadakis, "Modeling uncertainty in steady state diffusion problems via generalized polynomial chaos," *Computer Methods in Applied Mechanics and Engineering*, vol. 191, pp. 4927–4948, Sept. 2002.

[20] A. Saltelli, "Sensitivity analysis for importance assessment.," *Risk analysis : an official publication of the Society for Risk Analysis*, vol. 22, pp. 579–590, June 2002.

[21] D. Lucor, J. Meyers, and P. Sagaut, "Sensitivity analysis of large-eddy simulations to subgrid-scale-model parametric uncertainty using polynomial chaos," *Journal of Fluid Mechanics*, vol. 585, pp. 255–279, Aug. 2007.

[22] P. Ni, Y. Xia, J. Li, and H. Hao, "Using polynomial chaos expansion for uncertainty and sensitivity analysis of bridge structures," *Mechanical Systems and Signal Processing*, vol. 119, pp. 293–311, Mar. 2019.

[23] P. L. T. Duong, W. Ali, E. Kwok, and M. Lee, "Uncertainty quantification and global sensitivity analysis of complex chemical process using a generalized polynomial chaos approach," *Computers & Chemical Engineering*, vol. 90, pp. 23–30, July 2016.

[24] M. Tosin, A. M. A. Côrtes, and A. Cunha, "A Tutorial on Sobol' Global Sensitivity Analysis Applied to Biological Models," in *Networks in Systems Biology: Applications for Disease Modeling*, pp. 93–118, Cham: Springer International Publishing, 2020.

[25] K. Weise, M. Carlstedt, M. Ziolkowski, and H. Brauer, "Uncertainty Analysis in Lorentz Force Eddy Current Testing," *IEEE Transactions on Magnetics*, vol. 52, pp. 1–4, Mar. 2016.

[26] G. B. Saturnino, A. Thielscher, K. H. Madsen, T. R. Knösche, and K. Weise, "A principled approach to conductivity uncertainty analysis in electric field calculations," *NeuroImage*, vol. 188, pp. 821–834, Mar. 2019.

[27] O. Numssen, K. Weise, A. Thielscher, T. R. Knösche, and G. Hartwigsen, "Determining the stimulation site in TMS: Sensitivities with respect to uncertain model parameters," June 2019.

[28] T. Fel, R. Cadène, M. Chalvidal, M. Cord, D. Vigouroux, and T. Serre, "Look at the variance! efficient black-box explanations with sobol-based sensitivity analysis," in *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, (Red Hook, NY, USA), Curran Associates Inc., 2024.

[29] V. Petsiuk, A. Das, and K. Saenko, "RISE: Randomized Input Sampling for Explanation of Black-box Models," *arXiv:1806.07421*, 2018.

[30] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization," *International Journal of Computer Vision*, vol. 128, pp. 336–359, Feb. 2020.

[31] D. Xiu and G. E. Karniadakis, "The Wiener–Askey Polynomial Chaos for Stochastic Differential Equations," *SIAM Journal on Scientific Computing*, vol. 24, no. 2, pp. 619–644, 2002.

[32] R. Askey and J. A. Wilson, *Some basic hypergeometric orthogonal polynomials that generalize Jacobi polynomials*, vol. 319. American Mathematical Soc., 1985.

[33] I. M. Sobol, "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates," *Mathematics and Computers in Simulation*, vol. 55, pp. 271–280, Feb. 2001.

[34] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[36] D. Bacioiu, G. Melton, M. Papaelias, and R. Shaw, "Automated defect classification of Aluminium 5083 TIG welding using HDR camera and neural networks," *Journal of Manufacturing Processes*, vol. 45, pp. 603–613, Sept. 2019.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

INPUT PARAMETER SPECIFIC UNCERTAINTY QUANTIFICATION OF DEEP LEARNING MODELS USING MONTE CARLO DROPOUT

Eugen Boos[1,*], Maik Linnemann[2], Hajo Wiemer[1], Verena Psyk[2] and Steffen Ihlenfeldt[1,2]

[1] Institute of Mechatronic Engineering, Technische Universität Dresden, Germany

[2] Fraunhofer Institute for Machine Tools and Forming Technology, Chemnitz, Germany

* e-mail: eugen.boos@tu-dresden.de

Abstract

Low-data environments, common in industrial use-cases, exhibit significantly elevated levels of both model and data uncertainty in deep learning applications. Data generation in such settings is generally financially expensive. Therefore, strategic data collection plays a critical role in improving the performance of deep learning models. Effective model uncertainty quantification plays an essential role in identifying input domains with poor model confidence, enabling more efficient data collection strategies. While Monte Carlo dropout is a well-known method for model uncertainty quantification, this study explores a novel input parameter specific application of Monte Carlo dropout to sequentially applied input channel layers rather than the entire model. This approach allows for a more detailed uncertainty quantification focused on specific input parameters, yielding additional insights into model robustness and model confidence across different input domains. A case study on condition monitoring in manufacturing using the Temporal Fusion Transformer demonstrates how these insights can guide smarter data generation strategies, ultimately improving model performance in low-data, high-cost environments.

Keywords: Uncertainty Quantification, Monte Carlo Dropout, Deep Learning, Design of Experiment

1 INTRODUCTION

Machine Learning (ML) and particularly Deep Learning (DL) have seen a significant growth in development as well as application in various engineering disciplines, particularly speaking in production engineering. These developments lead to meaningful progress in areas of application such as for instance condition monitoring [1] and predictive maintenance [2]. The resulting advancements enable improved efficiency, reduced downtime, and enhanced decision-making through data-driven insights. However, despite these advancements, the application of ML and DL in production engineering environments still presents unique challenges. One of the most significant limitations in engineering environments are low data quantities and high generation costs [3]. Especially in regards to data collected from malfunctioning processes or machines. In contrast to other domains, where large datasets are accessible, engineering applications often cope with data that is sparse, noisy, and costly to generate. These constraints hinder the ability of data-driven models to repeatedly output reliable predictions. This is particularly relevant for DL models, as they are more dependent on large datasets while generally achieving a higher accuracy than traditional ML models [4].

A critical aspect influencing the performance and reliability of DL models in these contexts is uncertainty. Uncertainty impacts model explainability, robustness and overall trustworthiness [5]. Uncertainty quantification (UQ) addresses these factors, resulting in an overall improved insight into model performance and reliability [6]. This is particularly the case in low-data engineering environments [7]. Uncertainty can be categorized into epistemic and aleatoric uncertainty, both of which influence model predictions differently [8], [9]. Epistemic uncertainty arises from a lack of knowledge about the model parameters and can be reduced by incorporating additional data, improving model architecture or refining the parameters underlying assumptions. This type of uncertainty is particularly relevant in cases where the model has not encountered sufficient examples of a specific input, making it essential for identifying out-of-distribution samples. In contrast, aleatoric uncertainty is inherent to the input data itself and cannot be mitigated through additional observations. It results from stochastic noise in the input, such as sensor errors, occlusion or environmental variations. Unlike epistemic uncertainty, which reflects model limitations, aleatoric uncertainty characterizes the inherent randomness in the system being modeled itself. While epistemic uncertainty can be directly addressed through data augmentation or improved modeling strategies, aleatoric uncertainty must be accounted for probabilistically to ensure robust and reliable predictions in real-world applications.

Generally, increasing the amount of available data reduces model uncertainty [7]. However, in environments where data generation is costly, strategic data generation and selection become crucial to efficiently improve model confidence while minimizing resource expenditure. Traditional UQ methods primarily focus on characterizing the uncertainty of the entire model, making them more effective in capturing epistemic uncertainty [6, 10, 11]. These approaches, while valuable for quantifying overall model uncertainty, do not offer a deeper look into the underlying sources of the uncertainty. Thus, besides general improvement strategies, such as increasing data quantity or improved model architectures, a strategic design of experiments (DoE) to cost efficiently lower model uncertainty is not directly derivable from UQ. Enabling an input parameter-specific UQ allows for a more specific identification of uncertainty sources, specifically in regards to the lack of data in smaller sized data sets. This approach enhances uncertainty interpretability and explainability by addressing the contribution of individual input parameters onto the overall model uncertainty, thereby distinguishing between inherent variability in the data and uncertainty stemming from model limitations. By isolating these two,

targeted mitigation strategies can be developed, optimizing DoE and resource allocation for improving model robustness and predictive reliability.

1.1 Research objectives and main contribution

Leveraging advanced DL models and UQ techniques becomes crucial in improving model reliability, optimizing financial resource allocation and enhancing predictive performance despite data scarcity. This study aims to address these challenges by developing an input parameter-specific UQ approach based on Monte Carlo dropout (MCD) [12]. By further distinguishing between aleatoric and epistemic uncertainty, this method enhances model interpretability and supports a strategic DoE to cost efficiently reduce uncertainty. Although aleatoric uncertainty remains constant in big data environments, since it is tied to intrinsic noise in the data rather than data scarcity [7], in small-data environments, its quantification improves as more data becomes available [13]. With limited data, aleatoric uncertainty estimates are often biased due to insufficient representation of the underlying stochastic variations. As the dataset grows, the model gains a more accurate understanding of the inherent noise, leading to a more reliable uncertainty quantification and an improved predictive performance. This underscores the importance of strategic data generation in small-data environments to enhance the robustness of uncertainty estimation. For demonstrating and validating the proposed method, this study employs experimental data from the electromagnetic forming (EMF) process [14].

2 RELATED WORK

2.1 Methods for uncertainty estimation in deep learning

UQ in DL has gained increasing attention as models are deployed in applications requiring reliable decision-making. While traditional research has primarily focused on improving point-estimate prediction accuracy, recent review studies show a growing tendency towards UQ [6], [15]. The quantification of uncertainty in deep neural networks can be roughly categorized into four groups: deterministic models, Bayesian methods, ensemble methods and test-time augmentation methods.

Deterministic methods estimate uncertainty by providing a scalar value or measure that quantifies the level of uncertainty associated with a prediction. These methods assume that the underlying model is deterministic and that uncertainty can be inferred from a single forward pass through the network, rather than requiring multiple stochastic evaluations. Prevalent methods are Evidentuial Deep Learning [16], Prior Networks [17] and distance aware networks [18]. Additionally, uncertainty-aware loss functions, such as quantile loss [19], are also part of deterministic models. Their primary advantage lies in their computational efficiency and simplicity, making them well-suited for applications requiring rapid inference with minimal overhead. Bayesian methods or Bayesian Neural Networks (BNN) are a class of approaches that incorporate uncertainty by representing model parameters as probabilistic distributions rather than fixed values [20]. This framework enables Bayesian methods to estimate predictive uncertainty by capturing the distribution of possible outputs for a given input, rather than a single-point prediction. However, computing the exact posterior distribution is intractable for deep neural networks, requiring the use of approximation methods [9]. The most prevailing approximation methods are Variational Inference (VI) [21] and Monte Carlo dropout (MCD) [12, 22]. VI approximates the posterior by optimizing a simpler normal distribution with the assumption the input data follows a normal distribution, while MCD estimates uncertainty by applying dropout at inference time, effectively sampling from different subnetworks to generate a distribution of predictions. Bayesian methods provide robust frameworks for quantifying both

epistemic and aleatoric uncertainty. However, their computational complexity is high, and their scalability remains limited compared to alternative approaches. Quite similar to the concept of sampling from multiple sub-networks are deep ensemble networks (DE). They estimate uncertainty by leveraging the output diversity among multiple independent neural networks for the same input to capture epistemic uncertainty [23]. Trained with different weight initializations or dataset partitions DE networks learn diverse representations of the data, allowing the ensemble to approximate the posterior distribution over model predictions. By aggregating the outputs through methods such as averaging, DE reduces overconfidence and provides a more reliable measure of epistemic uncertainty, particularly in cases where data is sparse or out-of-distribution. However, they are computationally expensive, scale inefficiently due to multiple forward passes and do not explicitly model aleatoric uncertainty. Test-Time Augmentation methods produce uncertainty estimations by applying data augmentations to input samples during inference, generating multiple predictions per input instead of a single deterministic output [24]. By analyzing the variance among augmented predictions, Test-Time Augmentation provides a computationally efficient estimation on aleatoric uncertainty, without requiring modifications to model architecture or training procedures. It is however mainly applicable in computer vision [6, 15].

2.2 Methods for strategic data generation and design of experiment

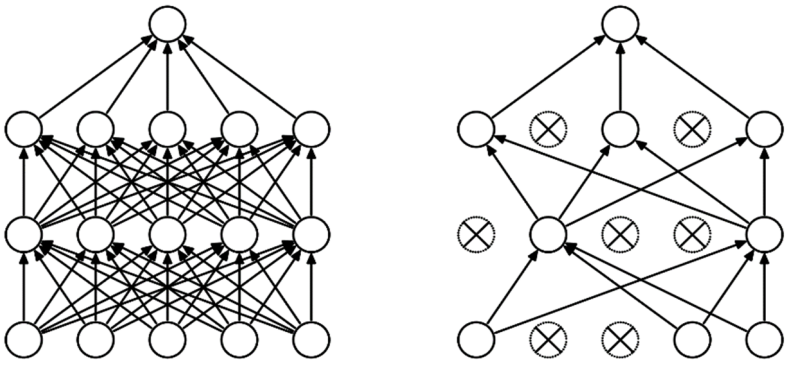
DoE is a significant tool in sparse data environments to strategically generate real world data for data driven applications, such as ML and DL. As often financial [25] or time [26] restrictions limit data generation, using the given resources strategically can influence the underlying model performance significantly. To address these hurdles there are different approaches. One of the most straight forward strategics for DoE comes down to space-filling and quasi-random sampling methods. They ensure uniform coverage of the input space by distributing samples evenly across the design space, making them ideal as an initial DoE strategy when data and model uncertainty is yet unknown. Techniques such as Latin Hypercube Sampling, Sobol Sequences, Halton Sequences, and Maximin Sampling help minimize gaps in the data space [27]. Space-filling methods offer a fixed experimental design for yet unknown data environments, reduce the risk of overfitting and contribute to model robustness. However, they do not account for aleatoric or epistemic uncertainty as well as model performance. In contrast, data-centric adaptive and sequential sampling strategies are designed to iteratively refine the data selection process to improve model performance while efficiently reducing uncertainty. Unlike fixed space-filling experimental designs, these methods dynamically adapt data collection based on model feedback, prioritizing high-uncertainty or high-value samples for learning. Well known approaches include active learning [28] and reinforcement learning-based adaptive sampling [29]. These strategies are particularly effective in environments, where data is expensive to obtain and the goal is to achieve maximum learning with minimal labeled data. As well as preferably environments where data generation and model training are directly intertwined enabling an automated or semi-automated training process. Alternatively, model-centric approaches focus on optimizing data generation through a probabilistic framework that systematically reduces model uncertainty. One of the most prevalent is Bayesian Experimental Design (BED) [30, 31] which selects experiments to maximize expected information gain, ensuring that each generate data point contributes maximally to refining posterior distributions, thus reducing epistemic uncertainty. By leveraging Bayesian inference, BED focuses on identifying the most informative regions in the design space. This makes it particularly effective for tasks where data collection is costly or limited. While BED ensures optimal data collection, it may be less practical for real-time applications, due to its high computational cost, requiring extensive posterior inference or optimization of expected information gain. In contrast to data-

and model-centric approaches, uncertainty-driven DoE methods target data generation by prioritizing regions of high uncertainty in the input space. These methods aim to reduce both epistemic and aleatoric uncertainty by selecting samples that are most informative for improving model predictions. Unlike data-centric approaches, which aim for multiple objectives such as sample diversity and model performance, uncertainty-driven methods focus mainly on reducing predictive uncertainty. Common techniques include variance-based sampling [32], which identifies regions with high predictive variance, and entropy-based sampling [33], which selects data points with high uncertainty across possible outputs. Additionally, approaches like Query-By-Committee leverage disagreements among ensemble models to guide sample selection, ensuring diverse and informative data collection [21]. By dynamically adapting the sampling process based on uncertainty feedback, these methods are particularly effective in environments where labeled data is scarce or expensive, enabling efficient learning with minimal data generation while enhancing model robustness and generalization.

3 FUNDAMENTELS

3.1 Dropout

Dropout is a well-known standard regularization practice [34] used to lessen overfitting in deep neural networks. During training, dropout prevents the model from over-relying on specific neurons by stochastically deactivating a subset of neurons in each iteration, thereby enforcing a more distributed representation of the input data. However, during inference, all neurons remain active, with their outputs are scaled by the dropout rate to ensure consistency between training and testing. The dropout rate governs the probability of neuron deactivation and is advised to be numerically optimized based on model architecture and dataset characteristics to achieve optimal generalization performance. Dropout is generally applied independently to each layer. Figure 1(b) illustrates the effect of applying dropout to a neural network during training in comparison to one without the application of dropout (Figure 1(a)). All in all the application of dropout not only mitigates overfitting but also enhances overall model accuracy by promoting more robust feature learning [35].



(a) Standard Neural Network (b) Neural Network with applied dropout

Figure 1: Comparison of a basic Neural Network (a) without applied dropout and (b) with applied dropout. [32]

The following equations describe the forward propagation process in a simple neural network with dropout regularization. Considering a neural network with L hidden layers, where each hidden layer is indexed by l, with $l \in \{1, \dots, L\}$. Let z^l represent the input vector to layer l and y^l denote the output vector, where $y^0 = x$ corresponds to the network input. The weight

matrix and bias vector for layer l are denoted as w^l and b^l, respectively. In a standard neural network, the pre-activation value z_i^{l+1} for neuron i in layer $l+1$ is computed as:

$$z_i^{l+1} = y^l w_i^{l+1} + b_i^{l+1}. \tag{1}$$

The activation function $f(\cdot)$ is then applied to produce the output y_i^{l+1}:

$$y_i^{l+1} = f\left(z_i^{l+1}\right). \tag{2}$$

When dropout is introduced, a binary mask r_j^l is sampled from a Bernoulli distribution with probability p, determining which neurons remain active in layer l. The adjusted activations $\tilde{y}^l$ are obtained by element-wise multiplication with r^l, leading to a modified pre-activation computation before applying the activation function f. The following equations present these principles:

$$\begin{aligned} r_j^l &\sim Bernoulli(p), \\ \tilde{y}^l &= r^l * y^l, \\ z_i^{l+1} &= \tilde{y}^l w_i^{l+1} + b_i^{l+1}. \end{aligned} \tag{3}$$

3.2 Monte Carlo Dropout

MCD utilized dropout to approximate the posterior distribution of a neural network [12]. It works by applying dropout not only during training, as traditional dropout does, but also at inference to generate multiple stochastic forward passes through the network. In a standard neural network, dropout randomly deactivates a subset of neurons during training, preventing overfitting. However, in MCD, this stochastic behavior is maintained at inference, effectively turning the network into an ensemble of models with different subsets of active neurons. To estimate the output for a given input, the model performs multiple forward passes T, where each forward pass is indexed by t, with $t \in \{1, \dots, T\}$, each time using a different dropout mask. Therefore, Equation (3) turns into:

$$z_i^{t,l+1} = \tilde{y}^{t,l} w_i^{l+1} + b_i^{l+1}. \tag{4}$$

This results in a distribution of predictions rather than a single deterministic output. The deterministic prediction can be obtained by averaging these stochastic outputs over T stochastic forward passes to:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^{T} y_L^t, \tag{5}$$

while the variance across them provides an estimate of the underlying model uncertainty:

$$\sigma^2 = \frac{1}{T} \sum_{t=1}^{T} (y_L^t - \hat{y})^2. \tag{6}$$

In case the variance is insufficient, the coefficient of variation (CV) serves as a unitless measure of relative dispersion, enabling comparison between datasets with different means. It is defined as:

$$CV = \frac{S}{\hat{y}}, \tag{7}$$

where S is the standard deviation:

$$S = \sqrt{\frac{1}{T-1}\sum_{t=1}^{T}(y_L^t - \hat{y})^2}\,. \tag{8}$$

By generating a distribution of predictions instead of a single deterministic output, MCD improves generalization and provides a measure of predictive confidence. This technique serves as a practical approximation of Bayesian inference in deep learning models.

4 PROPOSED INPUT PARAMETER SPECIFIC MODEL UNCERTAINTY QUANTIFICATION USING MONTE CARLO DROPOUT

4.1 Shortcomings of traditional Monte Carlo dropout

MCD enables the estimation of model prediction uncertainty by maintaining dropout activation during inference, effectively transforming the network into an ensemble of stochastic subnetworks. Therefore, in practice, every neural network layer incorporating dropout contributes to the overall uncertainty estimation, as the variability of each individual layer propagates through the network. By aggregating the uncertainty from all affected layers, MCD provides a measure for the prediction uncertainty of the overall model.

However, when applying MCD in the conventional manner, only the overall model uncertainty can be quantified. A differentiated attribution of uncertainty to individual layers is not feasible, as the method interprets dropout as a stochastic mechanism acting across the entire network. In each forward pass, a random substructure of the network is activated, but the resulting variability in the output is an aggregated effect of all stochastic components. Consequently, MCD does not provide insight into the relative contribution of individual layers to the total predictive uncertainty.

4.2 Application of Monte Carlo dropout on individual layers

The proposed application of MCD is aiming to utilize the principles of approximate Bayesian inference individually on single layers of the neural network. The idea is as follows: when applying the MCD principle to a single layer, the resulting output distribution reflects the uncertainty specifically to that one layer. Applying this concept to explicitly embedding or input channels of the neural network enables the UQ to represent the uncertainty of a specific input feature. In model architectures with separate input channels for each individual feature dimension, this method enables an assessment of uncertainty for each individual input parameter. Figure 2(a) shows a flowchart depicting the proposed input parameter-specific Monte Carlo dropout (IPS-MCD). While the overall workflow remains consistent with traditional MCD, the key distinction lies in the targeted application of dropout to a pre-selected input channel during inference.

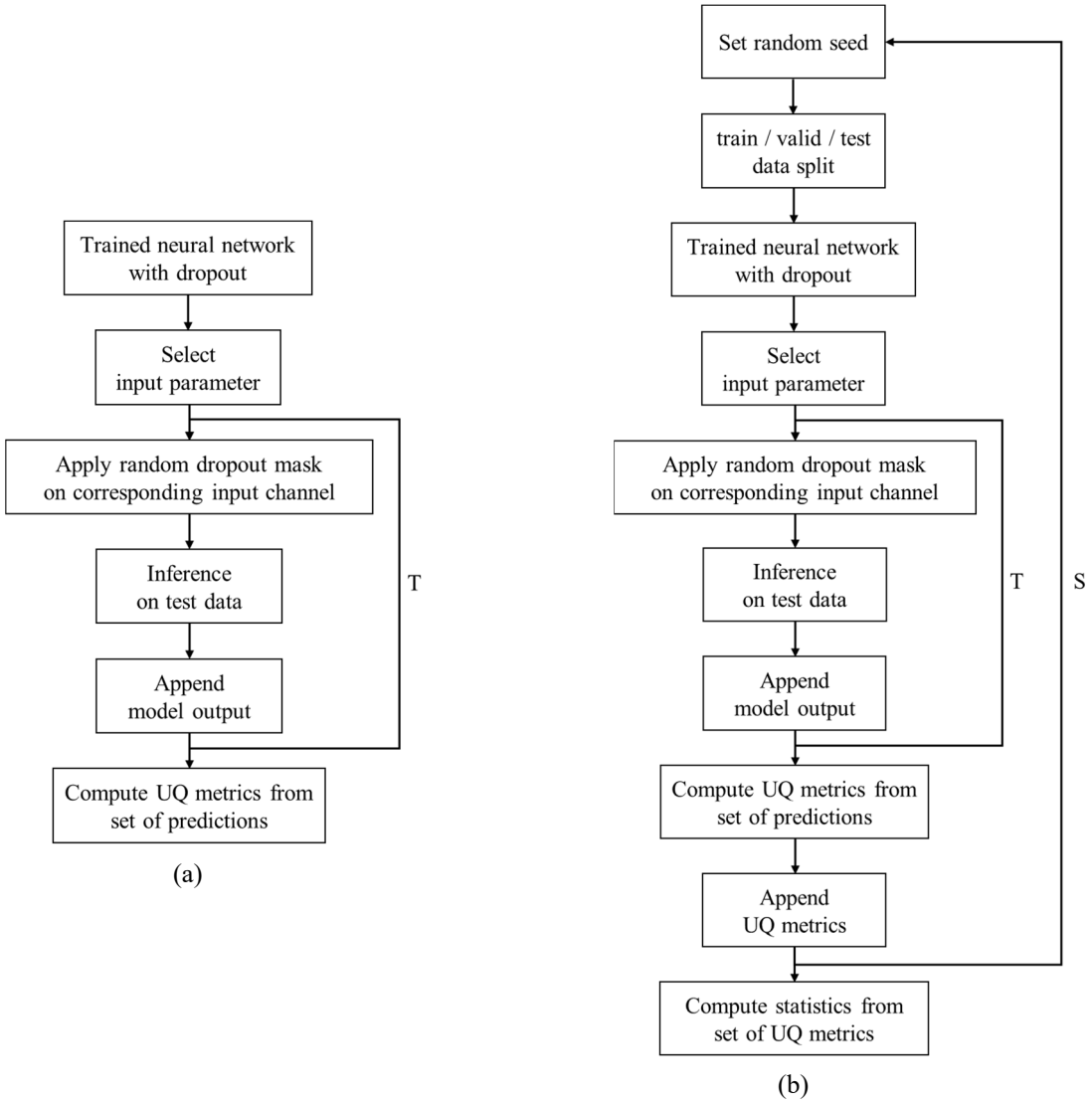


Figure 2: Flowchart of proposed input parameter-specific Monte Carlo dropout, as (a) the main loop and aus (b) the extended multi-seed loop.

5 EXPERIMENTAL SETUP AND VALIDATION

5.1 Machine Setup

EMF is a high-speed, contact-free manufacturing process that utilizes pulsed magnetic fields to plastically deform electrically conductive workpieces. However, the inherent uncertainties of the process complicate monitoring and control, particularly in predicting the exact forming depth of a workpiece from a single impulse. As part of a data-driven process monitoring model, this depth prediction is utilized to identify input parameter-specific uncertainties, enabling more efficient model improvement and data accumulation. The following paragraph briefly describes the inner workings of the EMF process. For the EMF process, a capacitor battery is charged to a defined voltage and then discharged via the tool coil (inductor) so that damped sinusoidal current flows, causing a transient magnetic field and eddy currents in the electrically conductive workpiece in the opposite direction to the inductor current. Interactions between magnetic field

and current cause Lorentz forces. Thus, the workpiece deforms plastically, when the yield stress is reached. Typical EMF pulses last 15 µs to 100 µs and the magnetic pressure increases up to several hundred mega Pascal [14] [36]. The workpiece is accelerated to speeds of up to several hundred meter per second. Strain rates are in the order of 103 s-1 to 104 s-1 [14]. Figure 3 sketches a free sheet metal forming process. Here, a flat coil with spiral coil winding is applied, but potential forming tasks are not limited to rotationally symmetric cases.

The experimental setup consists of a BlueWave PS 100-25 pulse power generator, which operates with a maximum nominal capacitor charging voltage V of 25 kV and a capacitor charging energy E of 100 kJ. The system includes 22 capacitors, each with a capacitance C of 15 µF, arranged into five capacitor banks that can be connected individually. The resulting capacitor charging energy E can be calculated by

$$E = \frac{1}{2} C U^2. \tag{9}$$

Discharge occurs through gas discharge switches when the capacitor charging voltage V is reached. The switches are connected via coaxial cables to a collector, which is then linked to the inductor using two rigid coaxial conductors. The forming process takes place within a drawing ring with a radius r_d of 20 mm, limiting deformation to a diameter D_d of 100 mm while allowing the rest of the workpiece to form freely. The distance between the coil and workpiece g, capacitance C, and capacitor charging voltage V are varied according to the test design.

For experimental process monitoring, a Rogowski probe (CWT 1500/4/1000, Power Electronics Measurement - PEM) is used to measure the actual transient current flow. The forming depth of the workpiece is recorded using a Keyence LK-H157 laser triangulation measuring system, which relies on a dot-shaped laser beam reflected from the workpiece surface to determine its position. To ensure accurate measurements, the evaluation focuses on the center of the workpiece, where the laser beam remains perpendicular to the surface.

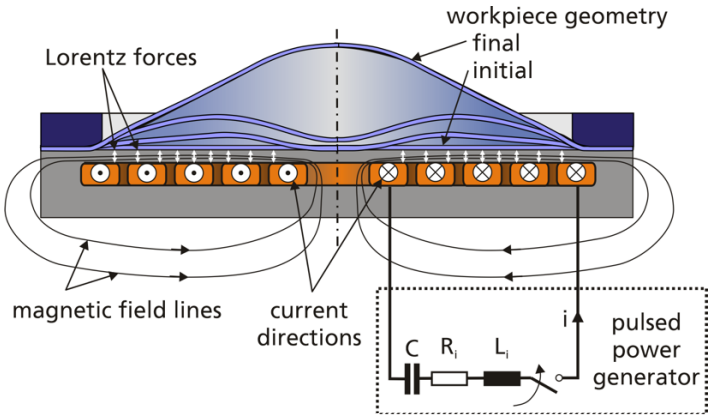


Figure 3: Principle sketch of an electromagnetic sheet metal forming process.

5.2 Electromagnetic forming - forming experiments

Forming experiments, with the in Chapter 0 presented EMF machine and monitoring setup, have been conducted. All in all, there are 4 adjustable input parameters, which directly influence the forming depth: the *yield strength* of the used workpiece material, the *distance* between the coil and workpiece and, the *capacitance* and *capacitor charging voltage* of the pulsed power generator. Table **1** presents the value domain of each input parameter resulting in a total of 36 feasible parameter combinations. The two resulting output values are the *current flow*,

measured by the Rogowski probe, and the *forming depth* of the workpiece, measured by the triangulation sensor. The former offers a deeper look into the actual energy transmission from the machine onto the workpiece, while the latter gives a precise numeric value of the true resulting forming of the workpiece. Figure 4 shows a plot of both measured output values for the input parameter combination 44 (yield strength 150 MPa, distance 1.5 mm, capacitance 180 µF and capacitor charging voltage 5.0 kV). The blue graph represents the current flow while the orange graph represents the forming depth over time. To fully grasp the process uncertainty multiple forming experiments are conducted for each parameter combination. In regard to parameter combination 44 the resulting forming depth of 5 different runs are {4.1; 3.66; 3.21; 3.93; 3.45} mm. Figure 5 shows a set of three workpieces after the experiment with different forming depths. The dataset obtained from the forming experiments is not publicly available but can be provided upon individual request.

Parameter	Values	Unit
Yield Strength	[125, 150]	MPa
Distance	[0.5, 1.5, 2.5]	mm
Capacitance	[180, 315]	µF
Charging Voltage	[5.0, 7.5, 10.0]	kV

Table 1: All possible parameter values for the forming experiments.

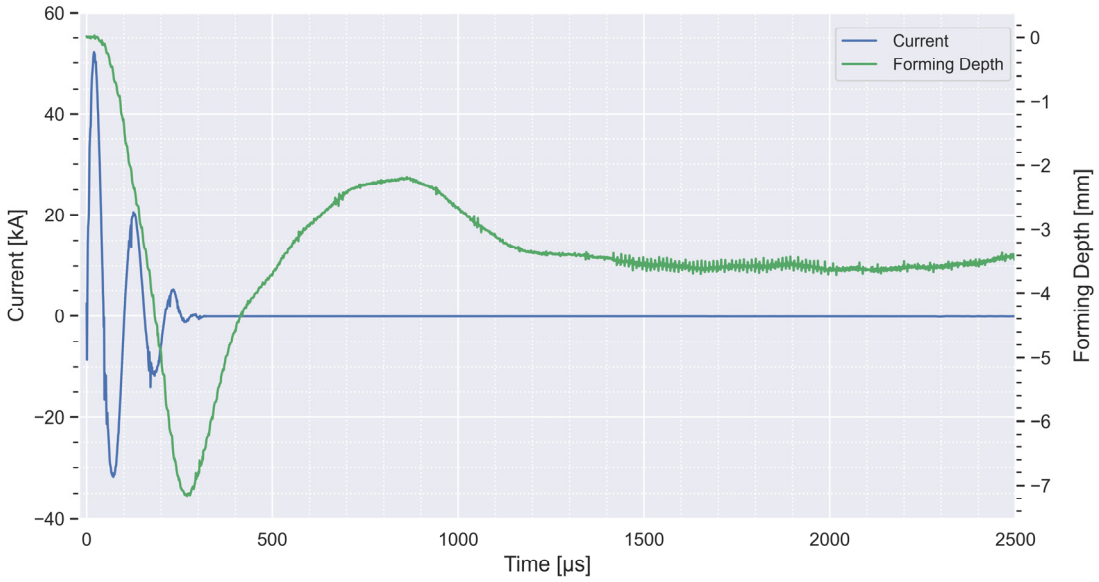


Figure 4: Measured current flow and forming during one electromagnetic impulse.

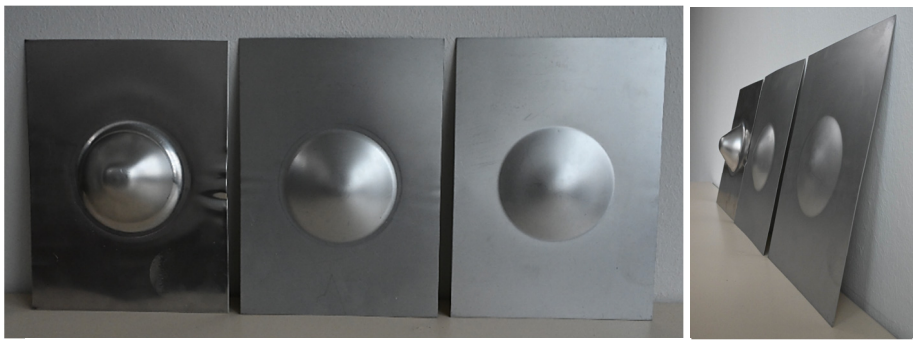


Figure 5: Examples of formed workpieces.

5.3 IPS-MCD experimental configuration and hypothesis

To validate the proposed IPS-MCD method, this study incorporates two distinct scenarios with varied experimental configurations, each associated with a specific hypothesis for proof. The model used for the validation is presented in detail in Chapter 5.4.

Scenario 1 – proof of concept

As a proof of concept, the total dataset collected from the initial forming experiments is divided into two equally sized subsets. The key distinction between them lies in the number of samples where the input parameter distance, describing the coil-to-workpiece distance, is set to 1.5 mm. Subset (A) contains significantly fewer samples with this parameter value compared to subset (B). Further details on the exact parameter distribution are provided in Chapter 5.5. The objective of this experimental configuration is to demonstrate that a lower representation of a specific parameter value in subset (A) leads to a higher uncertainty quantification compared to subset (B), thereby validating the effectiveness of the proposed IPS-MCD uncertainty quantification method.

Scenario 2 - strategic design of experiments

A key application of the proposed IPS-MCD method is the derivation from the uncertainty quantification for a strategic DoE in data-sparse environments. Scenario 2 illustrates this use case by applying IPS-MCD to the entire dataset, instead of subsets. The following analysis of the resulting uncertainty metrics (presented in Chapter 5.7 and Chapter 5.8) identifies specific parameter combinations that have the biggest impact on the model's prediction uncertainty. By strategically incorporating additional samples, collected from additional formation experiments, of these critical parameter combinations, the goal is to significantly reduce model uncertainty and improve model accuracy. This approach emphasizes targeted data generation rather than unsystematic or general sampling, thereby maximizing the efficiency and relevance of the collected data.

5.4 Model setup

For the verification of the proposed IPS-MCD method, the Temporal Fusion Transformer (TFT) [37] is employed as the base model. Originally developed for time-series forecasting, the TFT offers not only high predictive performance but also structural advantages that align with the characteristics of the given dataset and the implementation of IPS-MCD. A key feature of the TFT is its ability to incorporate static input parameters, allowing the integration of scalar machine and process parameters alongside time-dependent variables. Each static and dynamic input parameter is embedded into a separate input channel before being concatenated and processed by a variable selection network. This specific architectural feature of neural networks enables the input parameter-specific application of MCD. To adapt the TFT for the proposed approach, two modifications are introduced. First, the original multi-horizon forecasting capability of the TFT is restructured into a regression task [2], as the primary objective is the prediction of the formation depths. Second, the input channel architecture of the TFT is modified to enable the application of dropout during inference on pre-selected input channels, thereby allowing uncertainty estimation for specific input parameters.

The model generates quantile predictions and is thus trained using quantile loss as the objective function. The corresponding applied quantile values were $q \in [0.1, 0.5, 0.9]$, where the deterministic prediction is equivalent to $q_{0.5}$.

The hyperparameters of the TFT, specifically state size, number of attention heads, number of Long short-term memory layers, learning rate as well as the dropout probability, were

optimized using Optuna [38]. The optimized hyperparameters obtained were consistently employed throughout the study.

All in all, the input parameters for the TFT are the yield strength, distance, capacitance, charging voltage as static input parameters and the measured current as a variable input parameter. The output of the TFT is the corresponding formation depth obtained from the formation experiments. However, only the four static parameters are used for the proposed IPS-MCD.

5.5 Data setup

Scenario 1 – proof of concept

The primary challenge when splitting one dataset into two equally sized subsets with proportional parametric features, except for a specific nested parameter, leads to certain trade-offs. The goal of this proof of concept is to show a difference in the uncertainty quantification in regards to a specific input parameter value imbalance between two comparably equal datasets. Therefore, the complete dataset of 210 samples is divided into two subsets, each containing a total of 174 samples. Per random seed s each subset comprises a core set of shared samples, common to both subsets, along with a distinct set of unique samples specific to each subset. The unique samples within each subset primarily differ in the number of instances where the input parameter distance is set to 1.5 mm. Apart from this distinction, the parameter distributions for yield strength, capacitance, and charging voltage remain approximately equivalent across both subsets, subject to variations introduced by the random seed s. Merely, subset (A) contains fewer samples with the distance $\in \{1.5\}$ compared to subset (B). To maintain an equal total sample size across both subsets for training, validation, and testing, the additional samples with distance $\in \{1.5\}$ in subset (B) are counterbalanced by reducing equally the number of samples with distance $\in \{0.5, 2.5\}$. Consequently, subset (B) contains 19 additional samples with distance $\in \{1.5\}$ in the training set, 2 additional samples in the validation set, and 3 additional samples in the test set, with exact values subject to variations introduced by the random seed s. As a result, both subsets consist of 139 samples in the training set, 18 samples in the validation set, and 17 samples in the test set.

Scenario 2 – strategic design of experiments

For the second scenario all samples with a total of 210 are used as a base dataset and divided into 168 samples in the training set, 23 samples in the validation set, and 19 samples in the test set. After an initial uncertainty quantification by IPS-MCD is finalized, 29 more samples of overall three specific parameter combinations with the highest CV values are created via additional formation experiments as a strategically extended dataset. These 29 additional samples are split with a 75/25 ration onto the training and validation sets from the base set, not adding anything to the test sets. Resulting in a split of 189 samples in the training set, 31 samples in the validation set, and 19 samples in the test set. Then the uncertainty quantification via IPS-MCD is repeated.

5.6 Training setup

The hyperparameter optimization identified an optimal dropout value of 0.15. However, since its effectiveness in achieving the most accurate uncertainty quantification within the proposed IPS-MCD framework is not conclusively determined, the complete training process was repeated multiple times using dropout values from the set of values $D_{Input\ Channel} = \{0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5\}$. These modified dropout values

were exclusively applied to the input channel layers of the associated input parameters to assess their impact on model uncertainty estimation.

Scenario 1 – proof of concept

The training process was designed to ensure robustness as well as comparability in the outcomes. For each subset a total of 70 different random seeds were used to introduce variability in the data splits. Figure 2(b) presents a flowchart of the extended IPS-MCD with a multi random seed approach, where S represents the total number of random seeds used. For each seed s 10 full separate training loops of 1500 epochs each for subset (A) and subset (B) respectively were performed to explore diverse model initialization and data split states. Of these 10 training loops of each subset, only the best models were selected for inference where the minimal validation loss also is identical between subset (A) and subset (B). This guarantees that both models of each subset exhibit equal performance based on validation loss, thus ensuring a certain level of comparability. Additionally, early stopping, was implemented as a regularization technique to mitigate overfitting, where training was halted when the delta between training and validation loss exceeded a predefined threshold. This structured training setup ensured that model comparisons remained consistent, minimizing training variability and enabling a reliable assessment of parameter-specific uncertainties.

Scenario 2 – strategic design of experiments

The same training process described in scenario 1 is applied to scenario 2 with the sole difference being that as there are no data subsets only one model is trained for each of the 70 unique random seeds $s \in S$.

5.7 Evaluation metrics and inference setup

The evaluation metrics used for the uncertainty quantification and proof of the proposed IPS-MCD are twofold: The first and main metric is the CV (see Equation (7)) on the outcome distribution. It is derived from the test data, which undergoes $T = 30.000$ forward passes during inference, each using a different dropout mask during the MCD framework. This process introduces variations in the model predictions, generating an outcome distribution from 30.000 predictions for each individual test sample. This distribution is used to compute the CV. Unlike variance, which provides an absolute measure of spread, CV normalizes the standard deviation relative to the mean, allowing for a more scale-independent comparison across different formation depths. This procedure is repeated for each unique random seed s, resulting in a set of CV values that capture the variability in model predictions across different training initializations and data splits. This total procedure is performed separately on each input parameter and input channel. Since for every seed s the same model, data splits, and test samples are used for every analyzed input channel during the IPS-MCD, the results remain directly comparable, allowing for a consistent evaluation of the impact of individual parameters on model uncertainty.

Scenario 1 – proof of concept

To achieve uncertainty quantification for each specific input parameter and corresponding input parameter value (see Table 1), the process of performing T forward passes during inference is applied separately to each associated input channel. Consequently, the presence of four distinct input channels results in four independent IPS-MCD computations, enabling the uncertainty contributions associated with each input parameter.

Scenario 2 – strategic design of experiments

To achieve uncertainty quantification for each specific parameter combination, MCD is applied simultaneously to all four input channels, rather than individually. The resulting evaluation metrics are derived from the output distribution and subsequently assigned to each specific parameter combination. After conducting the process with a total of S different random seeds, a comprehensive evaluation of the underlying uncertainty for each parameter combination is obtained.

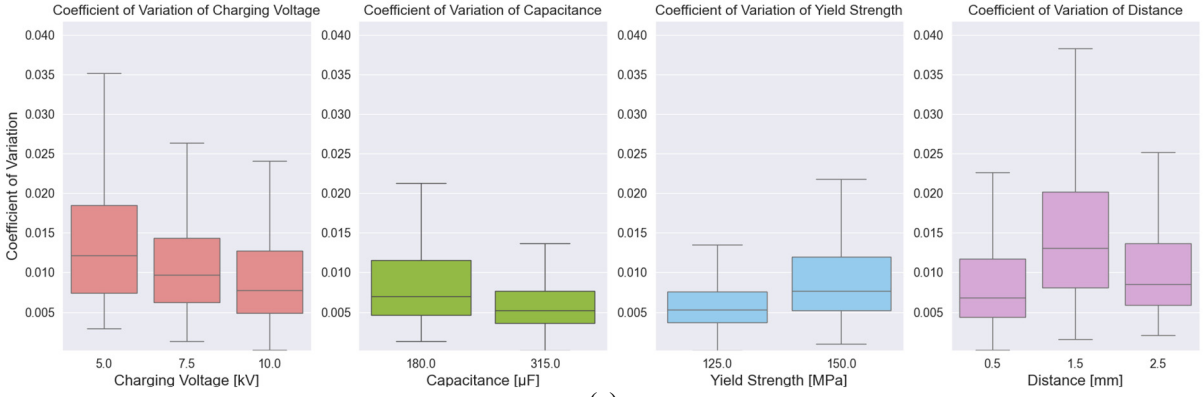
5.8 Results

Scenario 1 – proof of concept

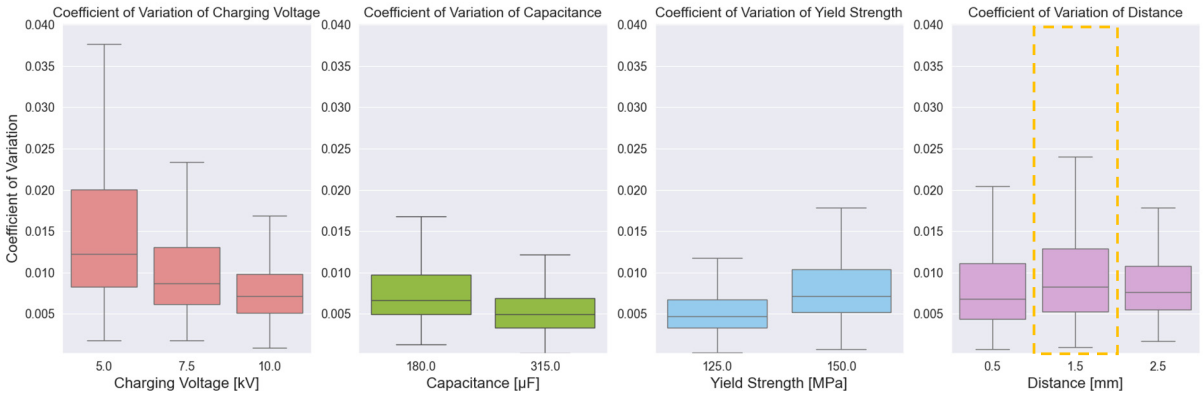
Figure 6 presents the primary results of the IPS-MCD for scenario 1, illustrating the uncertainty quantification of each input parameter value. The results are depicted as boxplots representing the distribution of collected CVs across all seeds for subset (A) in Figure 6(a) and subset (B) in Figure 6(b), with an input channel dropout value of 0.4. The first notable observation is a general conformity in the boxplots across all input parameters, except for the nested difference where distance equals to a value of 1.5 mm between subset (A) and subset (B) (see purple boxplot). As subset (B) contained a significantly higher number of samples where distance was equal to 1.5 mm, this disparity in parameter-specific sample representation (while maintaining equal total sample sizes) is reflected in the resulting model uncertainty quantification (see orange dotted box). This demonstrates the sensitivity of the proposed IPS-MCD framework to variations in input parameter distributions. Table 2 shows the results across all evaluated input channel dropout values. It contains specifically the median as well upper whisker values of the boxplots for the parameter distance with a value of 1.5 mm as well as the resulting percentage reduction. These results are consistent with those shown in Figure 6. The uncertainty represented as a boxplot of all collected CVs across all seeds is for the parameter distance with a value of 1.5 mm in subset (B) visibly lower, with an average reduction of 30%. Table 2 primarily emphasizes the nested differences between subset (A) and subset (B), as including boxplot values for all other input parameters exceeds the scope of this analysis. However, the results presented in Figure 6 for the dropout value of 0.4 are representative for the trends observed across the other dropout values analyzed in this study.

Scenario 2 – strategic design of experiments

Figure 7 displays the primary results of the IPS-MCD for scenario 2, providing the uncertainty quantification of each individual input parameter combination based on a dropout value of 0.4. The vertical axis corresponds to parameter pairs of yield strength and capacitance, whereas the horizontal axis represents parameter pairs of charging voltage and distance. Figure 7(a) specifically shows the results obtained from the base dataset. It is evident that relatively high CV values occur for the parameter combination comprising a yield strength of 150 MPa, capacitance of 180 µF, charging voltage of 5.0 kV, and distances equal to {0.5, 1.5, 2.5} mm, respectively. To explicitly address these comparatively high uncertainty values, an additional 29 samples were strategically incorporated into the base dataset, resulting in a strategically extended dataset as described in Chapter 5.5. These supplementary data include 10 samples for the previously identified parameter combination at a distance of 0.5 mm, 10 samples at 1.5 mm, and 9 samples at 2.5 mm. Figure 7(b) illustrates the results derived from this strategically augmented dataset. It is evident that the CV for these three parameter combinations is noticeably reduced. Meanwhile, the CV values for all remaining parameter combinations remain largely unchanged. Table 3 presents a deeper look into the improvements of the CV between the base and extended dataset on the key parameter combinations using other dropout values, and confirms the results.



(a)

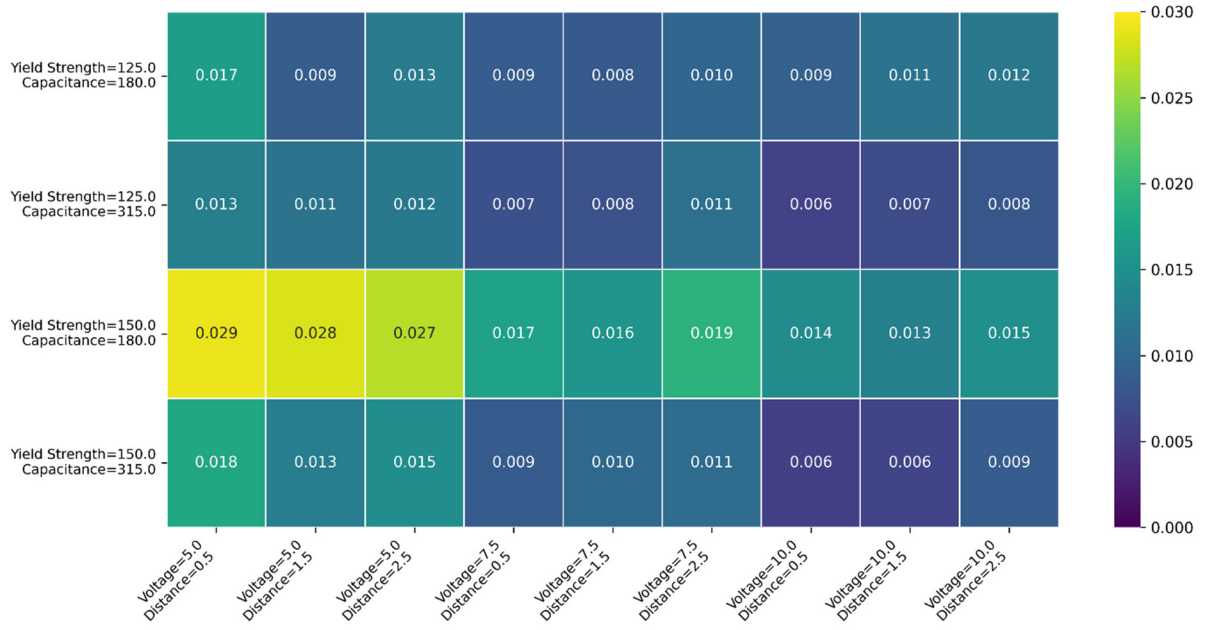


(b)

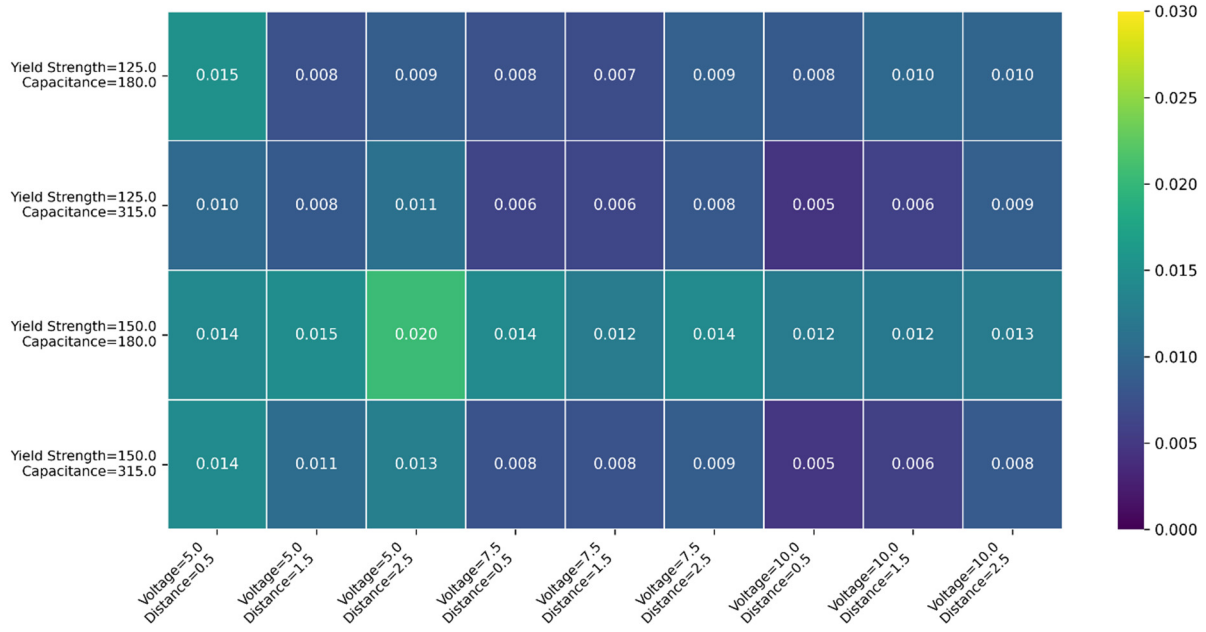
Figure 6: Coefficient of variation of each individual input parameter value using a dropout value of 0.4 for (a) subset (A) and (b) subset (B).

Dropout Value	Subset (A) [CV]		Subset (B) [CV]		Difference [%]
0.15	median	0.00791	median	0.00571	27.78
	upper whisker	0.02048	upper whisker	0.01555	24.04
0.20	median	0.00929	median	0.00601	35.26
	upper whisker	0.02632	upper whisker	0.01762	33.06
0.25	median	0.00965	median	0.00681	29.38
	upper whisker	0.02728	upper whisker	0.01915	29.80
0.30	median	0.00993	median	0.00748	24.63
	upper whisker	0.02842	upper whisker	0.01935	31.87
0.35	median	0.01147	median	0.00702	38.79
	upper whisker	0.02795	upper whisker	0.01994	28.63
0.40	median	0.01308	median	0.00826	36.83
	upper whisker	0.03826	upper whisker	0.02398	37.30
0.45	median	0.01452	median	0.00923	36.42
	upper whisker	0.04220	upper whisker	0.02599	38.39
0.50	median	0.01555	median	0.00962	38.09
	upper whisker	0.04491	upper whisker	0.02713	39.57

Table 2: Boxplot results where distance equals 1.5 mm between subset (A) and subset (B) along different dropout values.



(a)



(b)

Figure 7: Coefficient of variation heat map for every possible input parameter combination for the (a) base dataset and (b) strategically extended dataset

Dropout Value	Distance Parameter [mm]	Base Dataset [CV]	Strategically Extended Dataset [CV]	Difference [%]
0.15	0.5	0.021	0.007	66.67
	1.5	0.021	0.010	52.38
	2.5	0.015	0.012	20.00
0.20	0.5	0.022	0.008	63.64
	1.5	0.022	0.009	59.09
	2.5	0.018	0.011	38.89
0.25	0.5	0.023	0.011	52.17
	1.5	0.025	0.012	52.00
	2.5	0.021	0.012	42.86
0.30	0.5	0.025	0.011	56.00
	1.5	0.025	0.012	52.00
	2.5	0.020	0.016	20.00
0.35	0.5	0.026	0.014	46.15
	1.5	0.027	0.013	51.85
	2.5	0.026	0.017	34.62
0.40	0.5	0.029	0.014	51.72
	1.5	0.028	0.015	46.43
	2.5	0.027	0.020	25.93
0.45	0.5	0.029	0.018	37.93
	1.5	0.033	0.018	45.45
	2.5	0.032	0.022	31.25
0.50	0.5	0.035	0.019	45.71
	1.5	0.031	0.021	32.26
	2.5	0.034	0.027	20.59

Table 3: Comparison of coefficient of variation between base dataset and strategically extended dataset for the parameter combination {YS: 150 MPa, CA: 180 µF, V: 5.0 kV, D: (0.5, 1.5, 2.5) mm}

6 DISCUSSION

The experimental results presented in Chapter 5 demonstrate that the targeted application of Monte Carlo dropout exclusively on parameter-specific input channels within a deep learning network can effectively identify the contribution of individual input parameters to the overall model uncertainty. This approach can be utilized on specific parameter values, as presented in Figure 6, as well as parameter combinations, depicted in Figure 7. The resulting uncertainty quantification, which is represented by the coefficient of variation, can be leveraged for a strategic design of experiments. This capability is particularly valuable in sparse-data environments, where obtaining additional data can be time-consuming and/or costly.

Besides the presented proof through the targeted reduction in the uncertainty quantification, the results from the IPS-MCD also effectively represent the underlying uncertainty within the EMF process itself. Due to the internal measurement of the capacitor charging voltage within the pulsed power generator, the measured voltage value is subject to noise effects. As the measured voltage approaches the lower boundary of the measurement range (3.3 kV), the impact of noise on measurement accuracy becomes increasingly pronounced. Consequently, the coefficient of variation is generally elevated at lower capacitor charging energies (see Figure 6). Additionally, the capacitors experience inherent degradation, resulting in a

discrepancy between their nominal and actual capacitance values. Since the current through the inductor is the cumulative sum of the individual capacitor currents, slight variations in switching times can significantly affect the transient behavior of the current curve. With lower capacitance (fewer capacitors) the influence of both effects together has a greater relative influence on the inconsistencies of the transferable energy and thus the uncertainty within the resulting forming depth. Therefore, from a purely technological perspective, the parameters *charging voltage* and *capacitance* have per default a notable impact on the process uncertainty. The results of the proposed IPS-MCD are able to reflect this coherence.

7 CONCLUSION

This study successfully advances uncertainty quantification techniques in deep learning networks by strategically applying Monte Carlo dropout to solely input-specific channels. The proposed method enables a more detailed look into the underlying sources of uncertainty, especially in sparse-data environments. The resulting insight can then be utilized for a more strategic design of experiment, effectively saving time and financial resources in the generation of more data. By selectively targeting specific input parameters, the proposed method, which is introduced as input parameter-specific Monte Carlo dropout (IPS-MCD), accurately identifies parameter values as well as parameter combinations associated with high uncertainty. Consequently, it is able to quantify uncertainty in critical areas and enables a targeted reduction in uncertainty, thereby enhancing model reliability and predictive confidence.

8 ACKNOWLEDGEMENTS

The authors gratefully acknowledge funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - funding number IH 123/32-1.

REFERENCES

[1] E. Boos, M. Schwarzenberger, M. Jaretzki, and S. Ihlenfeldt, “Melt pool monitoring using fuzzy based anomaly detection in laser beam melting,” in *Proceedings of the Metal Additive Manufacturing Conference, Örebro, Sweden*, 2020, pp. 1–11.

[2] E. Boos, J. Zimmermann, H. Wiemer, and S. Ihlenfeldt, “Investigation of alternative attention modules in transformer models for remaining useful life predictions: addressing challenges in high-frequency time-series data,” *Procedia CIRP*, vol. 122, pp. 85–90, 2024, doi: 10.1016/j.procir.2024.01.012.

[3] C. Li, Y. Chen, and Y. Shang, “A review of industrial big data for decision making in intelligent manufacturing,” *Engineering Science and Technology, an International Journal*, vol. 29, p. 101021, May 2022, doi: 10.1016/j.jestch.2021.06.001.

[4] A. Schimke, O. Gnepper, and E. Boos, “Data-Driven Damage Assessment of Axial Piston Pumps,” in *2024 IEEE International Conference on Engineering, Technology, and Innovation (ICE/ITMC)*, Funchal, Portugal: IEEE, Jun. 2024, pp. 1–9. doi: 10.1109/ICE/ITMC61926.2024.10794220.

[5] X. Bai *et al.*, “Explainable deep learning for efficient and robust pattern recognition: A survey of recent developments,” *Pattern Recognition*, vol. 120, p. 108102, Dec. 2021, doi: 10.1016/j.patcog.2021.108102.

[6] M. Abdar *et al.*, “A review of uncertainty quantification in deep learning: Techniques, applications and challenges,” *Information Fusion*, vol. 76, pp. 243–297, Dec. 2021, doi: 10.1016/j.inffus.2021.05.008.

[7] H. Li, J. Jiao, Z. Liu, J. Lin, T. Zhang, and H. Liu, “Trustworthy Bayesian deep learning framework for uncertainty quantification and confidence calibration: Application in machinery fault diagnosis,” *Reliability Engineering & System Safety*, vol. 255, p. 110657, Mar. 2025, doi: 10.1016/j.ress.2024.110657.
[8] A. Kendall and Y. Gal, “What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?”.
[9] E. Hüllermeier and W. Waegeman, “Aleatoric and Epistemic Uncertainty in Machine Learning: An Introduction to Concepts and Methods,” *Mach Learn*, vol. 110, no. 3, pp. 457–506, Mar. 2021, doi: 10.1007/s10994-021-05946-3.
[10] L. Basora, A. Viens, M. A. Chao, and X. Olive, “A benchmark on uncertainty quantification for deep learning prognostics,” *Reliability Engineering & System Safety*, vol. 253, p. 110513, Jan. 2025, doi: 10.1016/j.ress.2024.110513.
[11] A. Loquercio, M. Segu, and D. Scaramuzza, “A General Framework for Uncertainty Estimation in Deep Learning,” *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 3153–3160, Apr. 2020, doi: 10.1109/LRA.2020.2974682.
[12] Y. Gal and Z. Ghahramani, “Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning,” in *Proceedings of The 33rd International Conference on Machine Learning*, PMLR, Jun. 2016, pp. 1050–1059. Accessed: Jan. 28, 2025. [Online]. Available: https://proceedings.mlr.press/v48/gal16.html
[13] M. Valdenegro-Toro and D. Saromo, “A Deeper Look into Aleatoric and Epistemic Uncertainty Disentanglement,” Apr. 20, 2022, *arXiv*: arXiv:2204.09308. doi: 10.48550/arXiv.2204.09308.
[14] V. Psyk, D. Risch, B. L. Kinsey, A. E. Tekkaya, and M. Kleiner, “Electromagnetic forming—A review,” *Journal of Materials Processing Technology*, vol. 211, no. 5, pp. 787–829, May 2011, doi: 10.1016/j.jmatprotec.2010.12.012.
[15] J. Gawlikowski *et al.*, “A Survey of Uncertainty in Deep Neural Networks,” Jan. 18, 2022, *arXiv*: arXiv:2107.03342. doi: 10.48550/arXiv.2107.03342.
[16] J. Gao, M. Chen, L. Xiang, and C. Xu, “A Comprehensive Survey on Evidential Deep Learning and Its Applications,” Sep. 07, 2024, *arXiv*: arXiv:2409.04720. doi: 10.48550/arXiv.2409.04720.
[17] A. Malinin and M. Gales, “Predictive Uncertainty Estimation via Prior Networks,” Nov. 29, 2018, *arXiv*: arXiv:1802.10501. doi: 10.48550/arXiv.1802.10501.
[18] J. van Amersfoort, L. Smith, Y. W. Teh, and Y. Gal, “Uncertainty Estimation Using a Single Deep Deterministic Neural Network”.
[19] J. Cheung, S. Rangarajan, A. Maddocks, X. Chen, and R. Chandra, “Quantile deep learning models for multi-step ahead time series prediction,” Nov. 24, 2024, *arXiv*: arXiv:2411.15674. doi: 10.48550/arXiv.2411.15674.
[20] D. J. C. MacKay, “Bayesian neural networks and density networks,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 354, no. 1, pp. 73–80, Jan. 1995, doi: 10.1016/0168-9002(94)00931-7.
[21] T. Huix, S. Majewski, A. Durmus, E. Moulines, and A. Korba, “Variational Inference of overparameterized Bayesian Neural Networks: a theoretical and empirical study,” Jul. 08, 2022, *arXiv*: arXiv:2207.03859. doi: 10.48550/arXiv.2207.03859.
[22] A. Shamsi, H. Asgharnezhad, A. Tajally, S. Nahavandi, and H. Leung, “An Uncertainty-aware Loss Function for Training Neural Networks with Calibrated Predictions,” Feb. 06, 2023, *arXiv*: arXiv:2110.03260. doi: 10.48550/arXiv.2110.03260.
[23] B. Lakshminarayanan, A. Pritzel, and C. Blundell, “Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles,” Nov. 04, 2017, *arXiv*: arXiv:1612.01474. doi: 10.48550/arXiv.1612.01474.
[24] I. Ezhov *et al.*, “Geometry-Aware Neural Solver for Fast Bayesian Calibration of Brain Tumor Models,” *IEEE Transactions on Medical Imaging*, vol. 41, no. 5, pp. 1269–1278, May 2022, doi: 10.1109/TMI.2021.3136582.
[25] B. C. Haas, D. Kalyani, and M. S. Sigman, “Applying statistical modeling strategies to sparse datasets in synthetic chemistry,” *Sci. Adv.*, vol. 11, no. 1, p. eadt3013, Jan. 2025, doi: 10.1126/sciadv.adt3013.

[26] H. Song *et al.*, “Adaptive Experiments Under High-Dimensional and Data Sparse Settings: Applications for Educational Platforms,” 2025, *arXiv*. doi: 10.48550/ARXIV.2501.03999.
[27] I. Steponavičė, M. Shirazi-Manesh, R. J. Hyndman, K. Smith-Miles, and L. Villanova, “On Sampling Methods for Costly Multi-Objective Black-Box Optimization,” in *Advances in Stochastic and Deterministic Global Optimization*, vol. 107, P. M. Pardalos, A. Zhigljavsky, and J. Žilinskas, Eds., in Springer Optimization and Its Applications, vol. 107. , Cham: Springer International Publishing, 2016, pp. 273–296. doi: 10.1007/978-3-319-29975-4_15.
[28] A. Hoarau, B. Quost, S. Destercke, and W. Waegeman, “Reducing Aleatoric and Epistemic Uncertainty through Multi-modal Data Acquisition,” Jan. 30, 2025, *arXiv*: arXiv:2501.18268. doi: 10.48550/arXiv.2501.18268.
[29] T. Blau, E. V. Bonilla, I. Chades, and A. Dezfouli, “Optimizing Sequential Experimental Design with Deep Reinforcement Learning”.
[30] K. Chaloner and I. Verdinelli, “Bayesian Experimental Design: A Review,” *Statist. Sci.*, vol. 10, no. 3, Aug. 1995, doi: 10.1214/ss/1177009939.
[31] T. Rainforth, A. Foster, D. R. Ivanova, and F. B. Smith, “Modern Bayesian Experimental Design,” Nov. 29, 2023, *arXiv*: arXiv:2302.14545. doi: 10.48550/arXiv.2302.14545.
[32] P. Novello, G. Poëtte, D. Lugato, and P. Congedo, “Leveraging Local Variation in Data: Sampling and Weighting Schemes for Supervised Deep Learning,” *J Mach Learn Model Comput*, vol. 3, no. 1, pp. 77–105, 2022, doi: 10.1615/JMachLearnModelComput.2022041819.
[33] T. Xie *et al.*, “Structural-Entropy-Based Sample Selection for Efficient and Effective Learning,” Oct. 05, 2024, *arXiv*: arXiv:2410.02268. doi: 10.48550/arXiv.2410.02268.
[34] J. Kukačka, V. Golkov, and D. Cremers, “Regularization for Deep Learning: A Taxonomy,” Oct. 29, 2017, *arXiv*: arXiv:1710.10686. doi: 10.48550/arXiv.1710.10686.
[35] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”.
[36] Z. Lai *et al.*, “A comprehensive electromagnetic forming approach for large sheet metal forming,” *Procedia Engineering*, vol. 207, pp. 54–59, 2017, doi: 10.1016/j.proeng.2017.10.737.
[37] B. Lim, S. O. Arik, N. Loeff, and T. Pfister, “Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting,” *arXiv:1912.09363 [cs, stat]*, Sep. 2020, Accessed: May 04, 2022. [Online]. Available: http://arxiv.org/abs/1912.09363
[38] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, “Optuna: A Next-generation Hyperparameter Optimization Framework,” Jul. 25, 2019, *arXiv*: arXiv:1907.10902. Accessed: Nov. 10, 2023. [Online]. Available: http://arxiv.org/abs/1907.10902

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

A MULTI-FIDELITY REDUCED-ORDER MODEL TO QUANTIFY AERODYNAMIC FORCES ON A VERTICAL-AXIS WIND TURBINE WITH UNCERTAIN ROTATIONAL SPEED

Fausto Dicech[1] **and Lucia Parussini**[1]

[1]Università degli Studi di Trieste, Department of Engineering and Architecture
Via Alfonso Valerio 10, Trieste, Italy
e-mail: fausto.dicech@phd.units.it, lparussini@units.it

Abstract. *Wind turbines experience various uncertainties that can impact both design and control. In particular, rotational velocity is a crucial parameter, often controlled to ensure a stable and optimal power output or to avoid turbine overspeeding. This work investigates uncertainty quantification (UQ) for a vertical-axis wind turbine (VAWT), with a focus on the Darrieus VAWT type, using a multi-fidelity (MF) non-intrusive reduced order model (ROM), parameterized on the tip-speed ratio (TSR). A 2D computational fluid dynamic (CFD) model gives the high-fidelity (HF) representation of a three-bladed Darrieus VAWT with NACA0015 airfoils. A similar model with a coarser computational mesh represents the low-fidelity (LF) one. Both allow us to study the aerodynamic forces acting on the blades at different azimuthal angles for a wide range of TSRs. However, multiple evaluations of the HF model can become excessively expensive due to high computational costs, making demanding operations such as UQ not always a feasible choice. ROMs, on the other hand, can compute HF approximations in almost real-time. However, ROMs can be burdensome due to their high off-line computational costs relative to the training phase. Multi-fidelity methods extended to ROMs aim to reduce the amount of HF data required, leveraging inexpensive LF information. The present study compares the effectiveness of a MF ROM to more classical UQ approaches, such as Polynomial Chaos Expansion (PCE) on the HF solutions, and whether it is advantageous to equivalent single-fidelity ROMs. In particular, the focus is on the effect of rotational speed uncertainty on the aerodynamic forces acting on the blades, which are pivotal for turbine performances. Moreover, evaluating a ROM multiple times enables slow-convergence methods, such as Monte Carlo, with minimal effort. The MF ROM showed good UQ performance when compared to an equivalent single-fidelity ROM, especially for the mean estimates, closing the gap with a 4^{th} degree PCE in the proposed use-case.*

Keywords: Multi-fidelity, Uncertainty Quantification, Reduced Order Models, Vertical Axis Wind Turbine.

1 INTRODUCTION

Uncertainty quantification (UQ) plays an important role whenever there is limited knowledge about the problem of interest [1]. In general, uncertainties have an inherently stochastic nature and can originate from multiple sources, where the most common are: measurements, physical modelling of the phenomena, geometrical tolerances, or computational approximations.

Many uncertainties affect wind turbines, both in their design process and control. Different approaches have been used to study UQ for wind turbines [2, 3]. Few studies used surrogate models, such as non-intrusive Reduced Order Models (ROMs), to perform UQ on wind turbines, and the literature is especially lacking for what concerns vertical axis wind turbines (VAWTs). Most of the works use ROMs only to approximate some of the VAWTs flow features [4, 5], such as the flow around an airfoil or the wake, but they never consider UQ problems.

In this work we want to assess the UQ capabilities of a three-bladed Darrieus VAWT multi-fidelity (MF) ROM [6]. In particular, we aim to study the uncertainty introduced by the rotational velocity on the aerodynamic forces acting on the turbine blades during an entire revolution. Rotational velocity is a crucial control parameter for VAWTs, and wind turbines in general, since it ensures a stable and optimal power output, preventing the turbine from over-speeding and ensuring grid system compatibility [7, 8]. VAWT control can be achieved in different ways, such as through blade pitch control, mechanical brakes, electric dump loads, or variable rotor resistance [9, 10, 11]. The thrust force component is the sole contributor to the shaft's torque, making it responsible for the entire VAWT power generation, and it is affected by rotational speed. As a consequence, power output can vary and fluctuate due to uncertainties in the rotational speed.

This work utilizes a non-intrusive ROM to predict the forces acting on a Darrieus VAWT's blades with a parametric tip-speed ratio (TSR). Non-intrusive ROMs [12, 13] are used in many fields, such as optimization and various engineering applications, to approximate distributed outputs of numerical models, such as solution fields of computational fluid dynamics (CFD) problems or any vectorized quantity. Here, they enable almost real-time multiple observations of different configurations for the three-bladed Darrieus VAWT. This allows to perform effortless UQ on the uncertain input parameter. The ROM has been trained with the results of 2D transient CFD simulations of the VAWT flow, in particular the force profiles obtained during an entire turbine revolution. Since ROMs have significant offline computational costs due to CFD simulations, MF techniques aim to improve the accuracy of the ROM while limiting the cost. Indeed, MF models leverage inexpensive and abundant low-fidelity (LF) information, i.e. coarse grid simulations, to enhance the high-fidelity (HF) problem understanding, by learning the relationship between LF and HF data.

The paper is organized as follows. Section 2 discusses the proposed methodology, Section 3 presents the use-case and analyses the results, finally Section 4 outlines the conclusions and suggest future developments.

2 METHODOLOGY

In Section 2 the methodology used will be presented, starting from what ROMs are in Section 2.1 and how their MF extension is obtained. Successively, in Section 2.2 the UQ will be

discussed more in depth.

2.1 Parametrized reduced order models

Traditionally, a ROM aims to approximate the solution of a set of parametrized partial differential equations (PDEs), such as in CFD problems. Even though ROMs are not as accurate as the full order model (FOM) used to solve the PDE problem, the predictions for any given parameter combination can be obtained in almost real-time. Usually, ROMs are categorized as intrusive and non-intrusive ROMs. The former are more accurate, but they are not always a feasible solution, as they require access to the FOM source code. The latter assume that the FOM is a black-box, requiring only FOM solutions for some parameter configurations during training. In this work, we focus on the non-intrusive formulation.

2.1.1 Non-intrusive reduced order models

Let's have a set of snapshots, i.e. the FOM solution fields, $\mathbf{S} = \{\mathbf{u}_1(\boldsymbol{\mu}_1), ..., \mathbf{u}_n(\boldsymbol{\mu}_n)\}$ with $\mathbf{u}_i \in \mathbb{R}^N$ and $\boldsymbol{\mu}_i \in \mathbb{R}^d$ for $i = 1, ..., n$, where N is the snapshot dimension, n is the number of training snapshots, $\boldsymbol{\mu}$ is the input parameter and d is the input parameter space dimension. By applying proper orthogonal decomposition (POD) [14, 15] on the snapshot matrix $\mathbf{S}$, a set $\boldsymbol{\Phi} = \{\boldsymbol{\varphi}_1, ..., \boldsymbol{\varphi}_n\}$ of orthogonal basis function in $\mathbb{R}^N$ can be obtained, so that

$$\mathbf{S}\mathbf{S}^T\boldsymbol{\varphi}_j = \lambda_j^2\boldsymbol{\varphi}_j \quad \forall j \in 1, ..., n \tag{1}$$

where $\boldsymbol{\varphi}$ is a basis function, also called POD mode, λ_j is the singular value for the j-th mode and it is assumed that $\lambda_j \geq \lambda_{j+1}\ \forall j \in 1, ..., n-1$. Eventually, it is possible to truncate the set $\boldsymbol{\Phi}$ to the r-th mode, with $r < n$, since modes associated to higher values of λ^2 carry less information.

With the basis functions is possible to obtain a reduced representation $\mathbf{a} \in \mathbb{R}^r$, with $r \ll N$, of any snapshot $\mathbf{u}(\boldsymbol{\mu})$ by projection.

$$a_j = \mathbf{u}(\boldsymbol{\mu}) \cdot \boldsymbol{\varphi}_j \quad \forall j = 1, ..., r \tag{2}$$

Therefore, for all the snapshots belonging to $\mathbf{S}$ it is possible to compute their projection $\mathbf{a}_i$ onto the reduced basis, referred to as POD coefficients

$$\mathbf{a}_i = \{a_{i,1}, ...a_{i,r}\} = \{a_1(\boldsymbol{\mu}_i), ...a_r(\boldsymbol{\mu}_i)\} \quad \forall i = 1, ..., n \tag{3}$$

and fit a regression model $\hat{a}_j(\boldsymbol{\mu})\ \forall j = 1, ..., r$ on the pairs $(\boldsymbol{\mu}_i, a_{i,j})$, with $i = 1, ..., n$, to approximate the a_j POD coefficient for any value of the input parameter $\boldsymbol{\mu}$. Since the modes span a subspace of $\mathbb{R}^N$, the ROM prediction can be computed through a linear combination of the modes and the POD coefficients models, as follows

$$\mathbf{u}(\boldsymbol{\mu}) \sim \sum_{j=1}^{r} \hat{a}_j(\boldsymbol{\mu})\boldsymbol{\varphi}_j \tag{4}$$

Any interpolation or regression model can be used to approximate the POD coefficients models in Equation 4. The most common ones are radial basis functions (RBF), Gaussian processes (GP) regression, or neural networks (NN).

2.1.2 Multi-fidelity reduced order models

The non-intrusive ROM structure can be extended to a MF approach, where two or more FOM with different levels of accuracy are used to approximate the same PDE problem. In this framework MF models replace classical single-fidelity models for the POD coefficients $\hat{a}_j(\boldsymbol{\mu})$ in Equation 4.

A MF regression model [16, 17, 18] requires at least two information sources, an HF one and a LF one. In general, HF data is more accurate but computationally expensive, while LF data is less accurate but significantly cheaper to obtain. Therefore, if LF information can be used to enhance the knowledge of the HF problem, the overall data computational cost can be reduced without sacrificing the accuracy. To apply MF to POD coefficients, a LF and HF representation a_j^{LF}, a_j^{HF} of the POD coefficients $a_j \, \forall j = 1, ..., r$ is needed. There are different ways to obtain a_j^{LF}, a_j^{HF}, and for the purpose of this work we follow [6].

Firstly, we define two snapshot matrices, a LF one $\mathbf{S}^{LF} = \{\mathbf{u}_1^{LF}, ..., \mathbf{u}_{n_L}^{LF}\}$ and a HF one $\mathbf{S}^{HF} = \{\mathbf{u}_1^{HF}, ..., \mathbf{u}_{n_H}^{HF}\}$, with $n_H \ll n_L$. All the snapshots are required to have the same cardinality $\mathbf{u}^{LF}, \mathbf{u}^{HF} \in \mathbb{R}^N$ and must be consistent in terms of connectivity. Otherwise, the snapshots should be mapped on a reference discretization, such as a computational mesh or a node distribution.

Successively, as in Equation 1, the snapshots need to be encoded with POD. The best results for MF ROMs are usually achieved mixing LF and HF snapshots. The matrix of snapshots $\mathbf{S}$ used in Equation 1 will be

$$\mathbf{S} = \mathbf{S}^{LF} \cup \mathbf{S}^{HF} \tag{5}$$

and the POD coefficients will be obtained for each POD mode by projecting the different fidelities snapshots separately on the mixed fidelity basis functions

$$a_{i,j}^{HF} = \mathbf{u}_i^{HF} \cdot \boldsymbol{\varphi}_j \quad \forall i = 1, ..., n_H \; \forall j = 1, ..., r \tag{6}$$

$$a_{k,j}^{LF} = \mathbf{u}_k^{LF} \cdot \boldsymbol{\varphi}_j \quad \forall i = 1, ..., n_L \; \forall j = 1, ..., r \tag{7}$$

Given the HF and LF POD coefficients, $a_{i,j}^{HF}, a_{k,j}^{LF}$, corresponding to the training snapshots, a MF regression model can be trained for each mode. In this work, the Non-linear AutoRegressive multi-fidelity GP (NARGP) regression model [18] was chosen for its flexibility in capturing non-linear correlation patterns between fidelities.

An expected improvement of a MF ROM over single-fidelity one is to achieve more accurate predictions given the same number of HF snapshots. This is obtained through the combination of the POD basis $\boldsymbol{\varphi}$ obtained mixing snapshots from different fidelities,allowing for better representation of unseen HF snapshots, and the greater NARGP accuracy when approximating the POD coefficients. For further details on MF ROMs in general, as well as the specific methodology used in this work, we refer the interested reader to [6].

2.2 Uncertainty quantification

2.2.1 Polynomial Chaos Expansion

Orthogonal polynomials are used as a basis to approximate the functional form between the stochastic response output and each of its random inputs. The chaos expansion for a response

Y takes the form

$$Y(\xi) \cong \sum_{j=0}^{\infty} \alpha_j \psi_j(\xi) \tag{8}$$

with ψ_j the orthogonal polynomial of order j, α_j the chaos coefficients and ξ a random variable. From a practical standpoint, the infinite expansion is truncated at a finite expansion order, leading to

$$Y(\xi) \cong \sum_{j=0}^{P} \alpha_j \psi_j(\xi) \tag{9}$$

PCE includes a complete basis of polynomials up to a fixed total-order specification, and in this case the total number of terms N_t, in an expansion of total order p involving n random variables is given by

$$N_t = 1 + P = 1 + \sum_{s=1}^{p} \frac{1}{s!} \prod_{r=0}^{s-1} (n+r) = \frac{(n+p)!}{n!p!} \tag{10}$$

Stochastic Collocation technique [19] can be used to determine α_j coefficients with $j = 0, ..., P$. Once the chaos coefficients are known, a direct estimation of the systems' statistics can be done. The mean $\mathbb{E}[Y(\xi)]$ is analytically computed as

$$\mathbb{E}[Y(\xi)] \sim \sum_{j=0}^{P} \alpha_j \langle \psi_j(\xi) \rangle = \alpha_0 \tag{11}$$

and the variance σ_Y^2 as

$$\sigma_Y^2 = \mathbb{E}[Y(\xi)^2] - \mathbb{E}[Y(\xi)]^2 \sim \sum_{i=0}^{P} \sum_{j=0}^{P} \alpha_i \alpha_j \langle \psi_i(\xi) \psi_j(\xi) \rangle - \alpha_0^2 = \sum_{j=1}^{P} \alpha_j^2 \left\langle \psi_j^2(\xi) \right\rangle \tag{12}$$

where $\mathbb{E}[\cdot]$ is the expected value. In this work, ξ follows a Gaussian distribution, therefore the appropriate orthogonal polynomial ψ_j choice for the expansion is the Hermite polynomials.

2.2.2 Monte Carlo

Classical Monte Carlo (MC) methods for UQ are used for uncertainty propagation [20]. If the model's input parameter uncertainties are represented by a random variable ξ, a MC method aims to approximate the stochastic response output Y of the model. Indeed, to approximate the expected value $\mathbb{E}[Y(\xi)]$, MC methods requires to sample M times the ξ random variable and compute as many deterministic responses from the model $y_1, ..., y_M$. The $\mathbb{E}[Y(\xi)]$ is approximated with the average of the outputs, as in

$$\mathbb{E}[Y(\xi)] \sim \bar{y} = \frac{1}{M}\sum_{i=1}^{M} y_i \tag{13}$$

and the variance σ_Y^2 can be estimated as well, as

$$\sigma_Y^2 = \mathbb{E}[Y(\xi)^2] - \mathbb{E}[Y(\xi)]^2 \sim \frac{1}{M}\sum_{i=1}^{M}(y_i - \bar{y})^2 \tag{14}$$

Due to the strong law of large numbers, $\bar{y}$ should converge to the actual expected value $\mathbb{E}[Y(\xi)]$ with enough samples M. However, the convergence rate is rather slow, being $\mathcal{O}(\sqrt{M})$.

3 RESULTS

In Section 3 the results will be presented, starting from the case-study description in 3.1 and then moving to the actual results analysis in Section 3.2

3.1 Problem description

In this work the force profiles during an entire revolution of a VAWT are studied under rotational speed uncertainty. In particular, the considered turbine is a Darrieus three-bladed VAWT with NACA0015 profiles, each with a chord length of 420 mm, as in [21]. The CFD simulations simplify the problems to be two-dimensional, with a constant wind speed $U = 10$ m/s. The rotational speed of the turbine is the only control parameter, ensuring values of the TSR $\lambda \in [0.9, 1.32]$, defined in Equation 15.

$$\lambda = \frac{\omega R}{U} \tag{15}$$

where R is the radius between the turbine blade and the center of rotation, and ω is the rotational speed. Instead of approximating CFD solution fields on the spatial discretization, the ROM will be utilized to surrogate the thrust force F_T acting on a blade, as shown in Figure 1, on the time discretization. Specifically, the time values will be matched with their corresponding azimuthal angle θ discretization.

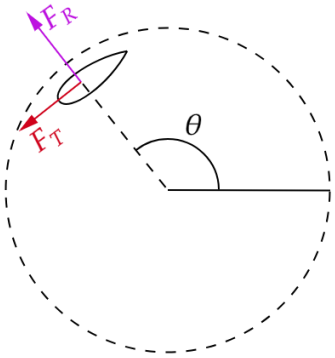


Figure 1: Thurst and radial forces F_T, F_R components acting on a VAWT blade at an angle θ.

This work will focus mainly on the F_T component, since it is the sole contributor to the power generation and it is more relevant for this type of application. The radial component F_R remains crucial from a structural design perspective. Figure 2 shows examples of force profiles obtained from simulations at different λ and will be used as snapshots for the ROMs.

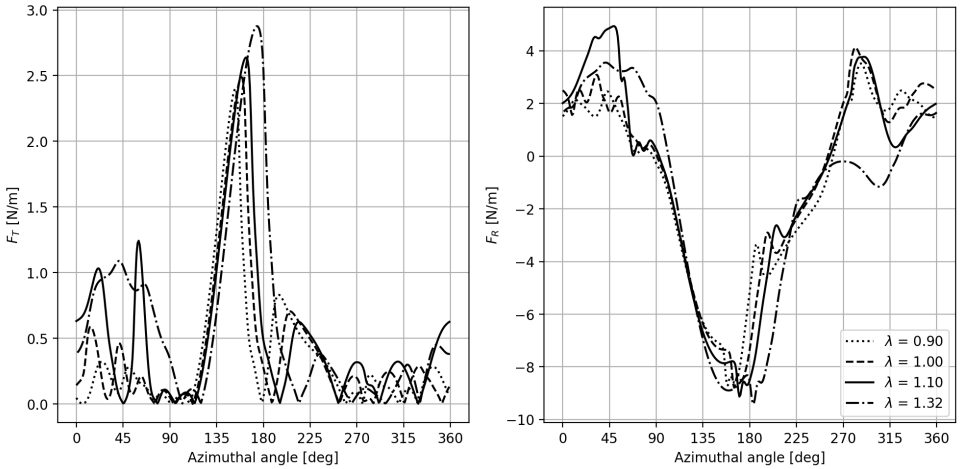


Figure 2: Thurst (left) and radial (right) forces components during an entire revolution for different values of λ.

3.1.1 Simulation set-up

The computation has been carried out with Ansys Fluent, solving for the transient RANS equations alongside the $k\omega$-SST turbulence model. The blades rotation was modelled with a cylindrical sliding mesh centred on the rotational axis of the VAWT. Figure 3 shows a schematic representation of the domain. Many works discuss in detail the best procedures and reccommendations to set-up a 2D VAWT CFD simulation [22].

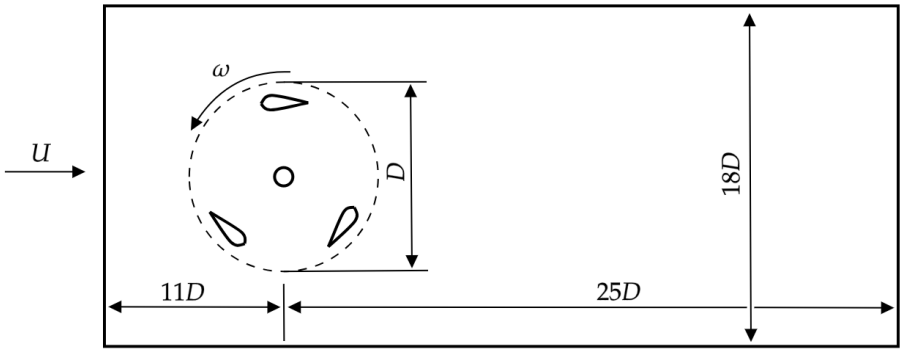


Figure 3: Schematics of the 2D CFD domain. $D = 2.8$ m here represents the diameter of the rotating mesh region.

The LF and HF solutions differ in their computational mesh granularity, where the LF grid is coarser, with 107k cells, while the HF one has 411k cells, as Figure 4 shows. Time discretization depends solely on ω, ensuring a rotation of 1° per time-step in both LF and HF simulations.

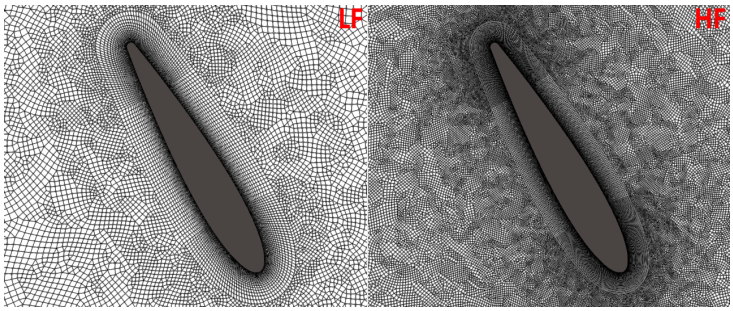


Figure 4: Close up of the computational mesh around one blade for a LF solution (left) and for an HF solution (right).

The HF results are reasonably consistent with the reference experimental measurements [21]. Figure 5 shows a power coefficient value comparison between the HF CFD and the experimental results. The power coefficient represents the performance of the turbine and is computed as in Equation 16. The slight CFD overestimation can be attributed to the 2D simplification of the problem, where the degrading effects of 3D flow features, such as tip vortices or the fluid interaction with the support structures, cannot be modelled.

$$C_P = \frac{P_t}{P_w} = \frac{\omega R N_b \frac{1}{2\pi} \int_0^{2\pi} F_T(\theta) d\theta}{\frac{1}{2}\rho U^3 2RH} \tag{16}$$

where P_t is the turbine power, P_w is the wind power, N_b is the number of blades, F_T is the thrust component of the aerodynamic force acting on a single blade, θ is the azimuthal angle of the blade, ρ is the air density and H is the height of the wind turbine's swept area.

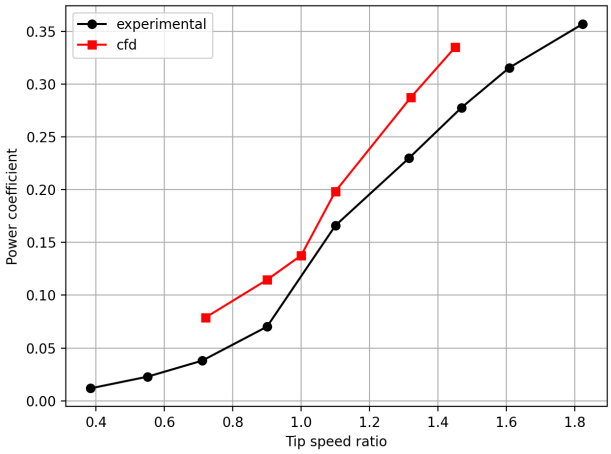


Figure 5: Power coefficient measurements from [21] vs current work HF CFD results.

3.1.2 Uncertain variable

As previously mentioned, in this work we consider the rotational speed ω to be the uncertain variable. Since the wind speed U is kept constant, the TSR λ is just a scaled version of ω, and are interchangeable.

Rotational speed in wind turbines, or in any low-speed rotor in general, can be measured using different sensors and techniques, such as Hall-effect sensors, tachometers or image-processing methodologies [23, 24, 25]. For the purpose of this work, we assume an accuracy of ± 0.5 rpm for the VAWT's hypothetical rotational speed measurement. In fact, under the considered operating conditions the turbine rotates at a speed of approximately $70 - 100$ rpm, and a $0.5 - 1\%$ error is close to what has been observed in literature. Therefore, the UQ is performed under the hypothesis that ω follows a Gaussian distribution with a standard deviation $\sigma = 0.25$ rpm, having the 95% of the distribution lying within 1 rpm of the mean.

3.2 Analysis and results

Considering $\lambda = 1.25$, or alternatively $\omega = 85.3$ rpm, we want to compare a 4^{th} degree PCE mean and second order statistical moment estimates to the ones obtained with a MC simulation performed with single- and multi- fidelity ROMs. The 4^{th} degree PCE will be used as the ground truth throughout this work.

Table 1 presents the HF solutions computed for the ROM design of experiment (DoE) and for the PCE.

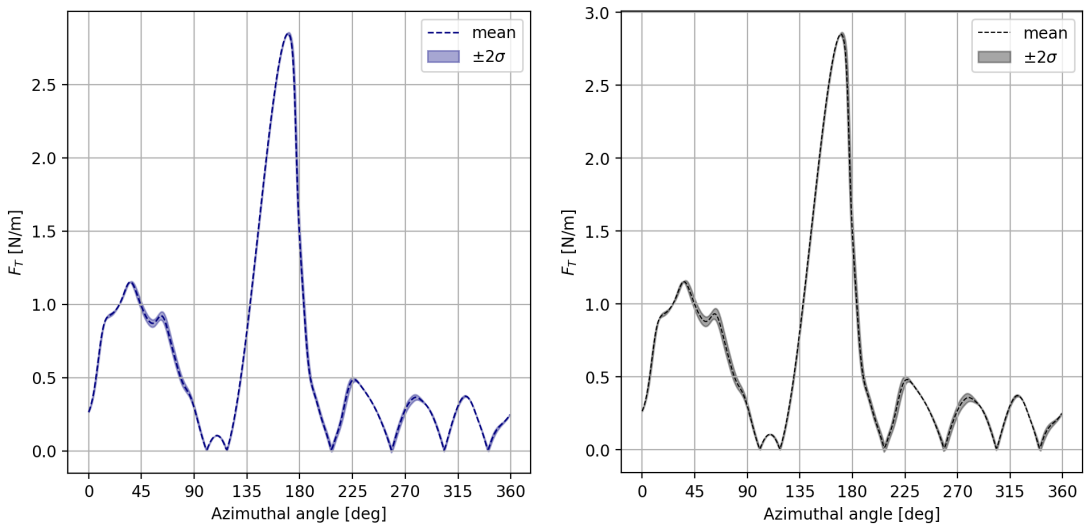
Tip-speed ratio	Use-case
0.9	ROM
0.95	ROM
1.0	ROM
1.1	ROM
1.15	ROM
1.2	ROM
1.225	ROM
1.23953	PCE 4^{th} deg
1.24365	PCE 2^{nd} deg
1.24503	PCE 4^{th} deg
1.25	PCE $2^{nd}/4^{th}$ deg
1.25497	PCE 4^{th} deg
1.25635	PCE 2^{nd} deg
1.26047	PCE 4^{th} deg
1.32	ROM

Table 1: Input parameter locations for the HF utilized for the ROMs and the PCE.

Figure 6 shows the differences between a 2^{th} and 4^{th} degree PCE for the thrust force. The mean estimate error is $E = 1.2\%$, while the second order moment tends to be underestimated by the 2^{nd} degree PCE. Equation 17 shows how the error metric is evaluated

$$E = \frac{\|\mathbb{E}[\mathbf{u}_{pred}] - \mathbb{E}[\mathbf{u}_{true}]\|_2}{\|\mathbb{E}[\mathbf{u}_{true}]\|_2} \tag{17}$$

where $\|\cdot\|_2$ is the l^2-norm, $\mathbb{E}[\mathbf{u}_{true}]$ is the ground truth mean estimate, i.e. the 4^{th} degree PCE, and $\mathbb{E}[\mathbf{u}_{pred}]$ is the estimate obtained with the compared model.



(a) 2^{nd} degree PCE.

(b) 4^{th} degree PCE.

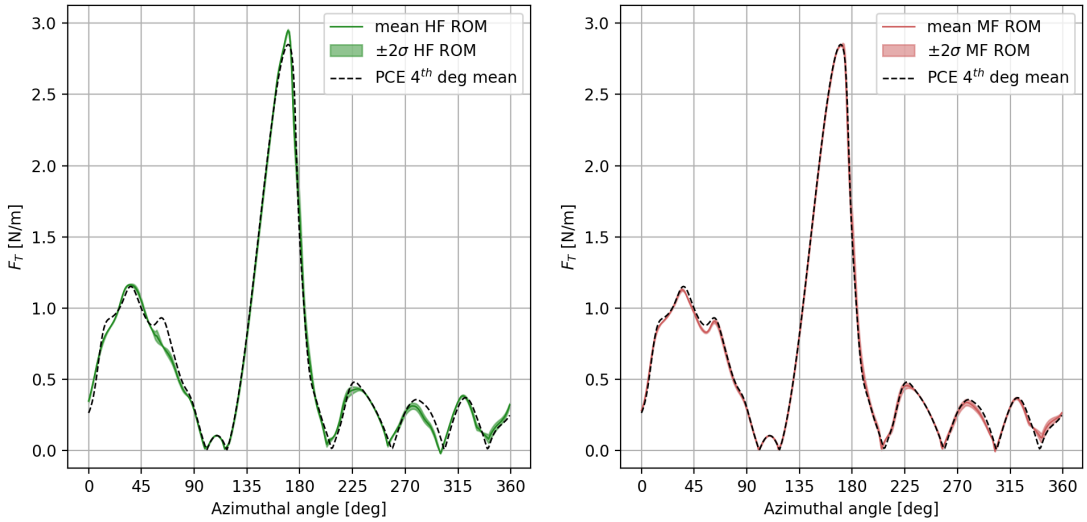
Figure 6: First and second order moment estimates of F_T with PCE.

A single-fidelity HF ROM and a MF ROM can be used to perform a MC simulation and estimate the statistical moments for F_T. According to Section 2.1.1, the HF ROM was trained using GP regression models to approximate the POD coefficients. A total of 8 HF snapshots were used, as in Table 1. The same 8 HF snapshots were considered for the MF model together with 23 LF snapshots, evenly distributed in $\lambda \in [0.9, 1.32]$, with an approximate computational cost ratio of $5 : 1$ when compared to HF simulations. Figure 7 shows the estimates of 3000 thousands MC simulations obtained with the HF ROM and the MF ROM.

The MF ROM mean estimate is more accurate than the HF ROM one, suggesting that the addition of LF snapshots is beneficial to the ROM training. Table 2 shows the two ROMs mean estimates errors with respect to the 4^{th} degree PCE, with the MF ROM having a better accuracy than the HF ROM.

	HF ROM	MF ROM
E	6.8%	2.9%
wall time [s]	0.0255	21.6

Table 2: ROMs MC simulation mean estimates error compared to ground truth, i.e. 4^{th} degree PCE, and wall time with a Intel Core i7-1255U CPU.



(a) Single-fidelity HF ROM.

(b) MF ROM.

Figure 7: First and second order moment MC estimates of F_T ROMs compared to 4^{th} degree PCE.

The errors on the mean estimates presented in Table 2 are aligned with the reconstruction error of both ROMs. The ROM reconstruction error E_r measures the ROM accuracy when approximating an unseen HF snapshot, and is computed as

$$E_r = \frac{\|\mathbf{u}^{ROM} - \mathbf{u}^{HF}\|_2}{\|\mathbf{u}^{HF}\|_2} \tag{18}$$

where $\mathbf{u}^{ROM}$ is the ROM approximation of the HF snapshot $\mathbf{u}^{HF}$. If the HF ROM and MF ROM are used to predict the F_T in the 7 λ locations associated with the 2^{nd} and 4^{th} degree PCE snapshots of Table 1, the mean reconstruction error for the two models are respectively of 6.7% and 2.8%. If the corresponding 7 LF snapshots were used as they are instead of the ROM approximations in Equation 18, the discrepancy between the LF and the HF snapshots would be 19.7%. This highlights that LF information is inaccurate and it is not suitable for studying the HF problem by itself, but it can significantly improve the MF ROM performances.

Figure 8 shows that the ROMs are able to identify the locations with high variances. However, both ROMs underestimate the PCE second order moment estimates

According to Equation 16, it is possible to integrate the F_T realizations obtained with the HF ROM and the MF ROM MC simulations, each corresponding to a different ω, and find the turbine power coefficient C_P. Similarly, the F_T PCE mean estimate can be used as well to compute the C_P for a ω corresponding to $\lambda = 1.25$.

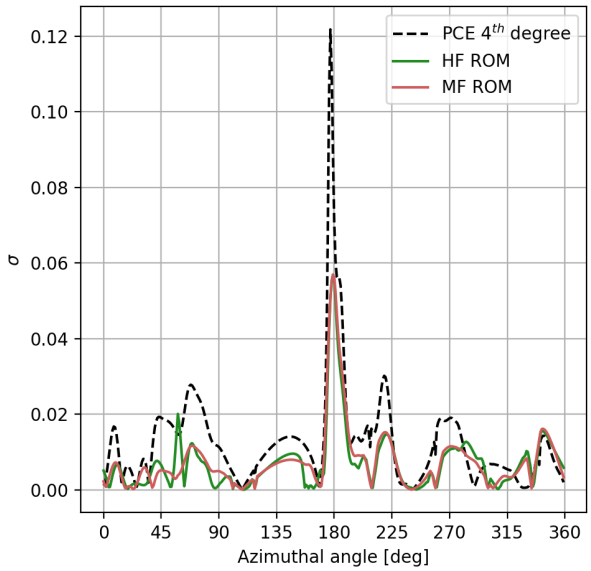


Figure 8: Comparison between the standard deviation σ associated to the second order moment estimates of the HF ROM, the MF ROM and the PCE.

Figure 9 shows that the mean C_P obtained with the integration of MF ROM F_T predictions is very close to the one obtained with the PCE mean estimation, with a relative error of 0.51%. Again, the single-fidelity ROM is less performant, showing a 1.43% relative error when estimating the mean C_P under uncertain rotational velocity.

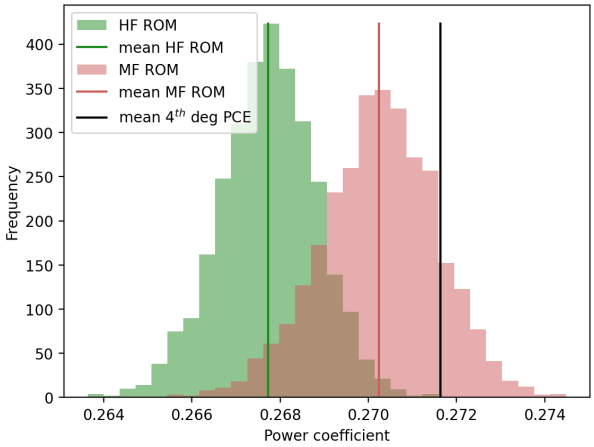


Figure 9: Comparison between the power coefficient C_P computed from HF ROM MC realizations, the MF ROM ones, and the PCE mean estimate.

4 CONCLUSIONS

This work compares a MC method applied together with a ROM against a classical PCE for the UQ of the aerodynamic forces acting on a Darrieus VAWT, under uncertain rotational velocity. In particular, a single-fidelity HF ROM and a MF ROM is proposed. Results show that the MF ROM in combination with the MC method performs similarly to a 4^{th} degree PCE, especially for the mean estimates. Moreover, the HF snapshots for the ROMs are sampled in a much wider input parameter domain than those for the PCE, resulting in an overall poor HF information density. On the other hand, a HF ROM trained on the same few HF snapshots shows higher errors than the MF ROM. This indicates that the addition of cheaper and otherwise useless LF information can be beneficial to the MF ROM. Additionally, the two ROMs reconstruction errors are in line with their MC mean error, suggesting that the ROM reconstruction error could have some potential as an empirical error estimator for UQ. Both HF ROM and MF ROM, however, underestimate the second order statistical moment. Finally, the MF ROM higher accuracy is again confirmed when considering the power coefficient mean estimates.

Future work will focus on extending the approach to a more complex parameterization to further understand MF ROM capabilities for UQ.

REFERENCES

[1] Cinnella, P., Congedo, P., Parussini, L. & Pediroda, V. Quantification of thermodynamic uncertainties in real gas flows. *International Journal Of Engineering Systems Modelling And Simulation*. **2**, 12-24 (2010) doi: 10.1504/IJESMS.2010.031867 .

[2] Santiago, A., Alonso, J., Palacios, F., Barone, M. & Eldred, M. Multi-Fidelity Uncertainty Quantification: Application to a Vertical Axis Wind Turbine Under an Extreme Gust. *AIAA AVIATION 2014 -15th AIAA/ISSMO Multidisciplinary Analysis And Optimization Conference*. (2014,6) doi: 10.2514/6.2014-3013 .

[3] Bortolotti, P., Canet, H., Bottasso, C. & Loganathan, J. Performance of non-intrusive uncertainty quantification in the aeroservoelastic simulation of wind turbines. *Wind Energy Science*. **4**, 397-406 (2019) doi: 10.5194/wes-4-397-2019 .

[4] Tabib, M., Tsiolakis, V., Pawar, S., Ahmed, S., Rasheed, A., Kvamsdal, T. & San, O. Hybrid deep-learning POD-based parametric reduced order model for flow around wind-turbine blade. *Journal Of Physics: Conference Series*. **2362**, 012039 (2022,11) doi: 10.1088/1742-6596/2362/1/012039 .

[5] Luo, Z., Wang, L., Xu, J., Wang, Z., Yuan, J. & Tan, A. A reduced order modeling-based machine learning approach for wind turbine wake flow estimation from sparse sensor measurements. *Energy*. **294** pp. 130772 (2024) doi: 10.1016/j.energy.2024.130772 .

[6] Dicech, F., Gkaragkounis, K., Parussini, L., Spagnolo, A. & Telib, H. Integration of multi-fidelity methods in parametrized non-intrusive reduced order models for industrial applications. *Journal Of Computational Science*. **85** pp. 102511 (2025) doi: 10.1016/j.jocs.2024.102511 .

[7] Mihet-Popa, L., Blaabjerg, F. & Boldea, I. Wind Turbine Generator Modeling and Simulation Where Rotational Speed is the Controlled Variable. *Industry Applications, IEEE Transactions On*. **40** pp. 3 - 10 (2004,2) doi: 10.1109/TIA.2003.821810 .

[8] Sadman Sakib, M., Todd Griffith, D., Hossain, S., Bayat, S. & Allison, J. Intracycle RPM control for vertical axis wind turbines. *Wind Energy*. **27**, 202-224 (2024) doi: 10.1002/we.2885 .

[9] Burnham, D., Santoso, S. & Muljadi, E. Variable rotor-resistance control of wind turbine generators. *2009 IEEE Power & Energy Society General Meeting*. pp. 1-6 (2009) doi: 10.1109/PES.2009.5275637 .

[10] Yamada, T., Kiwata, T., Kita, T., Hirai, M., Komatsu, N. & Kono, T. Overspeed Control of a Variable-Pitch Vertical-Axis Wind Turbine by Means of Tail Vanes. *Journal Of Environment And Engineering*. **7**, 39-52 (2012) doi: doi.org/10.1299/jee.7.39 .

[11] Pinar Pérez, J., García Márquez, F., Tobias, A. & Papaelias, M. Wind turbine reliability analysis. *Renewable And Sustainable Energy Reviews*. **23** pp. 463-472 (2013) doi: 10.1016/j.rser.2013.03.018 .

[12] Hesthaven, J. & Ubbiali, S. Non-intrusive reduced order modeling of nonlinear problems using neural networks. *Journal Of Computational Physics*. **363** pp. 55-78 (2018), doi: 10.1016/j.jcp.2018.02.037 .

[13] Scardigli, A., Arpa, R., Chiarini, A. & Telib, H. Enabling of Large Scale Aerodynamic Shape Optimization Through POD-Based Reduced-Order Modeling and Free Form Deformation. *Computational Methods In Applied Sciences*. **48** pp. 49 - 63 (2019) doi: 10.1007/978-3-319-89988-6_4 .

[14] Benner, P., Gugercin, S. & Willcox, K. A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems. *SIAM Review*, **57** pp. 483-531 (2015) doi: 10.1137/130932715 .

[15] Sirovich, L. Turbulence and the dynamics of coherent structures. Part I, II and III. *Quarterly of Applied Mathematics* **45** pp. 561-571 (1987) doi: 10.1090/qam/910462 .

[16] Kennedy, M. & O'Hagan, A. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*. **87**, 1-13 (2000) doi: 10.1093/biomet/87.1.1 .

[17] Le Gratiet, L. & Garnier, J. Recursive co-kriging model for design of computer experiments with multiple levels of fidelity. *Int J Uncertain Quantif*. **4**, 365-386 (2014) doi: 10.1615/int.j.uncertaintyquantification.2014006914 .

[18] Perdikaris, P., Raissi, M., Damianou, A., Lawrence, N. & Karniadakis, G. Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling. *Proceedings Of The Royal Society A: Mathematical, Physical And Engineering Sciences*. **473** (2017) doi: 10.1098/rspa.2016.0751 .

[19] Xiu, D. Numerical Methods for Stochastic Computations: A Spectral Method Approach. *Princeton University Press*. (2010) doi: 10.2307/j.ctv7h0skv .

[20] Zhang, J. Modern Monte Carlo methods for efficient uncertainty quantification and propagation: A survey. *WIREs Computational Statistics*. **13**, e1539 (2021) doi: 10.1002/wics.1539 .

[21] McLaren, K. A numerical and experimental study of unsteady loading of high solidity vertical axis wind turbines. (McMaster University,2011), url: http://hdl.handle.net/11375/11096 .

[22] Balduzzi, F., Bianchini, A., Maleci, R., Ferrara, G. & Ferrari, L. Critical issues in the CFD simulation of Darrieus wind turbines. *Renewable Energy*. **85** pp. 419-435 (2016) doi: 10.1016/j.renene.2015.06.048 .

[23] Liao, Y., Wang, L. & Yan, Y. Instantaneous Rotational Speed Measurement of Wind Turbine Blades using a Marker-Tracking Method. *2022 IEEE International Instrumentation And Measurement Technology Conference (I2MTC)*. pp. 1-5 (2022) doi: 10.1109/I2MTC48687.2022.9806658 .

[24] Wang, Y., Wang, L. & Yan, Y. Rotational speed measurement through digital imaging and image processing. *2017 IEEE International Instrumentation And Measurement Technology Conference (I2MTC)*. pp. 1-6 (2017) doi: 10.1109/I2MTC.2017.7969697 .

[25] Rapos, D., Mechefske, C. & Timusk, M. Dynamic sensor calibration: A comparative study of a Hall effect sensor and an incremental encoder for measuring shaft rotational position. *2016 IEEE International Conference On Prognostics And Health Management (ICPHM)*. pp. 1-5 (2016) doi: 10.1109/ICPHM.2016.7542858 .

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

A HIERARCHICAL BAYESIAN APPROACH FOR POPULATION-BASED SHM IN SHIP HULL STRUCTURES

Georgios Aravanis[1], Nicholas Silionis[2], Jacopo Bardiani[1], Marco Giglio[1], Konstantinos Anyfantis[2], and Claudio Sbarufatti[1]

[1]Politecnico di Milano, Department of Mechanical Engineering
Via La Masa 1, 20156 Milano, Italy
e-mail: {georgios.aravanis, jacopo.bardiani, marco.giglio, claudio.sbarufatti}@polimi.it

[2] National Technical University of Athens, School of Naval Architecture and Marine Engineering
9 Heroon Polytechniou Av., Athens, 15780 Zografos, Greece
e-mail: {nsil,kanyf}@naval.ntua.gr

Abstract. *Structural health monitoring (SHM) strategies involve the processing of structural response data to indirectly assess an asset's condition. These strategies can be enhanced for a group of structures, especially when they are similar, since mutual underlying physics are expected to exist. The concept behind population-based SHM exploits the sharing of data among individuals, so that data-rich members can support data-scarce ones. One approach to population-level modeling is the hierarchical Bayesian method, where the model is structured hierarchically in terms of its parameters, and correlation among learning tasks is enabled by conditioning on shared latent variables.*

This work investigates the application of a hierarchical Bayesian model to infer expected distributions of deflection amplitudes at both the population and domain levels, with the aim of detecting excessive initial deflections in a population of plate elements. Although these damages are typically localized, they can trigger unexpected events, if not properly monitored. The work is conducted in a numerical setting using a Finite Element model to generate strain response data, which serve as the monitoring data. Bayesian inference was conducted using Markov Chain Monte Carlo (MCMC), with a surrogate model employed to calculate the likelihood function. The hierarchical approach was compared to an independent model for a plate component with few data. The results revealed that, under data sparsity conditions, the hierarchical model can offer more robust results in terms of uncertainty, which is essential for decision-making tasks.

Keywords: Hierarchical Bayes, Population-based SHM, Uncertainty quantification, Markov chain Monte Carlo, Damage detection.

1 INTRODUCTION

Oftentimes, failure of many engineering assets is associated with localized damage events. In the case of ship hull structures, which are primarily composed of plates and stiffened panels, deterioration of these components can lead to reduced load-carrying capacity and potentially unexpected failure modes. Thus, the deployment of Structural Health Monitoring (SHM) campaigns on-board [1, 2], which are able to monitor such events, becomes highly relevant for helping operators make timely maintenance decisions.

In SHM, the challenge essentially lies in learning Quantities of Interest (QoIs) that characterize the underlying deterioration process. These are most often unobservable, and thus structural response quantities (e.g., strain or acceleration sensor data) are instead used to determine their values. Concurrently, this inference procedure needs to be robust in terms of uncertainty, as the confidence level in the deployed SHM system for decision-making directly depends on how well uncertainty is handled. This problem formulation lends itself well to a Bayesian framework [3], where the goal is to estimate the updated joint probability distribution of the QoIs conditioned on the acquired sensor data. Acknowledging this, numerous studies in the SHM field have employed Bayesian methods to address different tasks in the classical SHM hierarchy [4]; namely, related to (damage) diagnosis [5], identification [6], and prognosis [7, 8].

Despite the benefits of developing SHM approaches for structural integrity management purposes, their effectiveness is bounded by the availability of data relevant to the problem at hand. In real-world large-scale structures, this poses a critical challenge [9], since increased resource restrictions arising from the scale of the system influence data availability. At the same time, their inherent complexity requires sufficiently comprehensive datasets to describe the various phenomena, which one seeks to model. On the other hand, such structures often exhibit fundamental similarities in terms of their constituent components, presenting an opportunity to enable information sharing among them. If we know that they have been produced under similar specifications and with the same material, they can be considered nominally identical [10]. Even then however, they are not truly identical due to inherently random events (i.e., relating to manufacturing, assembly, materials, etc.). An example of this is the repeated presence of plate elements in a ship hull. Although produced on the same manufacturing line, they exhibit geometric variability due to fabrication-related factors during the ship's construction, which often manifests as deviations in their intended shape.

The variation in the structural attributes of individual components results in variations in their response, which complicates the development of a model that generalizes well across all of them. Hierarchical Bayesian modeling provides a principled statistical framework to model these variations within a group while leveraging statistical similarities among its members [11, 12], all while providing the key benefits of Bayesian modeling. The statistical correlations enabled through hierarchical models allow for improved inference in sparse data scenarios, whether due to data loss (e.g. transmission issues, sensor failure, etc.) or due to the challenges of online monitoring, where early predictions must be made with limited initial observations. An additional benefit of this approach is that it makes the most out of well-instrumented domains, and thus the overall added value of SHM implementation is increased compared to a more standard approach.

Hierarchical Bayesian models have seen particular success in several works related to population based SHM (PBSHM) [13, 14, 15], as well as other engineering problems framed in a hierarchical manner [16, 17]. This is because of their ability to capture both common trends across a population as well as individual differences among its members, which aligns naturally

with the concept of constructing a general population model. However, despite their benefits, these models have yet to be explored in SHM applications for marine structures.

In this work, we demonstrate a hierarchical Bayesian framework on a large-scale double-bottom model of a containership, where damage is defined as a set of bathtub-shaped plate deflections. The approach involves first generating realistic strain observations by Finite Element (FE) modeling. These observations are subsequently incorporated into a hierarchical model, alongside an FE-based surrogate, necessary for the calculation of the likelihood function and, ultimately, the prediction of parameters associated with the deflection amplitudes. Results are presented in terms of variance in the parameter estimates to assess whether data-poor plates can benefit from statistical information at the population level, to enable robust damage detection under data-sparsity.

2 THE DATASET

The dataset used in this work was generated from a detailed FE model, which corresponds to part of the double-bottom structure within the central hold of a containership, and is of size 27 m $\times$ 18 m $\times$ 2.2 m. A detailed description of the structural information has been omitted here for the sake of brevity but can be found in [18]. This section of the ship hull was selected because it is typically under constant in-plane compressive loading due to the nature of the operational loads a containership experiences. The combination of compression and initial out-of-plane deflections for plate elements can accelerate the onset of buckling, which makes this region particularly critical.

2.1 Finite element modeling particulars

The FE model features 4-node linear shell elements for the external bottom and 2-node linear beam elements to represent the various stiffening members of the double-bottom (i.e., girders, floors, longitudinal stiffeners). The stiffness contributed by the inner bottom has been implicitly captured by incorporating an effective width for the girders and floors. The group of $K = 6$ plates enclosed within the red boundaries in Figure 1(a) defines the region used for strain data generation for the purposes of this study. Each plate, of size $a \times b$, deforms based on a sinusoidal expression given by:

$$w(u, v; \tilde{w}) = \tilde{w} \sin\left(\frac{\pi u}{a}\right) \sin\left(\frac{\pi v}{b}\right) \tag{1}$$

where $\tilde{w}$ denotes the amplitude of the deflection, $u \in [0, a]$ and $v \in [0, b]$. Their adjacent plates were also modeled as deflected to provide realistic states for the surroundings of the monitored region, since all structural elements are expected to exhibit slight imperfections within allowable limits. They are shown with a slight black shading to indicate that they are not part of the data generation process. During the modeling process, $\tilde{w}$ was controlled parametrically to generate different realizations of out-of-plane deflections. The perspective shape of a simulated out-of-plane deflection can be seen in Figure 1(b), where the (conceptual) strain sensor location is also indicated by a red marker. This placement is consistent across all K plates.

The loads considered are those typically experienced by a containership while sailing in calm water conditions; namely, hogging (i.e., negative) hull girder bending, lateral pressure on the side shell and outer bottom, and the cargo weight [19]. At the double-bottom level, these translate into longitudinal (N_x) and transverse (N_y) in-plane compressive stresses, pressure (P) distributed over the hull exterior, and line loads (Q) on the inner bottom, respectively. These were assigned with representative values based on the loading manual of the vessel. In terms

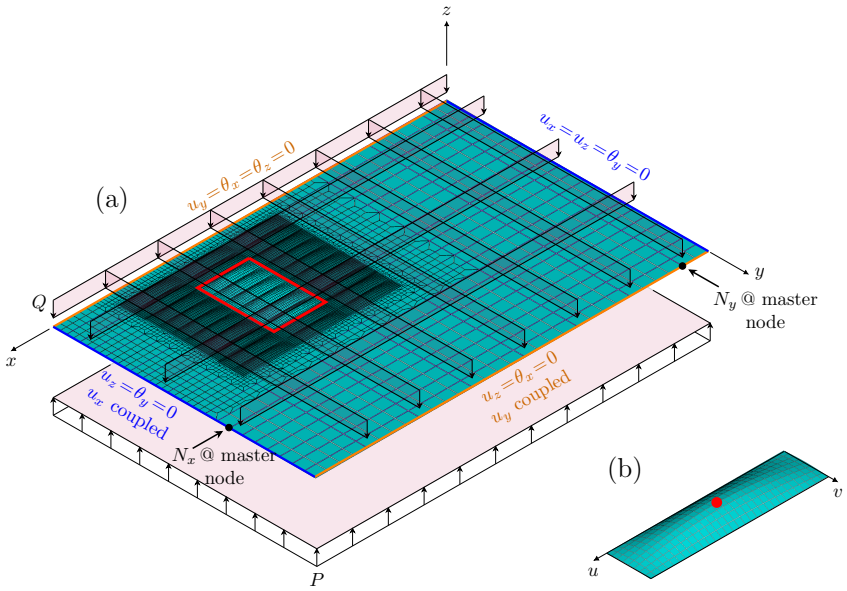


Figure 1: (a) FE model of the subject geometry including the monitored plates (within red boundaries), and (b) representative out-of-plane deflection with strain sensor at the center. Adapted from the original work of the authors [18].

of the boundary conditions, they were defined based on how the considered geometry deforms with respect to the entire structure and are detailed in the descriptions adjacent to the edges of the model.

Regarding the material, linear elastic and isotropic properties (Young's modulus $E = 207$ GPa, Poisson's ratio $v = 0.3$) were assigned. A static, linear elastic analysis was ultimately employed to solve the model, essentially representing conditions when the vessel is in port or sailing in still water (i.e., no waves). Nevertheless, the data sampling rate implied in this context is considered sufficient, given the relatively low temporal evolution of out-of-plane deflections.

2.2 Dataset generation

In the data generation process, transverse strains ε_{yy} were selected as the observed quantities, which falls in line with the underlying physics of the problem, as plate bending theory dictates that strains in the shorter dimension are the dominant ones. Hereinafter, these are denoted as ε for notational simplicity.

To generate the latent quantities $\tilde{w}$ that give rise to these observations, it is assumed that each plate $k \in K$ is deflected based on a local (Normal) distribution for $\tilde{w}$, whose parameters are themselves realizations from common higher-level distributions. To this end, global distributions were placed over both the expected value of the deflection amplitude and its variance. The former captures the variability introduced by fabrication processes, leading to differences in expected deflection amplitudes among plates. Meanwhile, the latter accounts for changing conditions that introduce additional variability around each expected deflection, reflecting uncertainties in the structural response beyond systematic fabrication-induced deviations.

By sampling from these global distributions, K pairs of local expectations and variances were obtained, defining an equal number of local distributions, from each of which N_k samples of $\tilde{w}$ were drawn. The resulting matrix of K-dimensional vectors $\{[\tilde{w}_{i,1}, \ldots, \tilde{w}_{i,k}, \ldots, \tilde{w}_{i,K}]\}_{i=1}^{N_k}$

$\in \mathbb{R}^{N_k \times K}$ was concatenated with additional samples corresponding to the deflections of the neighboring plates, which were assumed to remain within allowable tolerances. The augmented vectors were fed into the constructed FE model to generate transverse strain observations ε, which were subsequently perturbed with additive white Gaussian noise with a standard deviation of 5 $\mu\varepsilon$. The corresponding pairs of $\{\tilde{w}_{i,k}, \varepsilon_{i,k}\}_{i=1}^{N_k}$ for each $k \in K$ are shown with different colors in Figure 2, where $N_k = 20$ for $k \in \{1, \ldots, 5\}$ and $N_k = 2$ for $k = 6$. In this way, some measurements are hidden from the employed models, to demonstrate the robustness of hierarchical modeling in imbalanced dataset settings.

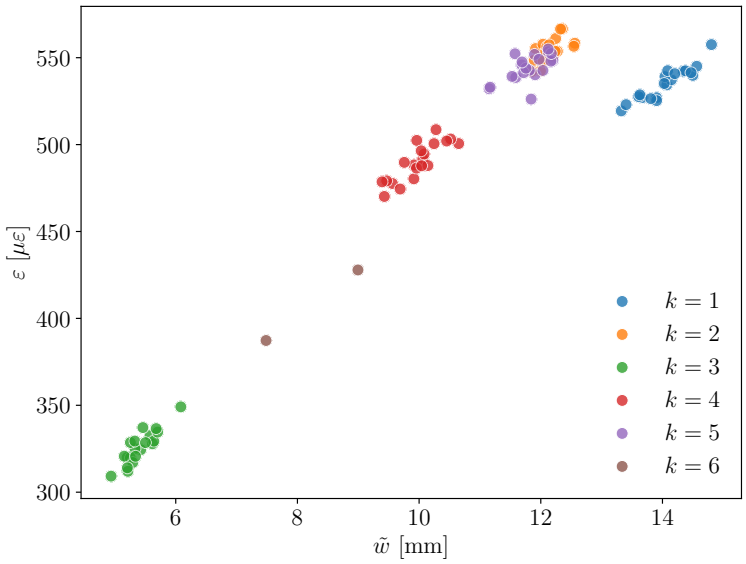


Figure 2: Clusters of generated observations of transverse strain for the six plate components. The brown markers show Plate 6 with scarce data.

3 THE HIERARCHICAL BAYESIAN APPROACH

To establish the hierarchical Bayesian formulation, we represent the data from the entire group of plates as:

$$\{\tilde{\mathbf{w}}_k, \boldsymbol{\varepsilon}_k\}_{k=1}^{K} = \left\{ \{\tilde{w}_{i,k}, \varepsilon_{i,k}\}_{i=1}^{N_k} \right\}_{k=1}^{K} \tag{2}$$

The objective is to learn the posterior distributions of the parameters (local expectations and variances) and hyperparameters (higher-level expectations and variances) that govern the deflection amplitude $\tilde{w}$, given the observed strain data ε.

3.1 Surrogate modeling

Each plate (domain) is related to its own task, which can be formally expressed as:

$$\varepsilon_{i,k} = \mathcal{M}(\tilde{w}_{i,k}) + \epsilon_{i,k} \tag{3}$$

where $\mathcal{M}(\cdot)$ denotes the model operator and $\epsilon_{i,k}$ is the additive noise term. The former represents the expected mapping from deflections to transverse strain, established by the constructed FE model, which was eventually replaced with a cheaper data-driven surrogate. This is typical practice to avoid computational overhead when employing sampling-based algorithms (i.e.,

Markov Chain Monte Carlo (MCMC)) for the estimation of posterior distributions in a Bayesian context.

The surrogate model was chosen to be a standard Gaussian Process Regression (GPR) model [20] because it guarantees against overfitting and automatically returns confidence intervals associated with its predictive capacity. The former property is quite relevant in our case for robustly capturing the general pattern of the dataset, as the presence of multiple out-of-plane deflections contributes to some level of variability in the data. The predictive variance provided by the GPR was also analyzed to assess the relative spread of strain values for different deflection amplitudes (i.e., the coefficient of variation), which was found to be $\sim 7\%$ at the mean deflection amplitude. Given this level of variance, while acknowledging the computational complexity and inherent high-dimensionality that comes with hierarchical models, it was decided to only take the GPR mean into consideration in the likelihood function evaluation. Figure 3 illustrates a trained GPR model for a specific plate $k \in K$. Independent surrogate models were trained for each task to prevent bias in the mean prediction, and thus the model operator $\mathcal{M}(\cdot)$ in Equation 3 is more pertinently denoted as $\mathcal{M}_k(\cdot)$ for clarity.

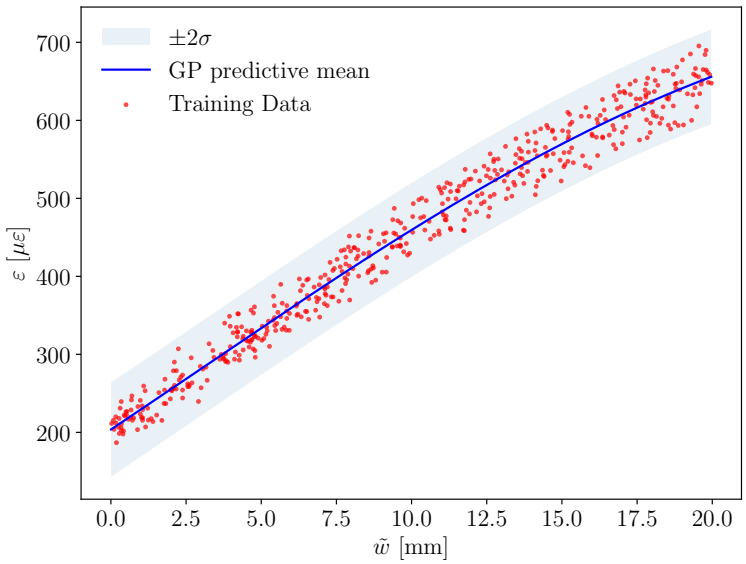


Figure 3: GPR surrogate model for Plate 6. Red markers represent the training-set (FE test data), the blue line shows the posterior predictive mean, and the shaded region indicates a 2σ credible area. Note that for the construction of the likelihood function we use only the mean of the GPR.

3.2 The hierarchical model

This work compares a partially-pooled hierarchical model, which assumes correlation between plate-specific parameters, with an independent (no-pooling) model that does not, to motivate sharing information within the group of plates. Pooling refers to the way information is shared across members in hierarchical models; a no-pooling approach predicts separate parameters for each member, while conversely, a complete pooling approach assumes all members share the same parameters. Partial pooling strikes a balance by allowing some parameters to vary across members and correlation is enabled among them by conditioning on shared latent variables [3]. Since the deflection amplitudes related to each plate are assumed to be sampled from distributions with correlated parameters for the partially-pooled hierarchical model, one

expects that the data-rich members ($k = 1, \ldots, 5$) should support the data-scarce one ($k = 6$).

The hierarchical Bayesian model structure employed in this work is now presented. This follows from the work of Brealy et al. [15]. The likelihood function for the model was constructed using the commonly employed zero-mean Gaussian measurement error model,

$$\left\{ \{\varepsilon_{i,k}\}_{i=1}^{N_k} \right\}_{k=1}^{K} \sim \mathcal{N}(\mathcal{M}_k(\tilde{w}_{i,k}), \gamma^2) \tag{4}$$

where γ^2 is the variance of the measurement error term ϵ. This was considered to be the same for each plate under the assumption that the same hardware were used for data acquisition.

To start the Bayesian formulation, one can set prior distributions over the deflection amplitude realizations $\tilde{w}_{i,k}$ related to each plate:

$$\left\{ \{\tilde{w}_{i,k}\}_{i=1}^{N_k} \right\}_{k=1}^{K} \sim \mathcal{N}(\mu_{\tilde{w}_k}, \sigma^2_{\tilde{w}_k}) \tag{5}$$

The expectation $\mu_{\tilde{w}_k}$ and variance $\sigma^2_{\tilde{w}_k}$ associated with each plate are themselves sampled from global Normal distributions,

$$\{\mu_{\tilde{w}_k}\}_{k=1}^{K} \sim \mathcal{N}(\mu_\mu, \sigma^2_\mu) \tag{6}$$

$$\{\sigma_{\tilde{w}_k}\}_{k=1}^{K} \sim \mathcal{N}(\mu_\sigma, \sigma^2_\sigma) \tag{7}$$

where μ_μ, σ^2_μ, μ_σ and σ^2_σ are the higher-level parameters, which are shared among plates. One should also encode prior knowledge of the learning tasks as prior distributions over these higher-level parameters, referred to as hyperpriors. All four were assigned Gamma hyperpriors, more specifically:

$$\mu_\mu \sim \mathcal{G}(\text{shape} = 3, \text{rate} = 0.2) \tag{8}$$

$$\sigma_\mu \sim \mathcal{G}(\text{shape} = 0.8, \text{rate} = 0.35) \tag{9}$$

$$\mu_\sigma \sim \mathcal{G}(\text{shape} = 3.6, \text{rate} = 6) \tag{10}$$

$$\sigma_\sigma \sim \mathcal{G}(\text{shape} = 4.8, \text{rate} = 16) \tag{11}$$

The Gamma distribution was chosen to ensure positivity, which is essential for the physical parameters involved in the model. The constants in Equations (8)-(11) (given in mm) were specified using the formulas for the two basic statistical moments (i.e., expected value and variance) of a Gamma-distributed random variable θ. These are given by [3],

$$\text{E}[\theta] = \frac{\text{shape}}{\text{rate}} \tag{12}$$

$$\text{Var}[\theta] = \frac{\text{shape}}{\text{rate}^2} \tag{13}$$

Following this, the shape and rate parameters were chosen in a way that the resulting variance for each higher-level parameter allows for sufficient exploration across a reasonable range of values. With the same rationale, the expected value was deliberately set to deviate to a certain extent from the ground-truth values used in data generation, ensuring that the model actually learns from the data. We also ensured that the chosen hyperprior parameters led to parameter/hyperparameter samples that, ultimately, determined non-negative deflection amplitudes, despite using Normal distributions.

Similarly to the higher-level parameters, a Gamma prior distribution was set for the shared noise variance γ^2,

$$\gamma \sim \mathcal{G}(\text{shape} = 80, \text{rate} = 16) \tag{14}$$

where the chosen parameter values reflect a fairly narrow support of possible values, as practically in most cases measurement noise is known a priori. Note that hypepriors for γ are given in $\mu\varepsilon$.

Figure 4(a) shows a directed acyclic graph (DAG) of the Bayesian model under a hierarchical approach. The plate-level parameters are indexed by k, and plate notation is used to indicate that these nodes are repeated over the K plate components. Shaded and unshaded nodes represent observed and latent variables, respectively. Higher-level nodes are positioned outside the plates and are not indexed by k, denoting that they are shared across all members. Figure 4(b) illustrates the DAG for the independent approach. The independent models can be derived by removing the K plate from the hierarchical model's DAG and explicitly indexing the noise variance by k. Although the plate-level parameters are still conditioned on shared higher-level nodes, in this case, these nodes are only informed by the data from one specific member.

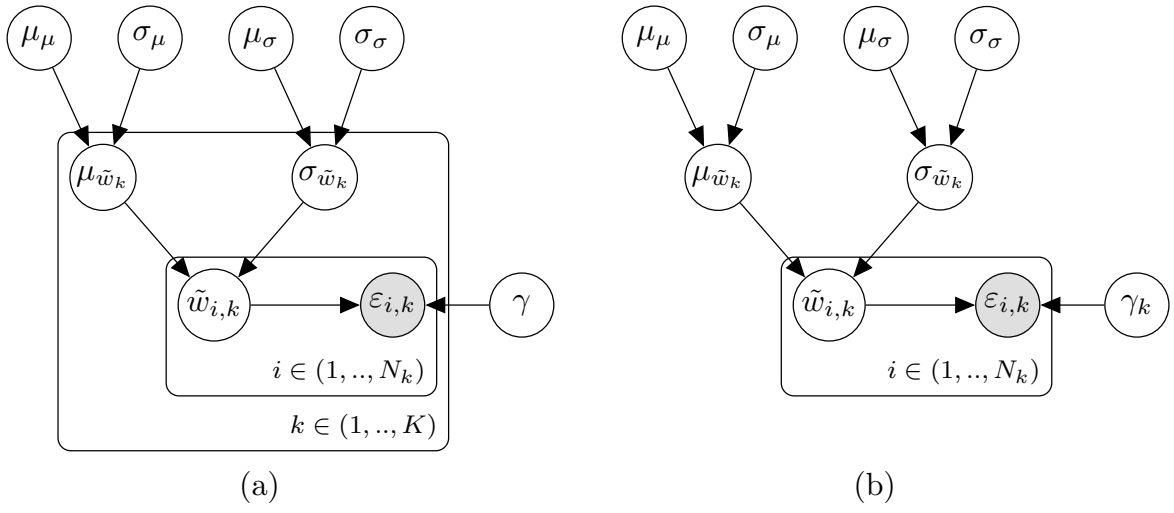


Figure 4: DAGs representing (a) the hierarchical (partial-pooling) model and (b) the independent model.

4 RESULTS

4.1 Inference

Inference of the parameters was performed using MCMC methods, via the no U-turn variant (NUTS) of Hamiltonian Monte Carlo (HMC) [21]. The implementation was carried out

in `NumPyro` [22], which uses automatic differentiation to make gradient-based samplers like NUTS feasible.

The setup comprised of 4000 samples for warmup and 2000 samples for inference, run across four parallel chains, with each chain starting from a randomly generated initial state to ensure convergence. The rank-normalized $\hat{R}$ diagnostic [23] was employed as quantitative check of convergence, and posteriors were accepted when this was lower than 1.01. This was satisfied for all parameters/hyperparameters, as can be seen by the trace plots on the right-hand side of Figure 5. The resulting kernel density function (KDE)-based posteriors for all chains (shown by each line style) are also shown on the left-hand side of the figure. The hyperpriors for the shared parameters have been plotted with magenta dashed lines, while the color scheme for the plate-specific parameters follows the convention used in Figure 2.

From the results, one notices that the modes of the posterior distributions for the shared higher-level parameters (apart from γ) have shifted away from their prior modes. Additionally, the variance of these posterior distributions is notably reduced. These observations provide assurance that the algorithm is working properly and that the model has learned from the measurements. As expected, the posterior distribution of the noise variance γ remains relatively unchanged from its prior due to the tightly constrained prior range and the inherent properties of the Gamma distribution, which features shallow tails that might affect the NUTS sampler during exploration of the posterior. However, this is compensated via μ_σ which encodes any additional variance observed in the data, and behaves as intended.

For the plate-specific parameters $\mu_{\tilde{w}_k}$ and $\sigma_{\tilde{w}_k}$, the posterior variance for the data-scarce plate (i.e., $k = 6$) is considerably larger than that of the data-rich plates. This is expected, given that it is informed by only two observations, whereas the other plates have 20 data points each. A comparison was conducted between this posterior distribution and the one obtained when modeling this plate independently (i.e., without pooling information from the broader dataset) in order to show the benefit of hierarchical models.

4.2 Damage detection

While the inferred latent variables related to deflection levels are those that describe the underlying structural behavior, they are not directly observed in an operational setting. Instead, for downstream tasks, such as damage detection, one should revert to the strain space, as this is what can be monitored and used by operators.

Along these lines, the posterior distributions of the plate-specific parameters were used for damage detection in the following way. First, posterior MCMC samples were taken for the expectation and variance for plate $k = 6$, to draw a deflection amplitude each time from a Normal distribution with the same statistics as the parameter draws (according to Equation 5). Next, to obtain posterior predictive samples for the transverse strain, the drawn deflection amplitude samples were evaluated through the mean function of $\mathcal{M}_{k=6}(\cdot)$, and the likelihood function was applied using the posterior predictive measurement noise γ^2 (see Equation 4), for each sample. KDE estimation was then employed to approximate the posterior predictive distribution of strains, using the samples from the likelihood. This process was applied for both the hierarchical (partial-pooling) and the independent (no-pooling) approaches.

The posterior predictive distributions for the transverse strain are shown in Figure 6, with the vertical lines indicating strain values associated with different deflection thresholds set by maritime regulatory bodies. Namely, the 8 mm level corresponds to a limit deflection threshold that can be seen as a damage indicator, while the 4 mm represents a standard level expected under normal operating conditions. The 6 mm level constitutes an intermediate state. The

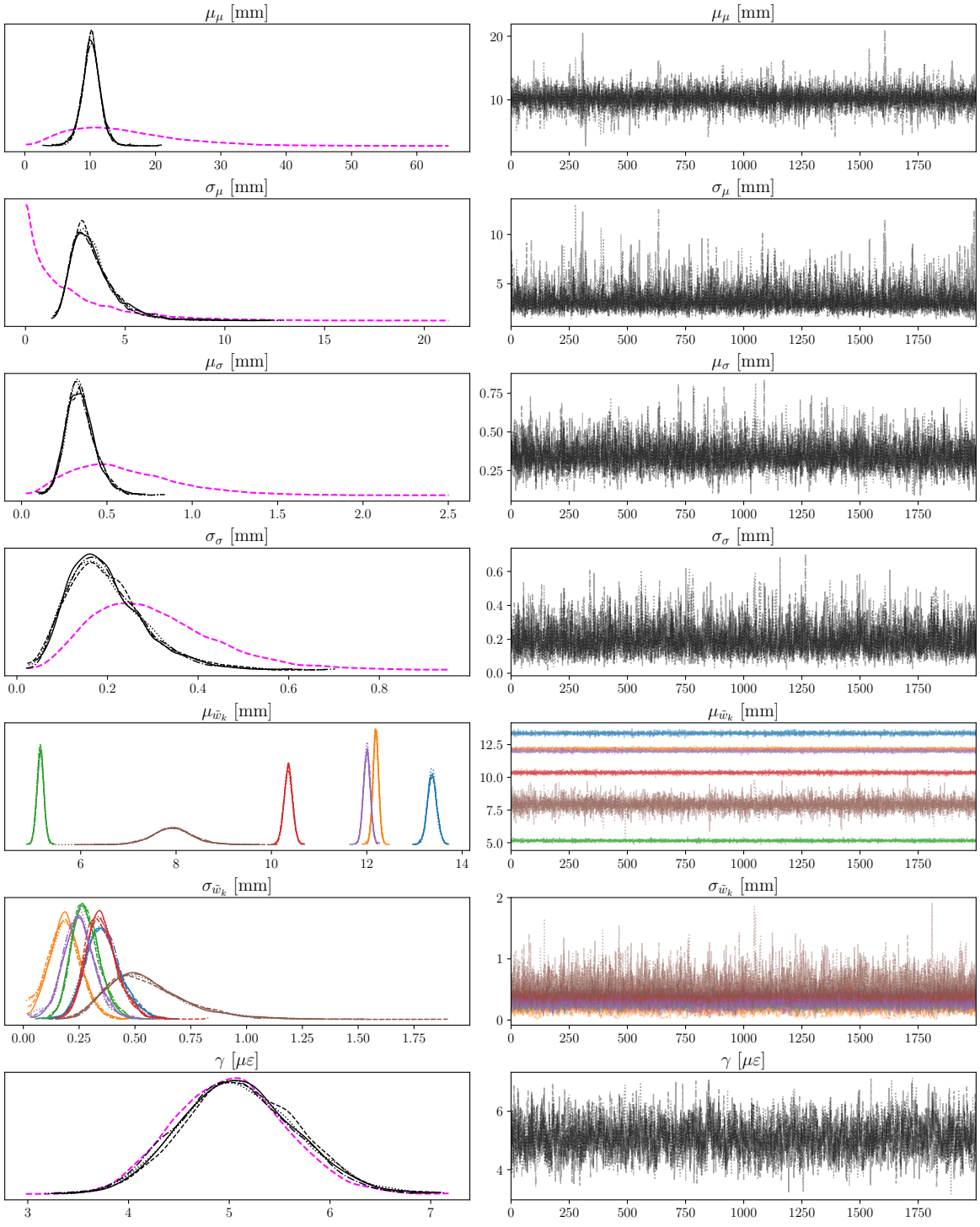


Figure 5: KDE-based posteriors of hierarchical model parameters/hyperparameters (left panel) and trace plots (right panel). Dashed magenta lines show the prior distributions applied to the shared parameters. The color scheme for $k = \{1, \ldots, 6\}$ follows: blue, orange, green, red, purple, and brown.

strain values for these thresholds were computed by passing them into the surrogate model an equal number of times as the posterior MCMC samples of γ^2, drawing observations as per the likelihood, and then averaging the results for each deflection level.

It is evident that the partial pooling model has managed a noticeable reduction in the level of

uncertainty in its predictions compared to the no-pooling model. This follows from the fact that the parameters pertaining to the latent variable have been learned within a joint inference setting. Such a reduction in uncertainty is particularly valuable in industrial applications, where high uncertainty can lead to operational challenges. For instance, if a deflection amplitude exceeding the 8 mm threshold implies inspection, large uncertainties could result in higher probability of false alarms to occur. This, in turn, would lead to unnecessary downtime and increased costs for stakeholders. At the same time, within a signal detection theory context [24], the reduction in uncertainty enhances the probability of detection by improving the model's ability to distinguish actual damage from normal variations. This leads to a more reliable assessment of structural integrity and reduces the risk of missed detections.

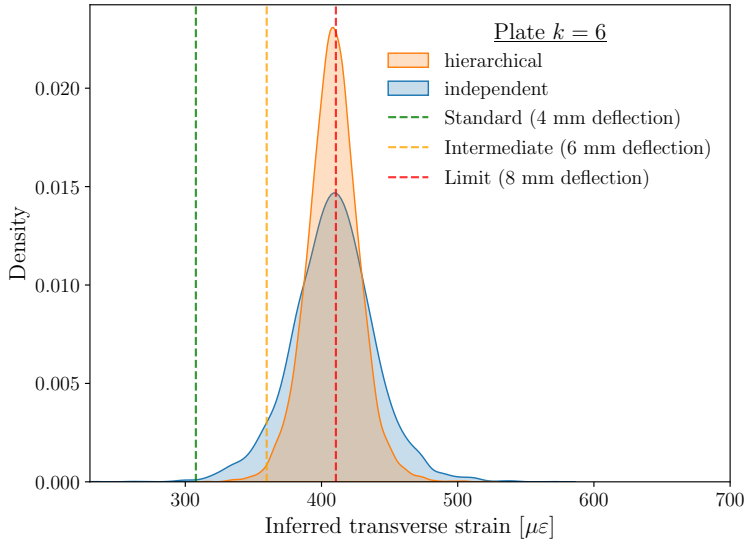


Figure 6: KDE-based posterior predictive densities of strain under a hierarchical (partial-pooling) approach and an independent (no-pooling) approach. The dashed lines represent averaged strain values for different deflection levels.

5 CONCLUSIONS

In this work, a hierarchical Bayesian model has been constructed to learn the latent (both local and global) parameters that drive a fabrication-induced deterioration phenomenon commonly encountered in ship hull structures, i.e., out-of-plane deflections. A set of deflected plates was assumed in a region of the double-bottom structure of a containership, and synthetic strain observations were generated through an FE model. These observations served as features in the inference process, where an MCMC sampler, specifically the NUTS algorithm, was employed. Given the computational demands of NUTS, a surrogate model was used in place of the FE model to improve computational efficiency.

To demonstrate the capabilities of hierarchical modeling, a specific plate was assumed with less data compared to the others. The hierarchical model enabled this data-poor plate to borrow statistical strength from the broader population, reducing prediction uncertainty. The refined posterior uncertainties of the parameters were compared against an independent no-pooling approach, with results showing that the hierarchical model enables more informed decision-making in data sparse scenarios.

REFERENCES

[1] C. R. Farrar and K. Worden, *Structural Health Monitoring: A Machine Learning Perspective.* John Wiley & Sons, 2012.

[2] C. R. Farrar and K. Worden, "An introduction to structural health monitoring", *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 365, no. 1851, pp. 303–315, 2007.

[3] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin. *Bayesian Data Analysis.* CRC press, third edition, 2013.

[4] A. Rytter, "Vibrational based inspection of civil engineering structures", Ph.D.dissertation, Aalborg University, Denmark, 1993.

[5] P. Gardner, L.A. Bull, N. Dervilis, K. Worden, "On the application of kernelised Bayesian transfer learning to population-based structural health monitoring", *Mechanical Systems and Signal Processing*, vol. 167, p. 108519, 2022.

[6] M. K. Ramancha, M. A. Vega, J. P. Conte, M. D. Todd, and Z. Hu, "Bayesian model updating with finite element vs surrogate models: Application to a miter gate structural system," *Engineering Structures*, vol. 272, p. 114901, 2022.

[7] D. Cristiani, C. Sbarufatti, F. Cadini, and M. Giglio, "Fatigue damage diagnosis and prognosis of an aeronautical structure based on surrogate modelling and particle filter," *Structural Health Monitoring*, vol. 20, no. 5, pp. 2726–2746, 2021.

[8] D. Cristiani, C. Sbarufatti, and M. Giglio, "Damage diagnosis and prognosis in composite double cantilever beam coupons by particle filtering and surrogate modelling," *Structural Health Monitoring*, vol. 20, no. 3, pp. 1030-1050, 2021.

[9] P. Cawley, "Structural health monitoring: Closing the gap between research and industrial deployment", *Structural Health Monitoring*, vol. 17, no. 5, pp. 1225-1244, 2018.

[10] L.A. Bull, P.A. Gardner, J. Gosliga, T.J. Rogers, N. Dervilis, E.J. Cross, E. Papatheou, A.E. Maguire, C. Campos, and K. Worden, "Foundations of population-based SHM, Part I: Homogeneous populations and forms", *Mechanical Systems and Signal Processing*, vol. 148, p. 107141, 2021.

[11] A. Gelman and J. Hill. *Data Analysis Using Regression and Multilevel/ Hierarchical Models*. Cambridge university press, 2006.

[12] I. G. Kreft and J. De Leeuw. *Introducing Multilevel Modeling*. Sage, 1998.

[13] L. A. Bull, D. Di Francesco, M. Dhada, O. Steinert, T. Lindgren, A. K. Parlikad, A. B. Duncan and M. Girolami, "Hierarchical Bayesian modeling for knowledge transfer across engineering fleets via multitask learning", *Computer-Aided Civil and Infrastructure Engineering*, vol. 38, no. 7, pp. 821–848, 2023.

[14] T.A. Dardeno, K. Worden, N. Dervilis, R.S. Mills and L.A. Bull, "On the hierarchical Bayesian modelling of frequency response functions", *Mechanical Systems and Signal Processing*, vol. 208, p. 111072, 2024.

[15] S. M. Brealy, A. J. Hughes, T. A. Dardeno, L. A. Bull, R. S. Mills, N. Dervilis, and K. Worden, "Multitask learning for improved scour detection: A dynamic wave tank study", *arXiv preprint arXiv:2408.16527*, 2024.

[16] N. Papadimas and T. Dodwell, "A hierarchical Bayesian approach for calibration of stochastic material models," *Data-Centric Engineering*, vol. 2, p. e20, 2021.

[17] D. Di Francesco, M. Chryssanthopoulos, M. H. Faber, and U. Bharadwaj, "Decision-theoretic inspection planning using imperfect and incomplete data," *Data-Centric Engineering*, vol. 2, p. e18, 2021.

[18] G.I. Aravanis, N.E. Silionis, K.N. Anyfantis, "Damage detection in ship hull structures under operational variability through strain sensing", *Ocean Engineering*, vol. 286, p. 115537, 2023.

[19] Y. Sumi *et al.*, *Final report of committee on large container ship safety*. Tokyo, Japan: Committee on large container ship safety, Maritime Bureau of Japan's Ministry of Land, Infrastructure, Transport and Tourism (MILT), 2015.

[20] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005

[21] M. D. Hoffman, and A. Gelman, "The No-U-Turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo", *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1593–1623, 2014.

[22] D. Phan, N pradhan and M. Jankowiak, "Composable effects for flexible and accelerated probabilistic programming in NumPyro", *arXiv preprint arXiv:1912.11554*, 2019.

[23] A. Vehtari, A. Gelman, D. Simpson, B. Carpenter, and P.-C. Bürkner, "Rank-Normalization, Folding, and Localization: An Improved $\hat{R}$ for Assessing Convergence of MCMC (with Discussion)," *Bayesian Analysis*, vol. 16, no. 2, 2021.

[24] S.M. Kay. *Fundamentals of Statistical Signal Processing Volume II: Detection Theory*. Prentice Hall, 1998.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

DOMAIN ADAPTATION IN STRUCTURAL HEALTH MONITORING USING LEARNING BOUNDS AND GENERATIVE NETWORKS

C.A. Lindley[1], N. Dervilis[1], and K. Worden[1]

[1]University of Sheffield
School of MAC, Dynamics Research Group
Mappin St, Sheffield S1 3JD
{c.a.lindley,n.dervilis,k.worden}@sheffield.ac.uk

Abstract. *The use of statistical models in engineering can be limited by the availability of data. This issue is particularly prevalent in structural health monitoring, where labelled data can be expensive to obtain since it requires data from most, if not all, operational conditions that a structure might experience. As a result, there has been a surge of interest in transfer learning within the research community, aiming to use existing data to improve the learning on new structures that have yet to experience previously-unseen conditions. The approach taken in this study is based on statistical learning theory to estimate an upper bound on the expected risk of a transfer learner. An adversarial-type neural network architecture is then designed to minimise this bound, producing a latent representation that aligns different domains and, in turn, improves learning in systems where labelled data are scarce. The outcome of this approach is a novel generative domain-adversarial neural network that enables the estimation of uncertainty in predictions. A simple simulated case study is presented here to demonstrate the effectiveness of the proposed model in addressing challenges related to unsupervised domain adaptation.*

Keywords: Domain Adaptation, Generalisation Bounds, Uncertainty Quantification, Multi-scale Analysis, Structural Health Monitoring.

1 INTRODUCTION

The incorporation of statistical methods into Structural Health Monitoring (SHM) is a subject of increasing interest in the literature. The success of these methods partly depends on the amount of labelled data, which may, unfortunately, not often be available in SHM. This shortcoming is because structures may exhibit unprecedented operational conditions that are not covered by the existing data. In theory, if one could collect data for every possible health state a structure may experience during its lifespan, the cost would be unfeasibly high, since this scenario would mean building many copies of the structure to capture all conditions. It is thus important to account for this limitation when employing statistical methods in SHM.

Nevertheless, progress in the field has shown that machine learning and statistical methods can be successfully used for damage detection, localisation, classification and prognosis [1]. These advances offer the prospect of a framework that could work reliably in practice. One promising approach pertains to a branch of machine learning known as *transfer learning* [2]. The main idea is simple; that is, to improve the performance of an intelligent system by using existing data from external sources. In other words, by including health states already observed in other systems, it may be possibly to predict the current health state of the system in question, even when facing a condition that has not been seen before.

To illustrate the motivation for this paper, the following framework is constructed. Let a structure S be modelled by a mathematical function M which attempts to predict the behaviour of S. Furthermore, the model may be expected to account for certain aspects of the behaviour of S, referred to here as the *context* C of S. These definitions follow those given in [3], whereby M is said to be an ϵ-Mirror of S, for a given context C, if and only if,

$$d^C(M^C(t), r^C(t)) \leq \epsilon \tag{1}$$

where $d^C(t)$ denotes some metric, and $r^C(t)$ denotes the response of the system to a time series set $\{t_i\}_{i=0}^T$ for the context C. Suppose the model M^C has been validated and shown to be an ϵ-mirror of S^C. The question then is: *for a new structure S', behaving in a context given by small changes ΔS and ΔM, is the corresponding M' a mirror of S'^C for some value of ϵ'?*

Although the model proposed here aims to address the problem of scarce labelled datasets, it is not its main motivation. The benefits of doing so appear when trying to answer the question above. Interestingly, the learning bounds found in *statistical learning theory* [4] can help validate M' without needing to build a prototype S'. As a result, the method presented in this paper is fundamentally based on these learning bounds to develop an algorithm that addresses the original problem of limited datasets.

The proposed algorithm relies on a novel adversarial neural network designed to minimise an upper bound on the expected error for the target domain. This approach is achieved by combining some available (unlabelled) data with (labelled) data deriving from an external source. The neural network learns a reduced latent space representation of the data, where the different datasets are aligned, enabling accurate classification across domains.

2 THEORY

2.1 The problem of transfer learning

The transfer learning problem is formalised here by establishing the following definitions:

1. *A domain $\mathcal{D}$, consisting of a marginal probability distribution p on a feature space $\mathcal{X}$. Therefore, $\mathcal{D} = \{\mathcal{X}, p(X)\}$, where $X = \{\mathrm{x}_i\}_{i=1}^n \in \mathcal{X}$ is a finite sample set from $\mathcal{X}$.*

2. *A task $\mathcal{T} = \{\mathcal{Y}, f(\cdot)\}$, for a given domain $\mathcal{D}$, defining an output space $\mathcal{Y}$ and a labelling function $f : \mathcal{X} \to \mathcal{Y}$.*

Two distinct domains are then established: (1) a *source domain* $\mathcal{D}_S$, and (2) a *target domain* $\mathcal{D}_T$, each with their corresponding tasks $\mathcal{T}_S$ and $\mathcal{T}_T$. Therefore, the idea of transfer learning is to utilise knowledge about the source $\langle \mathcal{D}_S, \mathcal{T}_S \rangle$ to improve the learning of a target predictive function $f_t(\cdot)$ for a given target $\langle \mathcal{D}_T, \mathcal{T}_T \rangle$, where $\mathcal{D}_s \neq \mathcal{D}_T$ and $\mathcal{T}_S \neq \mathcal{T}_T$. In the following sections, this problem is reduced to the case in which no labelled data are available in the target domain. The learning problem under consideration is thus one of *domain adaptation*, which may be thought of a branch of transfer learning whereby it is assumed that $\mathcal{X}_s = \mathcal{X}_t$ and $\mathcal{Y}_s = \mathcal{Y}_t$, but that $p_s(X, Y) \neq p_t(X, Y)$, with $Y = \{\mathrm{y}_i\}_{i=1}^n \in \mathcal{Y}$.

2.2 Derivation of generalisation bound on target expected risk

Given a finite training set $\mathrm{z} = \{(\mathbf{x}_i, \mathrm{y}_i)\}_{i=1}^n$, a supervisor is introduced to learn some *hypothesis* $h : \mathcal{X} \to \mathcal{Y}$, aimed at approximating $f(\cdot)$. The probability that the hypothesis disagrees with the true labelling function is defined by the *expected risk*,

$$R(\alpha) = \int \mathcal{L}(\mathrm{y}, h(\mathrm{x}, \alpha)) p(\mathrm{z}) \mathrm{dz}, \quad \alpha \in \Lambda \tag{2}$$

which is defined by an expectation of some *loss* functional $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, computed with respect to the (unknown) joint distribution $p(\mathrm{z})$. In this expression, the predictive function is assumed to be a conditional distribution $p(\mathrm{y}|\mathrm{x})$. One should note that the set Λ to which α belongs can be a set of scalar quantities, vectors, or of abstract elements [5]; that is, the function space is not limited to parametric models. This expression holds under the assumption that the data are generated from a unique underlying probability distribution. In the context of transfer learning, the expected risk of this form can only hold true for one domain; therefore, when referring to the risk of a hypothesis in the source domain, equation (2) can be rewritten as,

$$R_s(\alpha) = \int \mathcal{L}(\mathrm{y}, h(\mathrm{x}, \alpha)) p_s(\mathrm{z}) \mathrm{dz}, \quad \alpha \in \Lambda \tag{3}$$

where the expectation is now explicitly evaluated with respect to source-domain elements. Similarly, the parallel notations R_t and p_t are used for the target domain. While the expected risk gives some insights into how a supervisor learns to generalise from data, the integral cannot be evaluated analytically without knowing the underlying generative distribution. Nonetheless, a way in which one can make progress is to upper bound expected risk by employing an *Empirical Risk Minimisation* inductive principle [4]. In this case, however, the target risk is upper bounded by using Jensen's Inequality, which is defined as follows for a concave function $g(\cdot)$,

$$\log \mathbb{E}_{p(z)}\{g(z)\} \geq \mathbb{E}_{p(z)}\{\log g(z)\} \tag{4}$$

The upper bound on the target risk can then be derived as follows,

$$\begin{aligned} \log R_s &\geq \int p_t(\mathrm{z}) \log \frac{\mathcal{L}(\mathrm{y}, h(\mathrm{x})) p_s(\mathrm{z})}{p_t(\mathrm{z})} \mathrm{dz} \\ &= \int p_t(\mathrm{z}) \log \mathcal{L}(\mathrm{y}, h(\mathrm{x})) \mathrm{dz} + \int p_t(\mathrm{z}) \log \frac{p_s(\mathrm{z})}{p_t(\mathrm{z})} \mathrm{dz} \\ &= \mathbb{E}_{p_t(\mathrm{z})}\{\log \mathcal{L}(\mathrm{y}, h(\mathrm{x}))\} - \mathrm{KL}[p_t(\mathrm{z}) || p_s(\mathrm{z})] \end{aligned} \tag{5}$$

The second term in the right-hand side is the negative of the Kullback-Leibler (KL) divergence between the target joint distribution and source joint distribution. Rearranging the expression yields an upper bound on the target risk,

$$\mathbb{E}_{p_t(\mathrm{z})}\{\log \mathcal{L}(\mathrm{y}, h(\mathrm{x}))\} \leq \log \mathbb{E}_{p_s(\mathrm{z})}\{\mathcal{L}(\mathrm{y}, h(\mathrm{x}))\} + \mathrm{KL}[p_t(\mathrm{z})||p_s(\mathrm{z})] \tag{6}$$

The full derivation of the upper bound is defined as,

$$\begin{aligned}\mathbb{E}_{p_t(\mathrm{x},\mathrm{y})}\{\log \mathcal{L}(h(\mathrm{x}), \mathrm{y})\} \leq \log \mathbb{E}_{p_s(\mathrm{x},\mathrm{y})}\{\mathcal{L}(h(\mathrm{x}), \mathrm{y})\} + \mathbb{E}_{p_t(\mathrm{x})}\mathbb{E}_{p_t(\mathrm{y}|\mathrm{x})}\{\log p_t(\mathrm{y}|\mathrm{x}) - \log p_s(\mathrm{y}|\mathrm{x})\} \\ + \mathrm{KL}[p_t(\mathrm{x})||p_s(\mathrm{x})]\end{aligned} \tag{7}$$

or equivalently, in an unsupervised framework,

$$R_t(\hat{\mathcal{L}}) \leq \log R_s(\mathcal{L}) + \mathrm{KL}[p_t(\mathrm{x})||p_s(\mathrm{x})] \tag{8}$$

where $\hat{\mathcal{L}} = \log \mathcal{L}$. The second term in (7) is ignored when the set of labelled target data is unavailable. Removing this term from the expression holds under the assumption that conditionals are equal across domains, and reduces to zero when $\mathrm{KL}[p_t(\mathrm{x})||p_s(\mathrm{x})]$ approaches zero. Therefore, by this assumption, minimising the right-hand side of (8) implies minimising $\mathbb{E}_{p_t(\mathrm{x})}\mathbb{E}_{p_t(\mathrm{y}|\mathrm{x})}\{\log p_t(\mathrm{y}|\mathrm{x}) - \log p_s(\mathrm{y}|\mathrm{x})\}$. It must be noted, however, that such an allowance can promote the likelihood of *negative transfer*.

2.3 Interpretation

The terms in equation (7) are readily interpretable. The first term in the upper bound represents the log of the source expected risk. The second term measures the difference between the tasks across domains. Similarly, the third term calculates the divergence between the input marginals of the domain. In other words, for a given hypothesis h, this expression estimates the expected risk on the target domain by including elements from the source domain. One can thus leverage on this upper bound to determine how effective an algorithm is at improving the transfer of knowledge across domains. For example, if the aim is to use data from the source domain to improve the generalisation performance in the target domain, some algorithm could be devised with the objective to minimise the terms in equation (7).

Unfortunately, these terms cannot be (rigorously) computed, for two main reasons:

1. The target conditional $p_t(\mathrm{y}|\mathrm{x})$ is unlikely to be known.
2. The marginal distributions $p_s(\mathrm{x})$ and $p_t(\mathrm{x})$ are only possible to estimate either by having an infinite amount of data or knowledge about the distributions.

Nevertheless, an attempt to approximate the upper bound with finite samples may offer solutions that could be useful from a more pragmatic point-of-view. In fact, it is often the case in which statistical algorithms are employed without explicitly acknowledging converging guarantees. A paradigmatic example that may be of relevance here relates to *Variational Auto-Encoders* (VAE) [6]. In fact, the terms in (8) resemble those used in the objective function of a VAE, with the main difference being that VAEs use the KL divergence to regularise the inference of some posterior distribution with respect to a corresponding prior distribution in a Bayesian framework. While it is emphasised that a non-rigorous handling of this problem will require some form of empirical validation, if sufficient data are available, the result of minimising (8) may still provide a valuable outcome in an engineering application.

2.4 Towards a neural network approach

Looking closely into the parallels such an approach would have with a VAE, it is similarly proposed here to have a deep neural network minimise the RHS of (8). There is, however, a prevalent limitation preventing a conventional VAE from taking over such an objective. In particular, the objective-function formulation of VAEs is often derived by assuming an isotropic Gaussian prior distribution, which simplifies the KL divergence metric to a tractable closed-form solution (as long as the likelihood is also assumed Gaussian). It is desired to relax these assumptions for the domain adaptation problem, since the shape of both the source marginal $p_s(\mathrm{x})$, and target marginal $p_t(\mathrm{x})$, are unlikely to be known in advance.

This limitation has been acknowledged in the literature, and adversarial networks have been proposed as an alternative to improve the flexibility of VAEs when inferring complex latent distributions. Of particular interest here is the approach proposed by Meschder et al. [7]. Their model preserves the essence of a VAE while incorporating a discriminative network trained to distinguish samples drawn either from the inferred posterior or from the prior distribution. The encoder is then tuned to produce samples aimed at fooling the discriminator without compromising the likelihood maximisation, and thus produce samples that resemble those from the prior distribution, regardless of their shape.

By adopting the network framework used in [7], the terms in (8) can be similarly minismised. The difference here, however, is that the marginals are transformed to be aligned in a reduced latent space, rather than inferring a posterior distribution that assimilates to a given prior distribution.

3 CASE STUDY

The current demonstration involved simulating an impulse response of some structure S, modelled as a lumped-mass model consisting of five masses $\{m_i\}_{i=1}^{5} = 1$, with spring coefficients $\{k_i\}_{i=1}^{5} = 10^5$, and damping coefficients $\{c_i\}_{i=1}^{5} = 20$. Three different states were considered in the simulation; corresponding to one healthy (normal) operating condition, and two different damaged conditions. In this case, one of the damaged states was simulated by degrading the stiffness of spring k_3 to 75% of its original value. Similarly, the remaining damaged state was simulated by degrading the stiffness of spring k_5 by the same amount. The features of interest were the damped natural frequencies at each health state, which were extracted from the system parameters via eigendecomposition. Finally, different sequences of n i.i.d. Gaussian samples were added to each of the natural frequencies to introduce noise corresponding to 1% RMS of the damped-natural frequencies. These quantities thus comprise the source features used for analysis, $X_s \in \mathbb{R}^{n\times 5}$.

In order to generate the target data, a modified structure S' was also constructed by including additional connections to the original lumped-mass model of S. Concretely, the connections here were included to masses m_2 and m_5. The models are shown in Figure 1a and Figure 1b, respectively. Following the same steps as before, a healthy state and two damaged states were then simulated for S'.

The premise of the transfer task being pursued here is to make use of the data gathered from S, to improve the predictive ability of the learner when presented with data from S'. To illustrate the benefits of employing the algorithm outlined above, a classifier was first trained only using source domain data; or similarly, by training the neural network without the adversarial discriminator.

Therefore, for the first part of this demonstration, the main architecture of the neural network

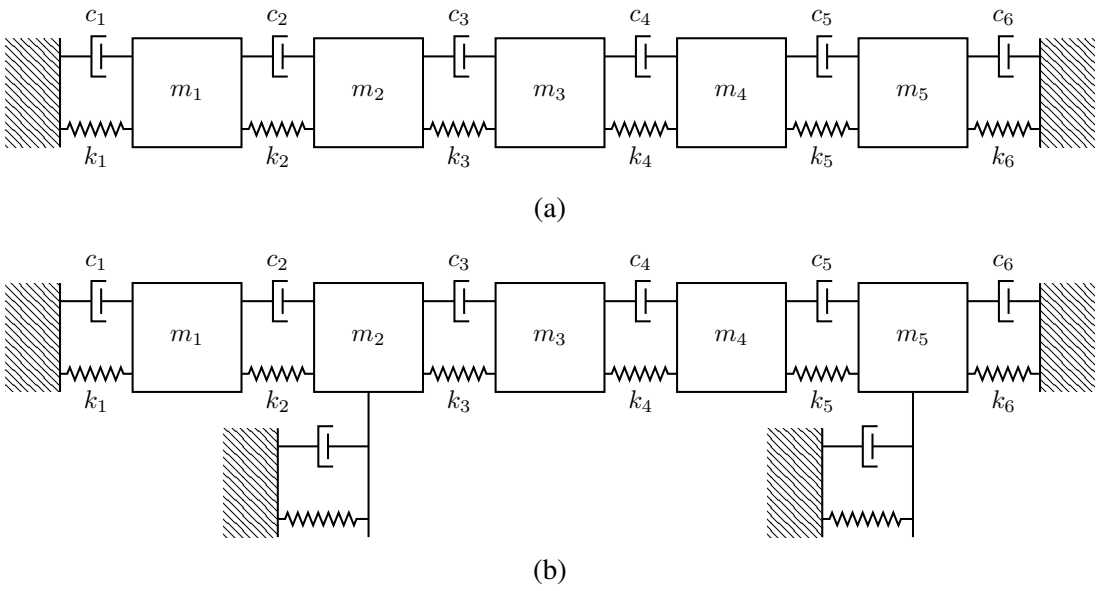


Figure 1: Lumped mass-damper-spring models: (a) Structure S and (b) Structure S'

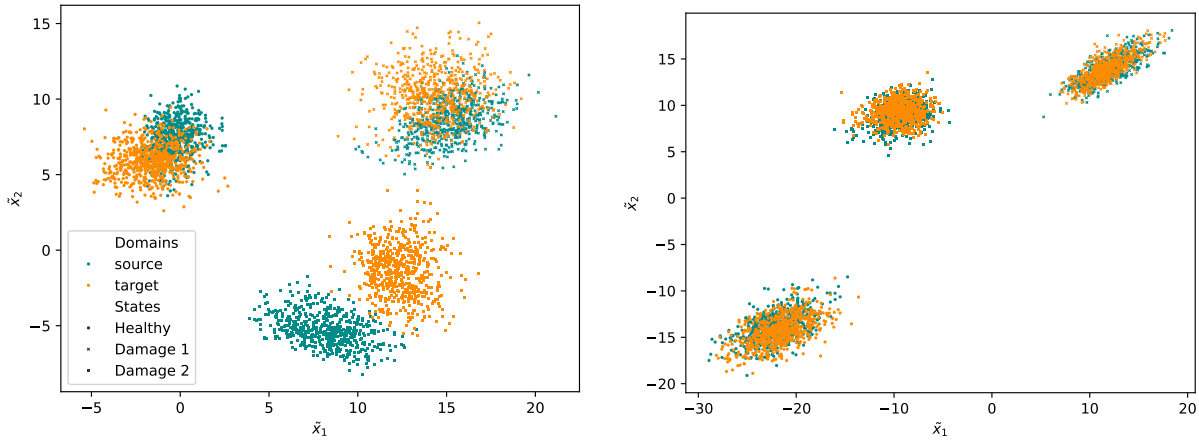
consisted of multi-layer perceptrons used to model the feature extractor g_ϕ and classifier h_θ. The feature extractor was designed with three hidden layers, each containing 128 nodes with *elu* activation functions, followed by a two-node linear output layer. The outputs from the feature extractor were then fed into the classifier, which also had three hidden layers of 128 nodes, each using *elu* activations. Since this task involved classification, a *softmax* output layer with three nodes was used-one for the healthy class and two for the damage classes, labelled as damage class 1 and damage class 2.

The source data deriving from structure S were split into a training set and a validation set. A simple preprocessing step was conducted in which the empirical mean was subtracted from the data. Additionally, the set of weights $\{\phi\}$ and $\{\theta\}$ were initialised with a *He standard* initialiser [8], and $L2$-regularised to mitigate the network from overfitting the training data. An *early-stopper* was also enabled for this purpose.

Although this exercise may seem trivial, the problem took a more interesting turn when feeding in data from the modified structure S'. On can note in Figure 2a the discrepancy that exists between source and target representations, which it is not all that surprising given that the feature extractor learnt from source data alone.

This approach required the addition of a discriminator to the main architecture of the neural network. Here, the discriminative network also consisted of a multi-layer perceptron, which was designed with five hidden layers, each containing 256 nodes and *elu* activations. A *sigmoid* output layer with a single node was used to determine the probability of a sample originating from the target domain. As before, the same preprocessing and regularisation measures were enforced to mitigate the possibility of overfitting.

Having trained the complete network, the entire validation set was fed to the feature extractor. Visually, as shown in Figure 2b, the latent-space representations of the domain marginals appear to have aligned nicely. With an accurate class alignment, the classifying network is able to correctly assign labels to both the source data and the target data. Therefore, the model can be said to have successfully transferred its ability to make predictions across domains.



(a) (b)

Figure 2: Latent-space representation: (a) pre-transfer and (b) post-transfer

4 DISCUSSION AND CONCLUSIONS

Despite the promising results shown in the previous section, there are a few considerations worth noting. The first and foremost, is the fact that the upper bound on the target expected risk defined in (8) is not rigorously computed by the neural network. Instead, the conducted strategy computes some empirical estimation of the upper bound in disregard of the model complexity when presented with finite amounts of data. In other words, the risk estimate is inaccurate, and therefore, no real guarantee can be made in terms of how well the model can generalise to new data. Nevertheless, this consideration is commonly overlooked in practice, particularly when adopting deep learning models to solve learning problems. The generalisation problem is rather addressed implicitly by assuming that enough data are available to ensure some reasonable degree of generalisation, in addition to regularisation measures that are often used in a more heuristic manner.

Another important consideration worth highlighting is the challenging validation process involved in tuning the hyperparameters of the network. Designing the architecture required deciding on the number of hidden layers, number of nodes and activation functions among other relevant factors. In this case, these were manually tuned to yield good-enough results for the sake of the demonstration. The appropriate model validation will be pursued in future work with more stringent strategies for optimal hyperparameter selection.

The benefits of this model over other methods for unsupervised domain adaptation will also be explored via appropriate benchmarking tests. Current state-of-the-art models should be examined to determine the conditions under which it is worth preferring the current model. For example, the unsupervised framework presented above resembles that of the *domain-adversarial neural network* (DANN) [9], developed from the upper bound estimates on the target expected risk proposed by Ben David et al. [10]. An immediate distinction that can be made at this stage, however, is that the current version can be somewhat interpreted as a generative version of the DANN, and thus account for uncertainty in the predictions. Having a generative model in this context may benefit regression-type problems whereby the prediction is of some dynamic response across domains. This exercise might also be worthy of further investigation.

Finally, further developments will be investigated by adopting a semi-supervised setting via the incorporation of the second term in (7).

5 ACKNOWLEDGMENTS

The authors of this paper gratefully acknowledge the support of the UK Alan Turing Institute via funding of the Turing Research and Innovation Cluster on Digital Twins. For the purpose of open access, the authors has applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising.

REFERENCES

[1] C. Farrar and K. Worden, *Structural Health Monitoring: A Machine Learning perspective*. Wiley, 2013.

[2] K. Weiss and T.M. Khoshgoftaar and D. Wang, A survey of transfer learning, *J Big Data*, 3(1):1–40, 2016.

[3] K. Worden and E.J. Cross and R.J. Barthorpe and D.J. Wagg and P. Gardner, On digital twins, mirrors, and virtualizations: Frameworks for model verification and validations, *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering*, 6(3): 030902, 2020.

[4] V. Vapnik, *The Nature of Statistical Learning Theory*. Springer New York, 1999.

[5] V. Vapnik and S. Kotz, *Estimation of Dependences Based on Empirical Data*. Springer New York, 2006.

[6] D.P. Kingma, Auto-encoding variational Bayes, *arXiv*, 2013.

[7] L. Mescheder and S. Nowozin and A. Geiger, Adversarial variational Bayes: Unifying variational autoencoders and generative adversarial networks, *International Conference on Machine Learning*, 2391–2400, 2017.

[8] K. He and X. Zhang and S. Ren and J. Sun, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification, *Proceedings of the 2015 IEEE International Conference on Computer Vision*, 1026–1034, 2015.

[9] Y. Ganin and E. Ustinova and H. Ajakan and P. Germain and H. Larochelle and F. Laviolette and M. March and V. Lempitsky, Domain-adversarial training of neural networks, *Journal of Machine Learning Research*, 17(59):1–35, 2016.

[10] S. Ben-David and J. Blitzer and K. Crammer and A. Kulesza and F. Pereira and J.W. Vaughan, A theory of learning from different domains, *Machine Learning*, 79:151–175, 2010.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

UNCERTAINTY QUANTIFICATION OF WIRE TOLERANCES IN COIL ASSEMBLY

Romas Zubavicius[1], Maria Nogal[1], Gan Fu[2], Mitrofan Curti[2], Oswaldo Morales-Nápoles[1]

[1]Delft University of Technology
Delft, the Netherlands
{R.Zubavicius, M.Nogal, O.MoralesNapoles}@tudelft.nl

[2] Eindhoven University of Technology
Eindhoven, the Netherlands
{G.Fu, M.Curti}@tue.nl

Abstract. *Manufacturing imperfections are an inherent aspect of the production process, affecting the reliability and performance of all engineered components, including actuators. This work investigates the effects of manufacturing uncertainty in coil assembly. Given the manufacturing tolerances of geometrical and material properties of wires and coils, the coil's electrical response is studied from a probabilistic perspective. While many researchers acknowledge these uncertainties, they often do not quantify their effects on the system's response. We propose the use of copulas to model nonlinear relationships and quantify the probabilistic dependence between design variables (i.e., wire and coil's properties) and their effects on electrical responses. Additionally, tolerances of the design variables are determined through unstructured expert elicitation. The results show the impact of the thickness of the wire's insulation in the performance of the coil. Furthermore, we establish a design space that is studied probabilistically, allowing for probabilistic design optimization.*

Keywords: Uncertainty Quantification, Copula, Coreless Electric Motor, Coils, Multi-physics model

---

1 INTRODUCTION

Variations in manufacturing imperfections are expected during the manufacturing of coil assemblies used in actuators. The coil's electrical properties depend on the wire's geometrical and material parameters in addition to the coil's geometry, and its positioning with respect to the entire topology of the actuator. Consequently, many parameters, such as electrical properties, may vary between coils depending on the tolerances of the manufactured wire and coil assembly. Therefore, manufacturing tolerance is critical for the overall performance, reliability, and efficiency. However, achieving higher manufacturing precision leads to rapidly increasing costs and, beyond a certain level, such fine precision may become unnecessary depending on the specific application. Therefore, it is of interest to know what variables engineers have to consider when designing, manufacturing, and assembling devices to improve quality control.

Linear actuators are types of devices that produce force for a linear motion, i.e. to move objects in a straight line. Such devices are ubiquitous in various applications, from everyday uses, such as standing desks, to more industrial applications, such as assembly lines in manufacturing, to high-tech applications in scientific equipment, where precise movement control is in nanometers [1]. Achieving precise control requires the precise manufacturing and assembly of such devices. Probabilistic approaches to quantify uncertainties in the design stage can serve to improve quality control. More specifically, understanding the probabilistic dependencies between the variables involved in the manufacturing and assembly processes of linear actuators and studying how they affect the performance metrics help engineers to design devices under uncertainties and make improvements based on desired outcomes.

In this work, we aim to identify the probabilistic dependencies between the variables involved in the production of actuators, and how they can be characterized from a probabilistic point of view. More specifically, we study the dependence between design uncertainties, such as material properties, manufacturing and assembly tolerances, and how these uncertainties affect the performance of the actuator.

In actuator design, researchers often consider uncertainties but do not quantify their effects. Two common approaches to overcome the lack of uncertainty quantification are mentioned in [2] : i) designing under uncertainty using robust design optimization methods, in which the uncertainty characterization is captured by a range of uncertain conditions, and ii) avoiding certain manufacturing processes and opting for alternative designs that are less sensitive to uncertainties. On the other hand, the finite element method (FEM) is commonly employed to analyse the behaviour of actuator designs, which can be computationally intensive. This limits the number of uncertainties that can be studied. E.g., an article by Jiqi Wu et al. [3] only considered magnet material uncertainties. Similar approaches are found in [4, 5].

Uncertainty Quantification (UQ) methods vary from classical approaches, such as Monte Carlo (MC) simulations and Polynomial Chaos Expansion (PCE) to deep learning surrogate models [6]. While PCE can fall short due to orthogonal basis assumptions, and machine learning techniques are usually black boxes, copulas provide the advantage of having probabilistic interpretations, as well as their capability to model nonlinear relationships.

A general framework for UQ has been previously proposed using copulas [7], which considers complex input dependencies (e.g. multi-physics) via vine-copulas, as well as similar works which address multidimensional correlations [8]. While the frameworks exemplify methodologies on e.g. truss structures, or multiscale composite material, as per the authors' knowledge, it has never been applied to electric motors.

Existing methods for estimating uncertainties in manufacturing and assembly are often lim-

ited due to their dependence on complex processes, machine variability, and handling conditions. To address the challenge of acquiring uncertainties of manufacturing and assembly of actuators, we propose uncertainties of the design variables through expert elicitation. Furthermore, to expand on the proposed uncertainties, and quantify their effects, probabilistic dependencies were studied using copulas and the implications of these uncertainties were examined.

The remainder of the document is as follows; copulas are briefly introduced in Section 2. The multiphysics nature of actuators and the corresponding probabilistic modelling using copulas is explained in Section 3. The results are presented and discussed in Section 4, with concluding remarks in Section 5.

2 PROBABILISTIC DEPENDENCE VIA COPULAS

Copulas are flexible functions used to model multivariate probabilistic dependence [9]. The advantages of copulas become apparent for nonlinear relationships between random variables, because classical approaches, such as using multivariate normal distributions and Pearson correlation coefficients, fall short of explaining nonlinear behaviour. Copulas are prevalent in the finance sector [10], and in recent years, they have been applied more frequently to engineering problems [11]. The viability of copulas originates from Sklar's theorem [12], which states that for a joint distribution function H with margins F_i, there exists a copula C for all $(x_1, ..., x_d) \in \mathbb{R}^d$ such that

$$H(x_1, ..., x_d) = C(F_1(x_1), ..., F_d(x_d)) \tag{1}$$

The margins F_i are cumulative distribution functions (CDF), describing non-exceedance probability $F(x) = P(X \leq x)$. If the same underlying physics governs two systems, their dependence structure (and thus their copula) is likely to be similar. Therefore, system behaviour can be characterized via copulas. The common measure of dependence for copulas utilises rank correlations, which assess the relationship between variables based on their relative rankings within the dataset, rather than their values. In this work, Kendall's rank correlation (τ) [13], is used as a metric to check the significance of the bivariate distribution.

There are two common types of copula families - elliptical and Archimedean. Elliptical copulas are always symmetric. Archimedean copulas, on the other hand, can model various dependence structures, including some symmetries and tail dependencies. Tail dependence measures how strongly extreme values of random variables are correlated. Probabilistically, the upper tail (λ^U) and lower tail (λ^L) dependence coefficients are

$$\lambda^U = \lim_{t \to 1^-} P(X_1 > F_1^{-1}(t) \mid X_2 > F_2^{-1}(t)) \tag{2}$$

$$\lambda^L = \lim_{t \to 0^+} P(X_1 \leq F_1^{-1}(t) \mid X_2 \leq F_2^{-1}(t)) \tag{3}$$

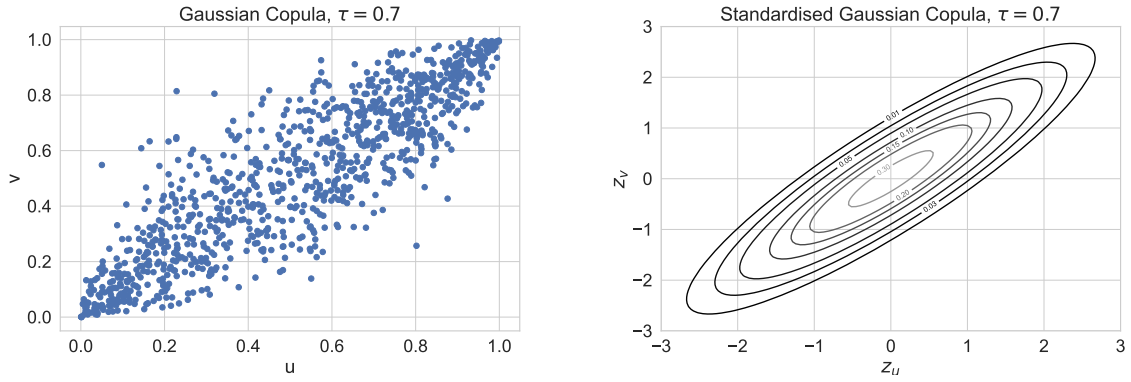
defined as the limit (if it exists) for λ^U and λ^L as the conditional probability. It represents the asymptotic probability that X_1 is greater than the threshold $F_1^{-1}(t)$, given that X_2 is greater than $F_2^{-1}(t)$, as t approaches 1 from the left. Since t is arbitrarily close to 1 but never exactly 1, this captures the probability of extreme co-movements in the upper tail of the joint distribution. For lower tail dependence, it is the opposite, with the condition that is less than or equal to the quantile function evaluated at t, as t approaches 0 from the right. When $\lambda^U = \lambda^L = 0$, implies no tail dependence. Our work has two dominant types of copulas: Gaussian and Frank. These copulas are elliptic, have no tail dependence and are characterized by a single parameter.

Gaussian and Frank copulas are defined as

$$C_{\text{Gaussian}}(u, v; \rho) = \Phi(\Phi^{-1}(u), \Phi^{-1}(v); \rho). \tag{4}$$

$$C_{\text{Frank}}(u, v) = \frac{1}{\theta} \ln \left(1 + \frac{(e^{-\theta u} - 1)(e^{-\theta v} - 1)}{e^{-\theta} - 1} \right). \tag{5}$$

where for Gaussian copula, Φ^{-1} is the inverse of standard normal CDF, and the dependence between two variables is controlled by correlation coefficient ρ. The dependence in Frank copula is described by parameter θ, with $\theta > 0$ indicating a positive relationship, and $\theta < 0$ negative. Similar to the Gaussian copula, it has symmetric dependence, however, it captures strong central dependence and weaker dependence in its tails. To facilitate a more intuitive interpretation of the dependence structure, the copula unit space (u, v) is transformed into standard normal space (z_u, z_v). This transformation is achieved by applying the inverse CDF of the standard normal distribution. It is defined as $z_u = \Phi^{-1}(u)$ and $z_v = \Phi^{-1}(v)$, where u and v are uniform $[0, 1]$ variates. If u and v originate from a Gaussian copula, the transformed plot will exhibit characteristics consistent with the bivariate normal. The visualization of the Gaussian copula in unit space is shown in Figure 1a, and the standardized copula is shown in Figure 1b, and Frank copula is shown in Figure 2a, and in standard normal space Figure 2b.



(a) (b)

Figure 1: Bivariate Gaussian copula of (u, v) with Kendall $\tau = 0.7$ in (a) unit space, and (b) standard normal space.

3 PHYSICAL MODEL OF THE ACTUATOR

The actuator under study is a Permanent Magnet Linear Synchronous Motor (PMLSM) with a single-sided permanent magnet (PM) configuration as depicted in Figure 3.

It employs rectangular wire for the winding. It features a configuration of four magnets, three coils, and six stacks. Each coil is wound with 261 wire turns while stacking refers to the arrangement of multiple coils positioned on top of one another. For an application-specific case, a 10 kg load is assumed for the motor. This case has been selected because mechanical performance, such as force ripples are less present in coreless motors; however, manufacturing tolerances are relatively more significant. The design variables and the associated uncertainties and the multiphysics model used to study this case are described in subsequent sections.

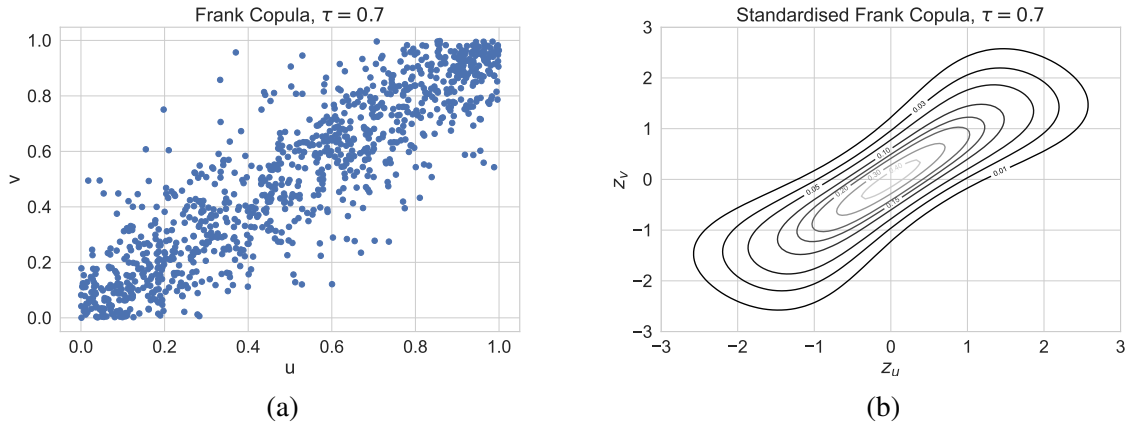


Figure 2: Bivariate Frank copula of (u, v) with Kendall $\tau = 0.7$ in (a) unit space, and (b) standard normal space.

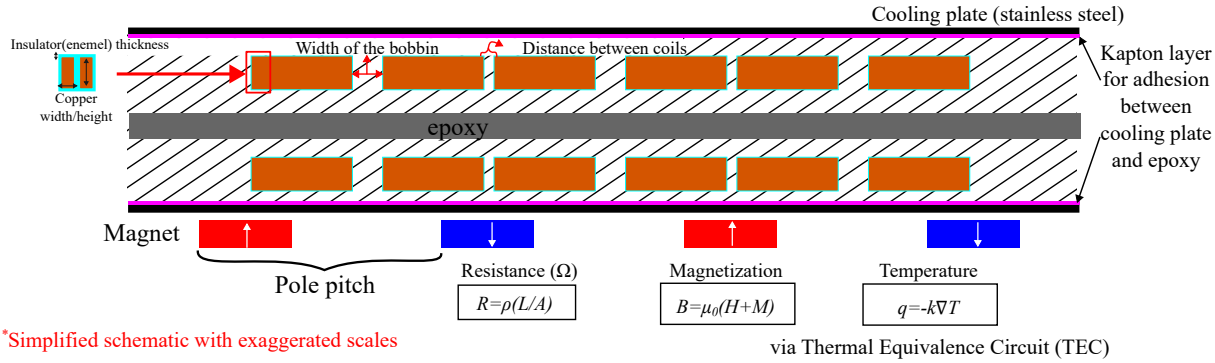


Figure 3: Illustration of a cross-section of a coreless linear motor, with 4 magnets and 3 coils. Epoxy is filled and encapsulates the coils. Each coil has 261 turns and is a 6-stack configuration.

3.1 Design variables of the actuator

The initial design variables are acquired from [14] and adjusted for high-voltage applications, resulting in a set of 20 input variables. Their description and uncertainty characterization are indicated in Table 1. Out of the 20 input variables, 18 are considered random variables. The exceptions are the air gap length between the coil and PM due to complex dependencies on the tolerances of epoxy and cooling plates. The rectangular part has a strong influence on the force, but is considered to be invariant, because the manufacturing tolerance is considered to reflect on the bent part, which considered has no contribution to the force.

The uncertainties presented in the table define the lower and upper bounds. Since obtaining precise uncertainty estimates is challenging, the uncertainty bounds were acquired through expert elicitation, by sending a questionnaire to experts with the design variables, and requesting their estimates of expected deviations. The generalized Beta distribution is used for the majority of the variables as it offers the advantage of being bounded and allows for nonsymmetric distributions. The Beta probability distribution function is defined on the interval $[0, 1]$, meaning that the random variable $X \sim GBeta(\alpha, \beta, x_L, x_U)$ defined in the range $[x_L, x_U]$ has to be transformed into the variable $Z \sim Beta(\alpha, \beta)$ through an affine transformation. Thus, $Z = \frac{X - x_L}{x_U - x_L}$. The uncertainty distributions are assumed to be symmetric, hence the shape parameters were chosen to be equal, resulting in a Gaussian shape distribution, with the advantage of being bounded. Given that the distributions depend on the manufacturing pro-

cesses, which are difficult to describe [2], the parameters of the beta distributions are assumed $\alpha = \beta = 7.5$. This choice should be validated. These parameters result in Beta distributions with $E[Z] = 0.5$ and standard deviation $\sigma_Z = 0.125$, which transformed to the original variable results in $E[X] = \frac{x_U + x_L}{2}$ and $\sigma_X = \frac{x_U + 7x_L}{8}$, respectively. Uniform distributions were chosen to represent less well-defined uncertainties.

Design variable [units]	Expected value	Support range	Probability Distribution
Width wire [mm]	0.06	$\pm$2%	GBeta
Height wire [mm]	1.27	$\pm$2%	GBeta
Thickness insulation [mm]	0.015	$\pm$10%	Uniform
Pole pitch [mm]	42	$\pm$1%	GBeta
Length rect. part [mm]	70	-	Deterministic
Radius of bobbin [mm]	4	$\pm$1 turn thickness	GBeta
Height of PM [mm]	10	$\pm$0.1 mm	GBeta
Length of PM [mm]	70	$\pm$0.15 mm	GBeta
Height stainless steel [mm]	0.25	$\pm$1%	GBeta
Height epoxy layer [mm]	0.2	$\pm$2.5%	GBeta
Height cooling plate [mm]	1	$\pm$1%	GBeta
Height kapton [mm]	0.05	$\pm$12 μm	GBeta
Distance between coils [mm]	1	$\pm$1 turn thickness	GBeta
Airgap length [mm]	0.5	-	Deterministic
Density epoxy [kg/m^3]	1300	$\pm$10%	Uniform
Density steel [kg/m^3]	8000	$\pm$0.3%	GBeta
Specific heat capacity epoxy [J/(kg°C)]	1100	$\pm$10%	Uniform
Thermal cond. epoxy [W/(mk)]	1.2	$\pm$20%	Uniform
Resistivity [$\Omega \cdot \mathrm{m}$]	1.724e-8	$\pm$0.5%	Uniform
Remanence [T]	1.23	$\pm$3%	GBeta

Table 1: Set of design and material properties used to run the simulation. Uncertainty of the variables was acquired via unstructured expert elicitation.

3.2 Multiphysics model of the actuator

A comprehensive multi-physics model has been developed as a design tool for actuator topology optimization. This model evaluates actuator performance across various topologies under a given motion profile and mass load. It integrates electrical, magnetic, mechanical, and thermal behaviours to achieve a thorough assessment. The design illustration can be seen in Figure 3.

For clarity, the model is structured into three main components: Inputs, Physical Models, and Outputs.

1) In the inputs block: we pre-define the geometric dimensions of the components of the actuator, namely, wire's height, thickness, with a certain thickness of enamel layer; number of turns for one coil; stacks of coils for one phase; together with the dimension of the epoxy potting layer, Kapton insulation layer, and the stainless steel cooling plates. In addition to the geometric structures, the environmental conditions and the materials' properties are predefined in this block. Most importantly, the fourth-order (position, velocity, jerk and snap) motion profile (shown in Figure 4a) and mass load are prescribed.

2) The core of the model is the multi-physics model. First, the prescribed motion profile dictates the mechanical requirements, enabling the evaluation of force density, motor constant, and different voltage components (seen in Figure 4b), which are inherently coupled with electrical excitations and magnetic field modeling. Second, a harmonic modelling technique based on Fourier analysis [15, 16] is employed to solve the 2D magnetic field distribution of the actuator, considering the structure of PMs and coils. Additionally, the inductance of the winding is modelled by analyzing the flux linkage induced by a unified single-phase current excitation, enabling precise calculation of inductive voltage components. Lastly, a 3D thermal equivalent circuit (TEC) [17] is implemented to predict the thermal behaviour of the system. This thermal model is coupled with the copper losses, determined by the computed three-phase current required to achieve the motion profile. Notably, the TEC model accounts for the bent geometry of the coils and their anisotropic thermal conductivity.

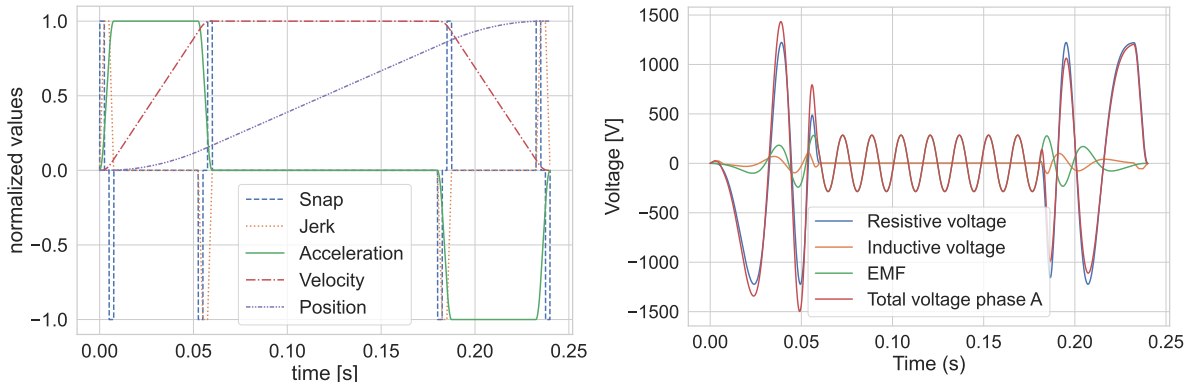
3) The model generates a diverse set of performance metrics, including fill factor; force density; steepness; motor constant; and terminal voltage, which includes Electromotive force (EMF) as well as resistive and inductive voltage components; phase currents; power consumption (copper losses and mechanical output); magnetic field distribution; and coil/insulation temperatures, etc. These outputs facilitate a comprehensive evaluation of different topological designs and aid in assessing the impact of uncertainties on actuator performance.

3.3 Performance metrics of the actuator

As mentioned in the previous section, several performance metrics are defined to evaluate the motor design: namely voltage, force density, power density, motor constant, fill factor, and temperature (calculated via TEC).

3.3.1 Voltage

Voltage calculations are based on the motion profile (as seen in Figure 4). The total voltage



(a) (b)

Figure 4: Required voltage is obtained based on motion profile. (a) Motion profile; (b) Voltage of a single phase

has three components: resistive, inductive, and electromotive force (EMF), for each of the three phases. The equation associated with one of the phase voltages is as follows:

$$V_{\text{an}} = I_a R_a + L_{aa}\frac{dI_a}{dt} + M_{ab}\frac{dI_b}{dt} + M_{ac}\frac{dI_c}{dt} + E_a \tag{6}$$

where the first component ($I_a R_a$) is the resistive voltage, three inductive voltages with (L_{aa}) self-inductance, and (M_{aa}, M_{ab}) mutual inductances and the EMF component (E_a) which is proportional with the speed of the mover. Subscripts a, b, c signify the three-phase electrical system. The EMF equation is given as

$$E_n = \frac{\partial \lambda_{\mathrm{n,pm}}}{\partial t} = \frac{\partial \lambda_{\mathrm{n,pm}}}{\partial x}\frac{\partial x}{\partial t} = \frac{\partial \lambda_{\mathrm{n,pm}}}{\partial x}\nu \tag{7}$$

where λ is the flux linkage in one phase, which is calculated from the 2D harmonic model.

3.3.2 Force density

Force density quantifies the force per unit volume of the actuator and is defined as

$$\text{Force density} = \frac{F}{V_{\mathrm{vol}}}, \tag{8}$$

where F is the force exerted by the actuator, calculated as the product of its mass and the acceleration derived from the motion profile, plus the friction force.

3.3.3 Power density

Similar to force density, power density describes the power output relative to volume and is given by

$$\text{Power density} = \frac{I \cdot V_{\mathrm{tot}}}{V_{\mathrm{vol}}}. \tag{9}$$

3.3.4 Motor constant

The motor constant is given by the ratio F/I. It quantifies how much force the actuator generates with a unit current excitation. Force is position independent, linear with current, and maximized if $\theta_0 = 90$. Therefore, it can expressed as follows

$$F_x = \sum_{n=a,b,c} \frac{E_n I_n}{\nu} = \frac{3\pi}{2\tau}\lambda_{\mathrm{pm}} I \sin(\theta_0) = K_t I \sin(\theta_0) \tag{10}$$

where K_t is the motor constant, and θ is the commutation angle, which is the phase shift between the magnet's magnetic field and the coil's magnetic field.

3.4 Filling factor

The filling factor is another key design variable used in the design considerations, affecting thermal management and electrical performance

$$\text{Filling Factor} = \frac{A_{\mathrm{conductor}}}{A_{\mathrm{total}}} \tag{11}$$

where $A_{\mathrm{conductor}}$ is the conductor cross-sectional area. It is calculated as the ratio of the conductor to the total available area, which includes enamel.

3.5 Probabilistic modelling

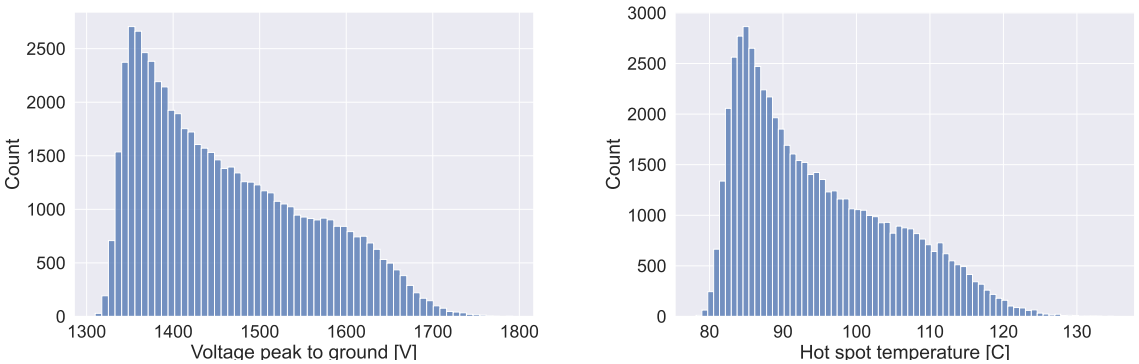
The multiphysics simulation was executed for 60,000 iterations, with each iteration sampling variables from the predefined probability distributions, following a standard Monte Carlo approach. All simulation data were stored in a CSV file, including the sampled design variables and the corresponding results. Bivariate copulas were fitted to the data to analyze the dependence structure between inputs and outputs. The fitting was performed using pyvinecopulib [18]. The following copulas with rotations were analyzed: Gaussian, Student, Clayton, Gumbel, Frank, BB1, BB6, BB7, BB8. The most appropriate copula model was selected based on the Akaike Information Criterion (AIC) because it penalizes overfitting, and allows a fair comparison between models.

4 RESULTS & DISCUSSION

This section presents the results of bivariate probabilistic modelling. Section 4.1 discusses the results between inputs and outputs of the model and Section 4.2 discusses dependencies between outputs, studying the design space probabilistically.

4.1 Dependencies between design variables and performance metrics

Small uncertainties in the assembly and material properties propagate and amplify in the performance of the manufactured component. For example, the variability ranges observed in the input parameters result in voltage peaks ranging from 1342 V at the 5th percentile to 1639 V at the 95th percentile, and temperature ranging from 82 °C at the 5th percentile to 114 °C, 95th percentile, as shown in Figure 5.

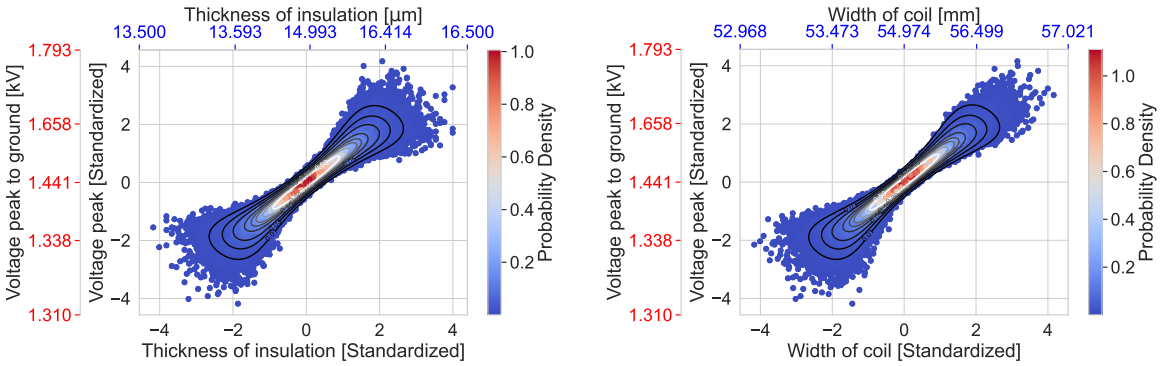


(a) (b)

Figure 5: Multiphysics model outputs distributions shown in (a) voltage, and (b) temperature

The bivariate distributions between the inputs (design variables) and outputs (simulation results) reveal that the wire insulation thickness is the most significant input variable, as it shows the highest rank correlation with voltage. On the other hand, the coil width, which is a latent variable determined by the wire's geometric properties, exhibits the highest rank correlation with the same variable. The best-fitted copula between the thickness of insulation and all performance metrics is the frank copula, indicating a stronger correlation at the median, i.e., values closer to the reference design. The standardised scatter plot of the data transformed to standard normal, along with the standard normal transformation of the fitted copula contour

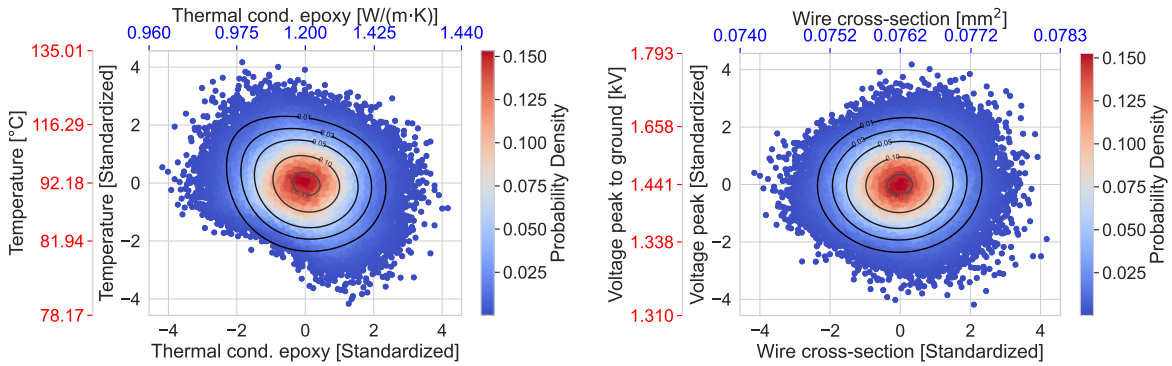
plot is shown in Figure 6a. The insulation thickness affects the geometrical properties of the assembled coil, such as its width and height, and this insulation significance is apparent for the effective coil dimensions (as seen in Figure 6b). The significance of insulation thickness, for performance and reliability is further highlighted in [19]



(a) (b)

Figure 6: Standardized copula results between (a) thickness of wire insulation and voltage, and (b) width of the coil and voltage. In standard axes, the median value is at (0,0). The contour plot is shown directly from the fitted copula.

Some of the design variables do not show a strong dependence under the defined uncertainties. For example, variations in the thermal conductivity of epoxy exhibit a weak correlation with the analysed performance metrics, such as temperature (see Figure 7a). However, two



(a) (b)

Figure 7: Standardized copula between (a) thermal conductivity of the epoxy and temperature and (b) cross-sectional area of the wire and voltage. In standard axes, the median value is at (0,0). The contour plot is shown directly from the fitted copula.

regions of interest can be identified: lower temperatures are not achievable at lower thermal conductivities, indicating that epoxy thermal resistance (proportional inverse to the thermal conductivity) is more significant in the system design. Conversely, at higher thermal conductivities, the thermal resistance from the epoxy becomes less dominant, allowing for better cooling. This illustrates a clear example of a probabilistic design space, where possible designs can be

probabilistically determined under uncertainties. Similarly, in contrast to the wire insulation, the cross-sectional area of the wire does not exhibit a strong correlation to any of the performance metrics, see Figure 7b. This relationship has a weak tail dependence, at lower cross-sectional area, the voltage variations are relatively smaller, compared to higher cross-sectional area. Design variables such as the bobbin radius, PM length, stainless steel height, PM height, Kapton height, epoxy layer height, cooling plate height, coil spacing, density, and specific heat capacity generally do not exhibit strong bivariate dependence on the performance metrics (see Tables 2 in the appendix). While wire resistivity does impact the total resistance, its influence is less significant compared to the geometric parameters.

4.2 Dependencies between performance metrics

The performance metrics describe the motor's efficiency. Analyzing the dependencies between performance metrics, such as voltage, temperature and motor constant allows defining the probabilistic design space. The probabilistic design space describes, for instance, under which voltage or what temperature ranges can be achieved from a probabilistic view.

In our case study, an important consideration is that this study focuses solely on uncertainties by introducing small variations in the design variables and analyzing their effects. If the design space were expanded, i.e. different topologies of the actuator were considered, the bivariate dependence could change due to nonlinear physical behaviour emerging at different parameter ranges. Nevertheless, several key insights are gained. An interesting tail dependence between voltage and motor constant has been observed. Figure 8a shows that at low voltages, higher motor constants are achieved, the inherent reason for this is that optimal design requires less current to achieve the same force and motion profile, which leads to higher motor constant and lower voltage, that's why we see the low voltage is highly correlated with higher motor constant. Vice versa for the high voltage-low K_t; however, the motor constant presents more variability in these range of values (upper left side of the figure). It is also observed that the expected design variables are still strongly correlated (central part of the figure), exhibiting very small variations in performance metrics, as explained in the previous section. When these optimal geometric proportions are altered, the motor constant deviates more significantly from expected values, resulting in increased variability, however, with a strong dependence showing motor inefficiency (lower right side of the figure).

Relatively similar results can be observed with temperature, showing that a higher required voltage also results in a higher temperature of the motor (Figure 8b). Interestingly, frank copula, which has no tail dependence has a better fit than other copulas with tail dependences based on the AIC. The lower and median voltage values exhibit similar temperature behaviour to the motor constant; however, the upper tail seemingly has more spread than the motor constant did. Table 3 in the appendix summarizes bivariate dependencies between performance metrics.

5 CONCLUSIONS & FUTURE WORK

In this paper, the propagation of the uncertainties associated with manufacturing, assembly and material properties to the performance metrics of a coreless linear motor are quantified with probabilistic methods. First, expert elicitation is performed to estimate design uncertainties, and probability distributions are defined. Secondly, a standard Monte Carlo approach is utilised with 60,000 runs of a multiphysics model of the coreless linear motor capturing the interdependencies between electrical, mechanical, magnetic and thermal behaviours. Then, bivariate copulas are fitted between the inputs and outputs of the model, quantifying the bivariate probabilistic

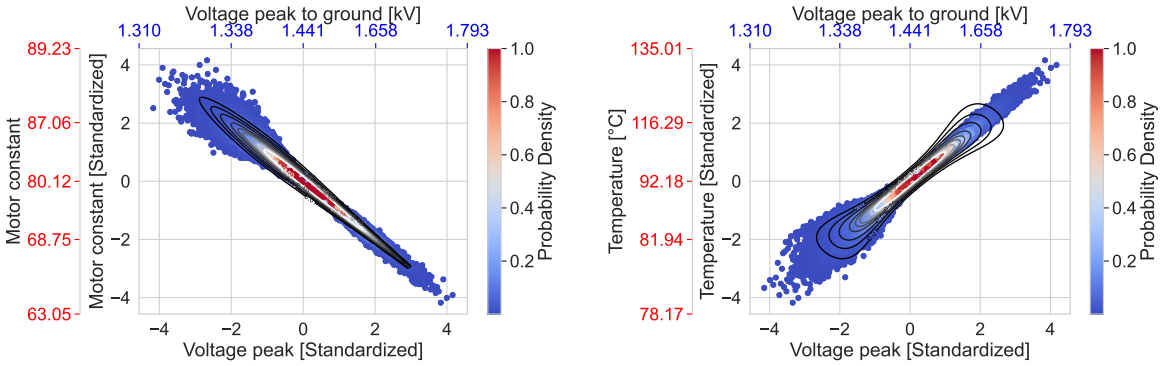


Figure 8: Probabilistic design space given by the standardized copula between (a) voltage and motor constant and (b) voltage and temperature. In standard axes, the median value is at (0,0). The contour plot is shown directly from the fitted copula. Figure a) shows motor inefficiency, by requiring higher voltage for the same application, and b) shows the temperature increase with higher voltage.

relationship and mapping the probabilistic design space. The results show that the uncertainty of the thickness of insulation plays the most important role in the design of the coreless linear motor. By increasing insulation thickness, it increases the mass of the actuator, while also affecting the ratios between the width of the coil and magnet, decreasing the overall efficiency. Other dependence structures, such as the bivariate copula between the thermal conductivity of the epoxy and temperature or the copula between the cross-sectional area of the wire and voltage do not show a dominant dependence under uncertainties; However, the scatter plots reveal regions of interest, indicating ranges of possible design configurations where performance metrics exhibit favorable characteristics or heightened sensitivity to specific parameter variations. Tables 2 and 3 in the appendix summarize all bivariate dependencies indicating the best-fit copula and rank correlation. The color scheme ranges from green, indicating a positive correlation, to red, representing a negative correlation. Text in bold highlights the discussed copulas in this paper. There are several limitations to this work. To manufacture and assemble these actuators is very costly, limiting the validation of these results. If we consider validation with only the coils, this would imply completely different assumptions, which would be invalid for this case study. Additionally, only a bivariate case was considered, which does not capture complex dependencies between multiple variables. Vine-copulas could be considered for multi-physics case studies.

REFERENCES

[1] Awaddy, B.A. and Wu-Chu Shih and Auslander, D.M., *Nanometer Positioning of a Linear Motion Stage under Static Loads*. IEEE/ASME Transactions on Mechatronics, 113-119.

[2] Y. Cheng, L. Ding, T. Zhao and S. Cui, *Design and Optimization of Electric Vehicle Traction Motor Considering Rotor Topology and Manufacturing Uncertainty* in IEEE Transactions on Industrial Electronics, vol. 71, no. 5, pp. 5034-5044

[3] J. Wu, X. Zhu, D. Fan, Z. Xiang, L. Xu and L. Quan, *Robust Optimization Design for Permanent Magnet Machine Considering Magnet Material Uncertainties,* in IEEE Transactions on Magnetics, vol. 58, no. 2, pp. 1-7,

[4] P. Offermann, H. Mac, T. T. Nguyen, S. Clénet, H. De Gersem and K. Hameyer, *Uncertainty Quantification and Sensitivity Analysis in Electrical Machines With Stochastically Varying Machine Parameters*, in IEEE Transactions on Magnetics, vol. 51, no. 3, pp. 1-4,

[5] V. Rafiee and J. Faiz, *Uncertainty Quantification of Permanent Magnet Motor Using Response Surface Surrogate Modeling,* 2020 11th Power Electronics, Drive Systems, and Technologies Conference (PEDSTC), Tehran, Iran, 2020, pp. 1-5

[6] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, V. Makarenkov, and S. Nahavandi *A review of uncertainty quantification in deep learning: Techniques, applications and challenges*

[7] Emiliano Torre and Stefano Marelli and Paul Embrechts and Bruno Sudret, *A general framework for data-driven uncertainty quantification under complex input dependencies using vine copulas*, Probabilistic Engineering Mechanics, 1-16

[8] Xu, C., Zhu, P., Liu, Z., & Tao, W. (2020). *Mapping-Based Hierarchical Sensitivity Analysis for Multilevel Systems With Multidimensional Correlations.* Journal of Mechanical Design, 143(1). https://doi.org/10.1115/1.4047689

[9] Nelsen, R. B. *An introduction to copulas.* springer.

[10] Jondeau, Eric, and Michael Rockinger. *The copula-garch model of conditional dependencies: An international stock market application.* Journal of international money and finance 25.5 (2006): 827-853

[11] B. Renard, M. Lang, *Use of a Gaussian copula for multivariate extreme value analysis: Some case studies in hydrology* Advances in Water Resources 897-912

[12] A. Sklar, *Fonctions de r ´epartition 'a n dimensions et leurs marges*, Publ. Inst. Statist. Univ. Paris 8 (1959) 229–231.

[13] Kendall, Maurice G. *"A new measure of rank correlation." Biometrika 30.1-2 (1938): 81-93.*

[14] van Beek, T. A. *Towards high voltage in coreless linear motors : solutions to thermal and electric-field problems in high force-density applications.* [Phd Thesis 1 (Research TU/e / Graduation TU/e), Electrical Engineering]. Technische Universiteit Eindhoven.

[15] Gysen, B. L. J., Meessen, K. J., Paulides, J. J. H. and Lomonova, E. A. [2010b], *General formulation of the electromagnetic field distribution in machines and devices using Fourier analysis*. IEEE Transactions on Magnetics 46(1), 39–52.

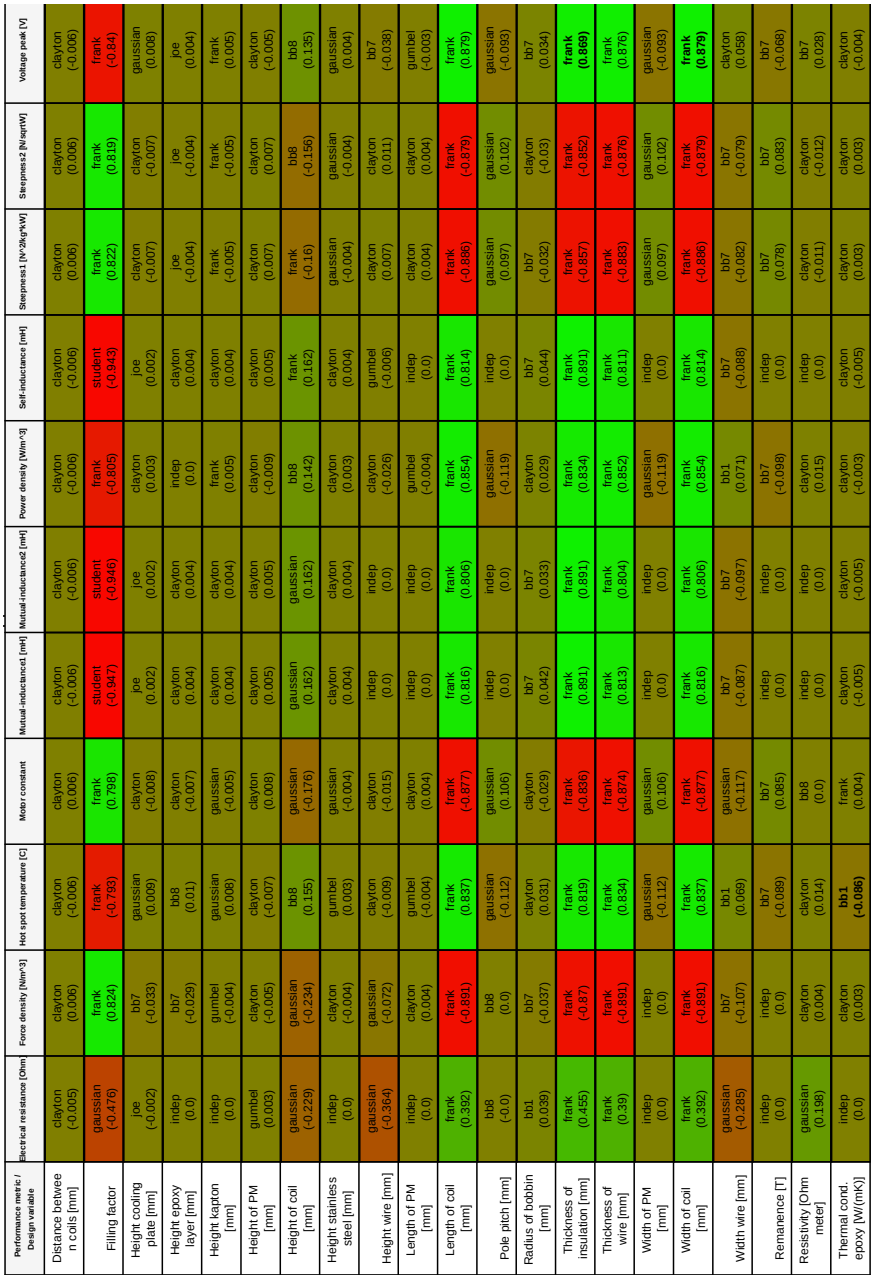
[16] Gysen, B. L. J., Ilhan, E., Meessen, K. J., Paulides, J. J. H. and Lomonova, E. A. [2010a], *Modeling of flux switching permanent magnet machines with Fourier analysis*, IEEE Transactions on Magnetics 46(6), 1499–1502.

[17] G. Fu, M. Curti and E. Lomonova, *Anisotropic 3-D Thermal Modeling for a Racetrack Foil Coil*, 2024 IEEE 21st Biennial Conference on Electromagnetic Field Computation-Extended Papers (CEFC-Extended), Jeju, Korea, Republic of, 2024, pp. 1-4, doi: 10.1109/CEFC65091.2024.10848784.

[18] Nagler, T. and T. Vatter. *pyvinecopulib (v0.7.0).* Zenodo. https://doi.org/10.5281/zenodo.14616803

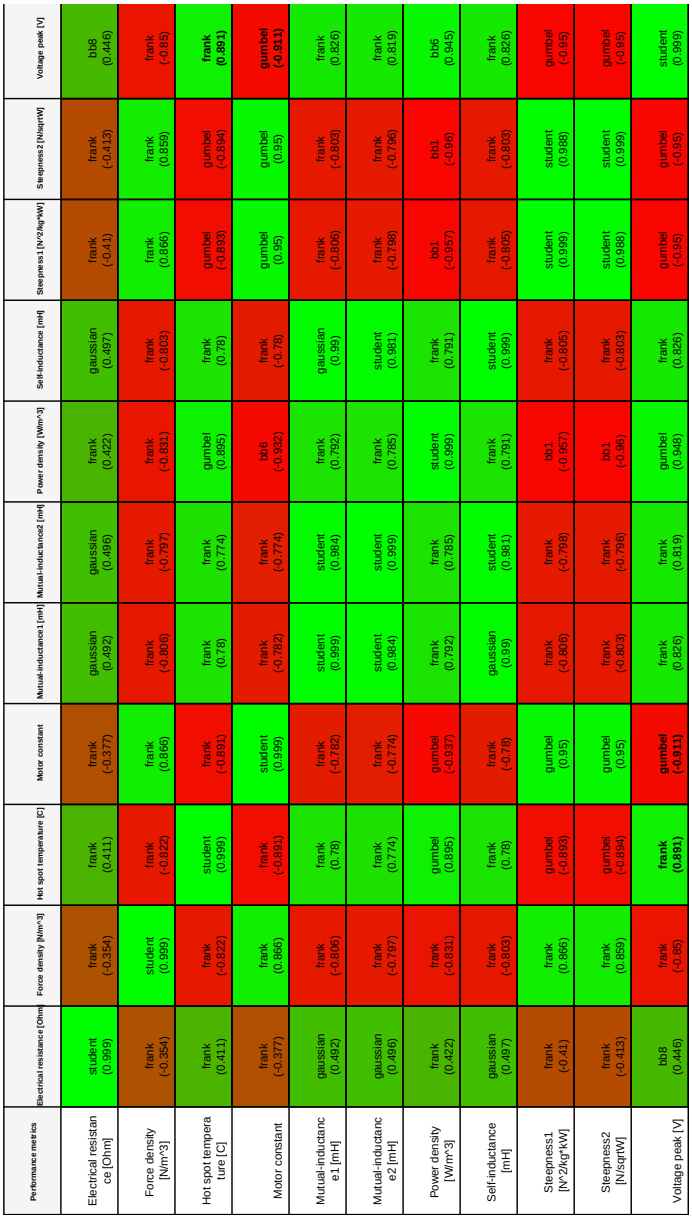
[19] L. V. Hanisch, J. Franzki and M. Henke, *Impact of Insulation and Cooling on Performance due to Reliability-Oriented Design of Electrical Machines,* 2022 24th European Conference on Power Electronics and Applications (EPE'22 ECCE Europe), Hanover, Germany, 2022, pp. 1-7.

APPENDIX



Performance metric / Design variable	Electrical resistance [Ohm]	Force density [N/m^3]	Hot spot temperature [C]	Motor constant	Mutual-inductance1 [mH]	Mutual-inductance2 [mH]	Power density [W/m^3]	Self-inductance [mH]	Steepness1 [N^2/kg*W]	Steepness2 [N/sqrtW]	Voltage peak [V]
Distance between coils [mm]	clayton (-0.005)	clayton (0.006)	clayton (-0.006)	clayton (0.006)	clayton (-0.006)	clayton (-0.006)	clayton (-0.006)	clayton (-0.006)	clayton (0.006)	clayton (0.006)	clayton (-0.006)
Filling factor	gaussian (-0.476)	frank (0.824)	frank (-0.793)	frank (0.798)	student (-0.947)	student (-0.946)	frank (-0.805)	student (-0.943)	frank (0.822)	frank (0.819)	frank (-0.84)
Height cooling plate [mm]	joe (-0.002)	bb7 (-0.033)	gaussian (0.009)	clayton (-0.008)	joe (0.002)	joe (0.002)	clayton (0.003)	joe (0.002)	clayton (-0.007)	clayton (-0.007)	gaussian (0.008)
Height epoxy layer [mm]	indep (0.0)	bb7 (-0.029)	bb8 (0.01)	clayton (-0.007)	clayton (0.004)	clayton (0.004)	indep (0.0)	clayton (0.004)	joe (-0.004)	joe (-0.004)	joe (0.004)
Height kapton [mm]	indep (0.0)	gumbel (-0.004)	gaussian (0.008)	gaussian (-0.005)	clayton (0.004)	clayton (0.004)	frank (0.005)	clayton (0.004)	frank (-0.005)	frank (-0.005)	frank (0.005)
Height of PM [mm]	gumbel (0.003)	clayton (-0.005)	clayton (-0.007)	clayton (0.008)	clayton (0.005)	clayton (0.005)	clayton (-0.009)	clayton (0.005)	clayton (0.007)	clayton (0.007)	clayton (-0.005)
Height of coil [mm]	gaussian (-0.229)	gaussian (-0.234)	bb8 (0.155)	gaussian (-0.176)	gaussian (0.162)	gaussian (0.162)	bb8 (0.142)	frank (0.162)	frank (-0.16)	bb8 (-0.156)	bb8 (0.135)
Height stainless steel [mm]	indep (0.0)	clayton (-0.004)	gumbel (0.003)	gaussian (-0.004)	clayton (0.004)	clayton (0.004)	clayton (0.003)	clayton (0.004)	gaussian (-0.004)	gaussian (-0.004)	gaussian (0.004)
Height wire [mm]	gaussian (-0.364)	gaussian (-0.072)	clayton (-0.009)	clayton (-0.015)	indep (0.0)	indep (0.0)	clayton (-0.026)	gumbel (-0.006)	clayton (0.007)	clayton (0.011)	bb7 (-0.038)
Length of PM [mm]	indep (0.0)	clayton (0.004)	gumbel (-0.004)	clayton (0.004)	indep (0.0)	indep (0.0)	gumbel (-0.004)	indep (0.0)	clayton (0.004)	clayton (0.004)	gumbel (-0.003)
Length of coil [mm]	frank (0.392)	frank (-0.891)	frank (0.837)	frank (-0.877)	frank (0.816)	frank (0.806)	frank (0.854)	frank (0.814)	frank (-0.886)	frank (-0.879)	frank (0.879)
Pole pitch [mm]	bb8 (-0.0)	bb8 (0.0)	gaussian (-0.112)	gaussian (0.106)	indep (0.0)	indep (0.0)	gaussian (-0.119)	indep (0.0)	gaussian (0.097)	gaussian (0.102)	gaussian (-0.093)
Radius of bobbin [mm]	bb1 (0.039)	bb7 (-0.037)	clayton (0.031)	clayton (-0.029)	bb7 (0.042)	bb7 (0.033)	clayton (0.029)	bb7 (0.044)	bb7 (-0.032)	clayton (-0.03)	bb7 (0.034)
Thickness of insulation [mm]	frank (0.455)	frank (-0.87)	frank (0.819)	frank (-0.836)	frank (0.891)	frank (0.891)	frank (0.834)	frank (0.891)	frank (-0.857)	frank (-0.852)	**frank (0.869)**
Thickness of wire [mm]	frank (0.39)	frank (-0.691)	frank (0.834)	frank (-0.874)	frank (0.813)	frank (0.804)	frank (0.852)	frank (0.811)	frank (-0.883)	frank (-0.876)	frank (0.876)
Width of PM [mm]	indep (0.0)	indep (0.0)	gaussian (-0.112)	gaussian (0.106)	indep (0.0)	indep (0.0)	gaussian (-0.119)	indep (0.0)	gaussian (0.097)	gaussian (0.102)	gaussian (-0.093)
Width of coil [mm]	frank (0.392)	frank (-0.891)	frank (0.837)	frank (-0.877)	frank (0.816)	frank (0.806)	frank (0.854)	frank (0.814)	frank (-0.886)	frank (-0.879)	**frank (0.879)**
Width wire [mm]	gaussian (-0.285)	bb7 (-0.107)	bb1 (0.069)	gaussian (-0.117)	bb7 (-0.087)	bb7 (-0.097)	bb1 (0.071)	bb7 (-0.088)	bb7 (-0.082)	bb7 (-0.079)	clayton (0.058)
Remanence [T]	indep (0.0)	indep (0.0)	bb7 (-0.089)	bb7 (0.085)	indep (0.0)	indep (0.0)	bb7 (-0.098)	indep (0.0)	bb7 (0.078)	bb7 (0.083)	bb7 (-0.068)
Resistivity [Ohm meter]	gaussian (0.198)	clayton (0.004)	clayton (0.014)	bb8 (0.0)	indep (0.0)	indep (0.0)	clayton (0.015)	indep (0.0)	clayton (-0.011)	clayton (-0.012)	bb7 (0.028)
Thermal cond. epoxy [W/(mK)]	indep (0.0)	clayton (0.003)	**bb1 (-0.086)**	frank (0.004)	clayton (-0.005)	clayton (-0.005)	clayton (-0.003)	clayton (-0.005)	clayton (0.003)	clayton (0.003)	clayton (-0.004)

Table 2: Bivariate dependencies between design variables and performance metrics, indicating the best-fit copula and rank correlation. Green color signs a positive correlation while red represents a negative correlation. Text in boldface indicates the discussed copulas.



Performance metrics	Electrical resistance [Ohm]	Force density [N/m^3]	Hot spot temperature [C]	Motor constant	Mutual-inductance1 [mH]	Mutual-inductance2 [mH]	Power density [W/m^3]	Self-inductance [mH]	Steepness1 [N^2/kg*kW]	Steepness2 [N/sqrtW]	Voltage peak [V]
Electrical resistance [Ohm]	student (0.999)	frank (-0.354)	frank (0.411)	frank (-0.377)	gaussian (0.492)	gaussian (0.496)	frank (0.422)	gaussian (0.497)	frank (-0.41)	frank (-0.413)	bb8 (0.446)
Force density [N/m^3]	frank (-0.354)	student (0.999)	frank (-0.822)	frank (0.866)	frank (-0.806)	frank (-0.797)	frank (-0.831)	frank (-0.803)	frank (0.866)	frank (0.859)	frank (-0.85)
Hot spot temperature [C]	frank (0.411)	frank (-0.822)	student (0.999)	frank (-0.891)	frank (0.78)	frank (0.774)	gumbel (0.895)	frank (0.78)	gumbel (-0.893)	gumbel (-0.894)	**frank (0.891)**
Motor constant	frank (-0.377)	frank (0.866)	frank (-0.891)	student (0.999)	frank (-0.782)	frank (-0.774)	bb6 (-0.932)	frank (-0.78)	gumbel (0.95)	gumbel (0.95)	**gumbel (-0.911)**
Mutual-inductance1 [mH]	gaussian (0.492)	frank (-0.806)	frank (0.78)	frank (-0.782)	student (0.999)	student (0.984)	frank (0.792)	gaussian (0.99)	frank (-0.806)	frank (-0.803)	frank (0.826)
Mutual-inductance2 [mH]	gaussian (0.496)	frank (-0.797)	frank (0.774)	frank (-0.774)	student (0.984)	student (0.999)	frank (0.785)	student (0.981)	frank (-0.798)	frank (-0.796)	frank (0.819)
Power density [W/m^3]	frank (0.422)	frank (-0.831)	gumbel (0.895)	gumbel (-0.932)	frank (0.792)	frank (0.785)	student (0.999)	frank (0.791)	bb1 (-0.957)	bb1 (-0.96)	bb6 (0.945)
Self-inductance [mH]	gaussian (0.497)	frank (-0.803)	frank (0.78)	frank (-0.78)	gaussian (0.99)	student (0.981)	frank (0.791)	student (0.999)	frank (-0.805)	frank (-0.803)	frank (0.826)
Steepness1 [N^2/kg*kW]	frank (-0.41)	frank (0.866)	gumbel (-0.893)	gumbel (0.95)	frank (-0.806)	frank (-0.798)	bb1 (-0.957)	frank (-0.805)	student (0.999)	student (0.988)	gumbel (-0.95)
Steepness2 [N/sqrtW]	frank (-0.413)	frank (0.859)	gumbel (-0.894)	gumbel (0.95)	frank (-0.803)	frank (-0.796)	bb1 (-0.96)	frank (-0.803)	student (0.988)	student (0.999)	gumbel (-0.95)
Voltage peak [V]	bb8 (0.446)	frank (-0.85)	**frank (0.891)**	**gumbel (-0.911)**	frank (0.826)	frank (0.819)	gumbel (0.948)	frank (0.826)	gumbel (-0.95)	gumbel (-0.95)	student (0.999)

Table 3: Bivariate dependencies between performance metrics, indicating the best-fit copula and rank correlation. Green color signs a positive correlation while red represents a negative correlation. Text in boldface indicates the discussed copulas.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

TRAINING AND DEPLOYMENT OF A DEEP ABSTAINING CLASSIFIER UNDER FEDERATED LEARNING

Cristina Garcia-Cardona[1] **and Jamal Mohd-Yusof**[1]

[1]Los Alamos National Laboratory
Bikini Atoll Rd., SM 30, Los Alamos, NM 87545
e-mail: {cgarciac,jamal}@lanl.gov

Abstract. *In this work we extend the original deep abstaining classifier (DAC) model to be trained under a Federated Learning (FL) setup, with the goal of tackling a wider range of problems including distributed data silos that cannot be freely exchanged between different owner entities, e.g. electronic health records (EHR) from different registries, due to privacy considerations. We use typical FL aggregation strategies for the model parameters and devise new strategies for aggregating the parameters related to the abstention loss function. These are needed to allow the models to achieve their accuracy targets on disparate silos. We implement the DAC+FL training under Flower, a simple, open source federated learning framework. We apply the models to EHR data and compare vs. different baselines. We demonstrate that DAC can be trained under FL, obtaining a performance that is similar to the DAC in consolidated training, sometimes with a minimal drift towards increased abstention rate. We also find that the different aggregation strategies proposed produce similar results. Finally, we indicate how the DAC+FL model can be used for anomaly detection with the possibility of tuning at a local silo level.*

Keywords: Deep abstaining classifier, federated learning, anomaly detection.

1 INTRODUCTION

Federated Learning (FL) was introduced in [1] as an alternative for training DNNs using privacy sensitive data that is distributed across separate devices (*silos*). Instead of storing data in a centralized location, FL learns a shared model via local aggregation of updates, each computed by a participating device (i.e. *client*) and communicated to a global coordinating *server*. This allows decoupling of the model training from the access to the raw data, collectively benefiting from the richness of the individual training datasets, while reducing privacy and security risks.

The basic FL approach usually involves performing local optimization, such as stochastic gradient descent (SGD), on each client and then computing an aggregated model on the server. McMahan et al. [1] demonstrate that FL is robust to unbalanced and non-independently and identically distributed (IID) data, and can operate under limited communication constraints. Further work has considered other open problems in FL [2], as well as general federated machine learning applications [3], and privacy issues [4, 5]. In medicine, in particular, where preserving privacy and reducing the security risks of training with data from electronic health records (EHR) are of utmost importance, FL constitutes an efficient and robust alternative [6, 7, 8] to consolidated training.

In this work we adapt the deep abstaining classifier (DAC) model [9] to be trained in a FL setting and demonstrate its application to EHR data. The DAC identifies data samples that are ambiguous given the input features and abstains from making predictions on them. Instead of discarding these ambiguous samples, they are kept during training to allow the deep neural network (DNN) model to learn useful information regarding the data distribution and associated noise characteristics. The DAC can be constructed on top of any DNN classifier by implementing minimal modifications to the output layer and the training procedure.

The DAC+FL combination brings the flexibility of the abstention framework to the efficient privacy-preserving distributed learning of FL. This has the potential to allow for more extensive applications of the DAC model to settings where privacy preserving considerations are crucial and, at the same time, enable a finer (i.e. local-level) control of the abstention performance by setting different operational targets per client, without significantly impacting overall performance. The main contributions of this work can be summarized as follows:

- Demonstrate that the DAC can be trained under FL setup.
- Show that the performance of the DAC+FL model is similar to the DAC in consolidated training, with a minimal drift towards increased abstention rate per client.
- Demonstrate the potential for the DAC+FL to be deployed as an anomaly detector.

This manuscript is organized as follows. Sec. 2 briefly reviews the DAC and describes how it is adapted to FL. Sec. 3 includes results for numerical experiments using EHR data. We evaluate performance with respect to consolidated (i.e. non-FL) models for homogeneous and heterogeneous datasets and demonstrate how the model can be deployed for anomaly detection with local tuning. Finally, Sec. 4 draws some conclusions about the combined model and suggests directions for future work.

2 DEEP ABSTAINING CLASSIFIER (DAC)

In a traditional classification problem, for a given set of input features $\boldsymbol{x}$, let y be the class to be predicted by the DNN. The probability of the ith class given $\boldsymbol{x}$, denoted $p_i = p_{\boldsymbol{w}}(y = i|\boldsymbol{x})$, can be defined as the i^{th} output of the DNN, which implements the probability model $p_{\boldsymbol{w}}(y = i|\boldsymbol{x})$

(using a softmax function as its final layer) with $\boldsymbol{w}$ denoting a set of weight matrices that parameterize the DNN. For notational brevity, we use p_i to represent $p_{\boldsymbol{w}}(y = i|\boldsymbol{x})$.

The standard cross-entropy training loss for a DNN classifier with k classes can be expressed as:

$$\mathcal{L}_{\text{standard}}(\boldsymbol{x}) = -\sum_{i=1}^{k} t_i \log p_i \tag{1}$$

where $t_i \in \{0, 1\}$ is the target for the current sample i.

The DAC [9] modifies the DNN output layer to include an additional *abstention* class with a corresponding additional $k + 1^{\text{st}}$ output p_{k+1} which is the probability of abstention. This is achieved by training the DAC with the following modified version of the k-class cross-entropy per-sample loss:

$$\mathcal{L}_{\text{DAC}}(\boldsymbol{x}) = (1 - p_{k+1}) \left(-\sum_{i=1}^{k} t_i \log \frac{p_i}{1 - p_{k+1}} \right) + \alpha \log \frac{1}{1 - p_{k+1}} \tag{2}$$

where k is the number of classes excluding the abstention class, t_i is the true label of training data for class i (it is one when i is the true label, zero otherwise), $k + 1$ is the abstention class, p_{k+1} is the probability of the abstention class and $\alpha > 0$ weights the penalty term for abstention.

This loss function behaves like a regular cross-entropy loss on the original classes and adds an additional loss term, scaled by a tuning parameter α, that controls the propensity for abstention. Since the balance between these terms depends on the data and the sources of confusion, the optimal value of α cannot be determined a priori and must be tuned during the training process. Importantly, this parameter is tuned during training to guarantee a target accuracy while minimizing the abstention rate. A very high value of α means a high penalty for abstaining, driving the model towards no abstention. Conversely, a very low value of α may allow the model to abstain on everything.

2.1 Adaptive Abstention

To better understand how the α parameter is tuned, we describe the actual training of a DAC. The training is carried out by optimizing (2) while simultaneously tuning α to satisfy two complementary goals: (i) to achieve a target minimum accuracy while (ii) minimizing the total abstention.

Algorithm 1 Algorithm for Alpha Adaptation in Classification

Require: $r_{\text{max-abs}} \in (0, 1)$, $A_{\text{min-abs}} \in (0, 1)$ and $\alpha^{(k)} > 0$
Ensure: $\alpha^{(k+1)} > 0$
$e_A \leftarrow \max\{A_{\text{min-abs}} - A, 0\}$
$e_r \leftarrow \max\{r - r_{\text{max-abs}}, 0\}$
$a \leftarrow 1 - g_A e_A + g_r e_r$
$\alpha^{(k+1)} \leftarrow a\, \alpha^{(k)}$

Algorithm 1 describes the process for tuning α, a process that is executed after each training epoch. Let the abstention rate be defined as r = # abstaining samples/# total samples, a fraction between 0 and 1. Also, let $r_{\text{max-abs}}$ denote the target maximum abstention rate and let $A_{\text{min-abs}}$ denote the target minimum accuracy. These target values are set at the beginning to reflect performance goals and are kept fixed during training. Note that the updating of a, the α

adaptive multiplier, depends on e_A, the difference between target and achieved minimum accuracy, and on e_r, the difference between target and achieved abstention rate. A positive e_A means that the accuracy is less than the target value set. As shown in the a update in algorithm 1, this nudges a to be less than one, which in turn means a smaller α, making it cheaper to abstain. This increases the abstention rate and potentially increases the accuracy by removing noisy samples from having any effect on the first term of eq. (2). On the other hand, a positive e_r means that the abstention rate is greater than the target set. In the a update, this pushes a to be greater than one, which increments α, making it more expensive to abstain. This should decrease the abstention rate, albeit at the price of a possible decrement in accuracy.

Starting from an initial value of alpha $\alpha^{(0)} > 0$, this adaptive process should achieve a stable α value asymptotically. To avoid large α swings while training, the factor a can be confined to vary in a specified range (e.g. $a \in [0.8, 1.25]$). The adaptation can be made more sensitive to one or the other target setting by using different multiplier factors (i.e. *gains* g_A, g_r) for e_A and e_r when updating a. Note that we use a validation set during training and base the α adaptation on the metrics computed over this set (not the training set). By this adaptive procedure, it has been shown [9] that the resulting DAC trained models are more robust to feature noise, while providing a powerful tool for identifying uncertain predictions at inference time.

The DAC has also shown to be efficient for multi-task problems [9], i.e. for simultaneously solving multiple non-overlapping classification tasks. For the multi-task case, the DAC poses each subclassification problem as a classification plus abstention, each requiring its own α and its own tuning during training. Earlier versions of the DAC were tuned using a combination of accuracy and abstention targets for each task. In the latest versions, used for this work, we have implemented training methods that, for each task, allow targets for either accuracy or abstention alone, in addition to the original mixed targets. Effectively this is equivalent to setting either $g_A = 0$ (pure abstention target) or $g_r = 0$ (pure accuracy target), but with a modified definition of e_A (respectively, e_r) aiming to preserve accuracy (or abstention rate) inside a given interval, i.e., for accuracy, if it is smaller than the lower limit, it triggers a decrease in alpha, while, if accuracy is greater than the upper limit, it triggers an increase in alpha. For this study, we train only for accuracy; specifically, in order to guarantee 97% accuracy, we choose accuracy targets of $97.5\% \pm 0.5\%$. It is possible to set tighter bounds on the accuracy (or abstention) targets, but this can produce significantly longer training times, especially in a multi-task setting. In a multi-task setting, the stopping criteria require that every task satisfy the desired minimum accuracy target (which can be set independently for each task), except when a task is able to exceed the target with zero abstention, in which case that task is considered to have satisfied its target.

2.2 DAC Training Under Federated Learning

For developing a DAC+FL training algorithm, we make use of FL methods typically employed to aggregate the model parameters and only introduce new methodologies to handle the α adaptation under the distributed FL setup.

The normal training of a model under FL is divided in multiple rounds, with each round composed of a number of epochs trained on individual silos, and local updates executed per client after each epoch, as in conventional DNN training. After a round is completed, the parameters of each silo are globally aggregated (typically, weighted mean values, with weights associated to the size of the silo) by the server, and an updated consolidated model is obtained. The server shares the consolidated model with the clients, so that they get back the updated parameters before a new training round starts.

To combine this round-based FL with the DAC training, we allow each round to be executed as in the normal FL training, with the α tuning proceeding independently in each client as described in Algorithm 1. What we need to consider then is how the client-wise tuned α's should be synchronized by the server to produce globally updated α parameters that will be used to optimize the abstention loss function (2) in the following round. For this synchronization we devise and compare two different strategies.

Local: This strategy allows the clients to proceed with the optimization using their *'local'* α parameter values without any global synchronization by the server, i.e. the α's evolve independently in each client, and the only global synchronization comes via the updates to the DNN model parameters performed by the server after each round. Hence, each client starts the next round with aggregated network weights and with the *'local'* α's saved from the previous round.

Shared: This strategy allows the server to aggregate both α parameter values and model weights after each round, following the same policy (e.g. weighted average across clients). This yields a model where each client starts the next round with aggregated network weights and *'shared'* α's across the different clients.

These strategies aim to capture the intuition that if the data silos are similar, then the shared strategy should produce the fastest convergence, while if the silos are dissimilar (in size, level or type of noise in the features, etc.) then allowing each one to adapt their α's independently may be the most robust and stable strategy.

We implement these two strategies for the α aggregation in the training of the FL+DAC model using the Flower Framework [10], a simple framework to bring existing machine learning workloads into a federated setting. This results in a scalable, well-tested, and open-source infrastructure.

3 NUMERICAL EXPERIMENTS

This section presents some of the results of training the DAC under a FL setting. We compare performance with baselines including no-abstention and non-federated learning DNN classifier models. First we contrast the performance of the DAC trained with the regular procedure (a.k.a. consolidated learning) vs. the performance based on a FL implementation, and demonstrate that the FL setup does not affect the main characteristics of the DAC. Then, we compare the performance of the two proposed alpha aggregation approaches, namely local and shared, both in experiments with homogeneous and heterogeneous data silos. We finalize by illustrating how the DAC+FL can be used for silo-wise anomaly detection.

3.1 Dataset Description

In this work, we use text pathology reports from the US National Cancer Institute SEER (Surveillance, Epidemiology, and End Results) registries [11]. Each case of cancer (individual tumor), given by the case ID, is identified by a combination of a patient ID and a tumor ID. Ground truth for each case of cancer is obtained from the manually abstracted and consolidated records in the cancer registries. There may be multiple reports for each case, each of which are identified by a combination of a patient ID, a tumor ID, and a report ID, and has assigned values for five tasks: behavior (4 classes), histological type ($\sim$ 550 classes), laterality (7 classes), primary site ($\sim$ 70 classes) and primary subsite ($\sim$ 300 classes). In addition to the disparity in number of classes, there is a wide disparity in prevalence across classes. The ground truth is

consolidated for each case ID (i.e., each individual tumor), meaning all the reports pertaining to a particular tumor have the same ground truth regardless of the content of the text pathology report. Previous work to develop models for automated annotation of cancer pathology reports [12, 13, 14, 15] demonstrated the value of DAC methods in a consolidated training setting. For the purposes of this work, we focus on the mechanics of enabling DAC training in a FL setting rather than the details of the dataset or the specific FL aggregation schemes.

3.2 Homogeneous Dataset Experiments

To assess the performance of DAC under consolidated and FL training modalities we selected the registry of Louisiana (LA), which is composed of 482,914 samples, divided into static train (338,249 samples), validation (72,391 samples) and test (72,274 samples) partitions. We used the registry to train one DAC model with a target accuracy of $97,5\% \pm 0.5\%$ for each task. We also simulated a federated environment by using two silos with the same dataset. By mimicking a FL environment using the same registry per silo and target configuration we build an homogeneous dataset case where the results should be comparable with the consolidated training. Furthermore, by making homogeneous silos, we are more certain that any differences detected in the results are probably introduced by the different training modalities evaluated, and not caused by dataset issues.

We configured the FL for 4 rounds of training with a maximum of 25 epochs per round. We trained DAC+FL models using the proposed α aggregation strategies (i.e., local and shared). All models used analogous configurations, with the same rounds $\times$ max. epochs and with the same abstention and accuracy targets. A maximum of 100 epochs was set for the non-FL training. We repeated the training for each model 5 times using different initializations.

Figure 1 displays the evolution of α for one instantiation of the training of the DAC+FL model. The left column shows the α evolution for local aggregation in each of the two clients. The righ column shows corresponding results for the shared aggregation. The 4 rounds are concatenated and the transition between rounds is indicated by vertical dotted lines. It can be seen that α's evolve smoothly in both aggregation strategies. We also observe that the shared strategy seems to require a few more iterations to converge, but, in the end, both strategies converge to very similar α values. Figures 2– 3 show analogous results for the evolution of the abstention rate and the accuracy in the validation set. Recall that is the trade-off between abstention rate and accuracy *in the validation set* which allows the dynamical tuning of the α values. These figures demonstrate that both strategies are able to satisfy the target accuracy, and that performance is similar in both strategies, albeit with a slightly slower convergence for the shared strategy as already mentioned.

To get a sense of the dispersion with respect to different training initializations, we report in Table 1 the number of training epochs used per round, per each repetition of the models evaluated. It can be seen that the base DNN classifier, that does not use abstention or FL ('No Abs, No FL' row in table), requires an average of 32 epochs to converge[1]. The same DNN classifier when trained in a FL setup ('No Abs' row in table) requires an average of about 41 epochs[2] (same convergence definition as[1]), suggesting that FL tends to yield marginally slower convergence, which is usually justified by the noise that is injected to maintain the privacy

[1]Convergence here is defined as achieving the target minimum accuracy per task, or early stopping if no improvement is obtained after a patience of 5 epochs.

[2]The direct estimation of about 58 epochs on average to converge would include some extra patience epochs that are triggered due to the length of the rounds in FL. The average of 41 epochs for convergence is obtained after discarding the extra patience epochs, i.e. 5 patience epochs $\times$ 3 rounds $\simeq$ 15 extra epochs.

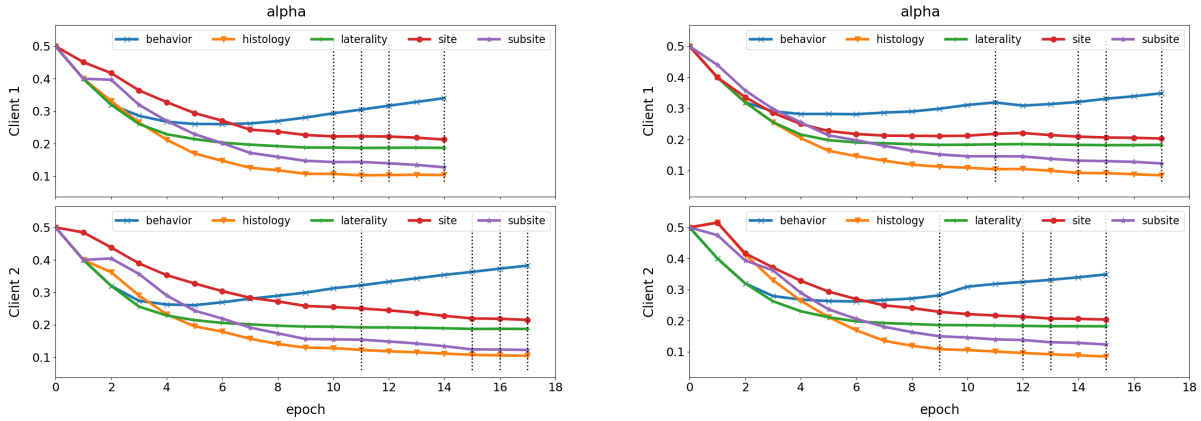


Figure 1: Evolution of alpha: one instantiation of DAC+FL training for for each of the 5 tasks (behavior, histology, laterality, site, subsite). Rounds are concatenated, vertical dotted lines indicate the different rounds. Left: local alpha aggregation. Right: shared alpha aggregation.

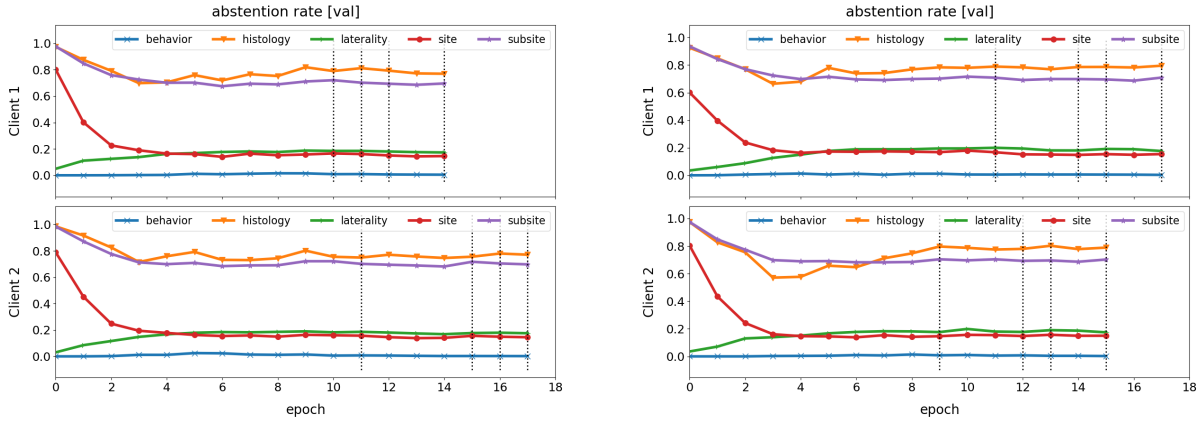


Figure 2: Evolution of abstention rate in validation set: one instantiation of DAC+FL training for each of the 5 tasks (behavior, histology, laterality, site, subsite). Left: local alpha aggregation. Right: shared alpha aggregation.

of the silo. Comparing with the DAC baseline ('No FL' row in table) which uses abstention training but no FL, it can be seen that convergence is much faster, requiring about 12 epochs to converge[3]. This illustrates the influence of the abstention loss function (2) in regularizing the training and in helping to identify the more noisy samples. The proposed DAC+FL models exhibit a convergence (same convergence definition as[3]) that is closer to the DAC baseline, with about 17 epochs on average for both local and shared strategies. Even though in one realization it seems that 'shared' was slower than 'local', both strategies have the same performance on average. This is somewhat expected, due to the fact that the silos are basically copies of each other in the homogeneous experiments, so there are no differences in data noise to prompt different local adjustments in α's or abstention rates.

Figures 4, 5 and 6 plot the α's, accuracies and abstention rates, respectively, obtained at the end of training for the validation set. It is evident that any of the abstention-based methods is able to achieve the set minimum accuracy of 97%. This highlights the benefit of using abstention

[3]Convergence here is defined as achieving the target minimum accuracy per task, without surpassing the maximum target abstention rate per task, or early stopping if no improvement is obtained after a patience of 5 epochs.

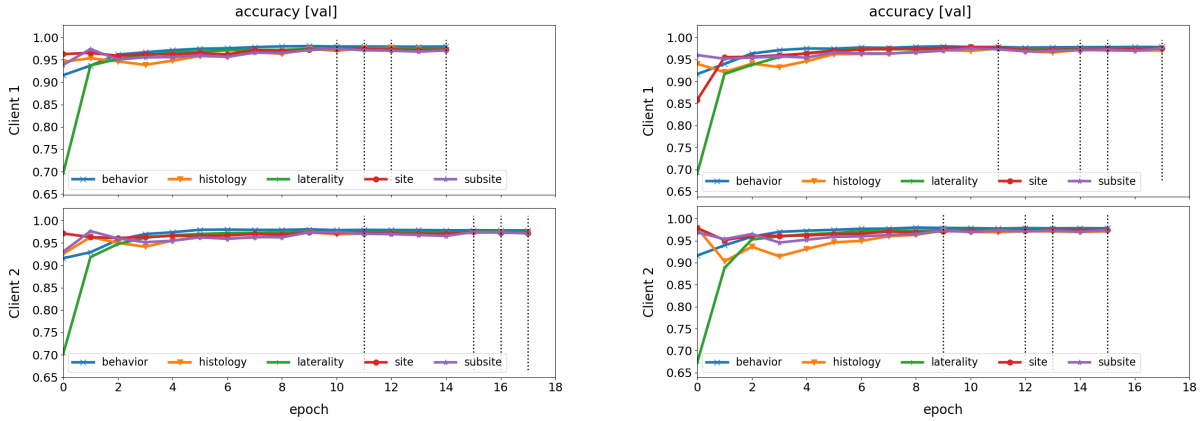


Figure 3: Evolution of accuracy in validation set: one instantiation of DAC+FL training for each of the 5 tasks (behavior, histology, laterality, site, subsite). Rounds are concatenated, vertical dotted lines indicate the different rounds. Left: local alpha aggregation. Right: shared alpha aggregation.

Method	Repetition 1	Repetition 2	Repetition 3	Repetition 4	Repetition 5
DAC + FL Local	11, 1, 1, 2 12, 4, 1, 1	10, 3, 1, 1 11, 1, 1, 1	13, 6, 1, 1 15, 2, 2, 2	14, 1, 1, 1 12, 2, 2, 1	11, 2, 2, 1 11, 1, 2, 1
DAC + FL Shared	12, 3, 1, 2 10, 3, 1, 2	14, 1, 1, 1 13, 1, 3, 1	12, 1, 1, 1 12, 2, 2, 2	12, 1, 2, 2 14, 2, 2, 1	12, 2, 1, 1 12, 2, 2, 1
No FL	10	11	12	13	12
No Abs	25, 16, 9, 6 25, 16, 9, 7	25, 19, 6, 7 25, 19, 6, 10	25, 17, 9, 6 25, 17, 9, 6	25, 18, 6, 9 25, 18, 9, 9	25, 17, 7, 9 25, 17, 7, 7
No Abs, No FL	35	33	34	28	32

Table 1: Number of epochs in training per repetition. For FL-trained models this is reported by client and by round.

for detecting unreliable samples. It is also clear that both local and shared strategies converge to similar α values and accuracies. The major dispersions in accuracy observed are correlated with the tasks that exhibit lower abstention rates. This can be explained as follows. The abstention in the behavior task is zero, so the α parameter is really meaningless since it is being applied to a zero abstention term. Hence, the *apparent* differences in α between local and shared strategies for the behavior task are irrelevant. Laterality and site tasks show only minor divergences and low abstention rates. The histology and subsite tasks are much more tightly correlated in α's, reflecting the increased difficulty posed by tasks with hundreds of classes, high class imbalance, and more ambiguity ([14]). This also results in higher abstention levels required to achieve the target minimum accuracy of 97%.

To finalize the set of experiments with homogeneous silos, we present in Tables 2 and 3 the accuracy and abstention rate, respectively, evaluated in the testing set and averaged over all the trained models in each modality. These results show that the results in the testing set are consistent with the ones obtained in the validation set. Overall, both local and shared DAC+FL strategies match the DAC baseline, corroborating the statement that DAC can be trained under FL setup while achieving a performance that is very similar to the DAC in consolidated training for the homogeneous dataset, as expected.

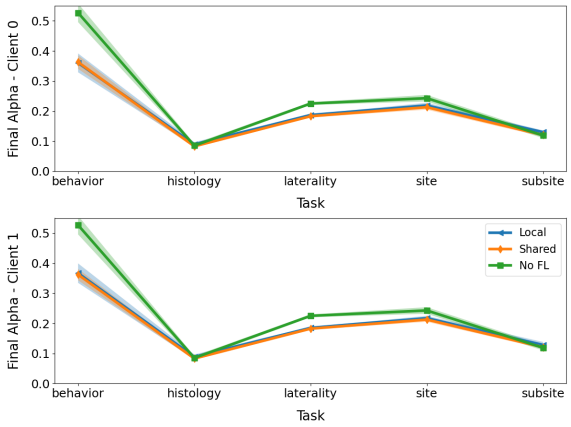


Figure 4: Values of alphas at the end of training for the compared modalities: DAC+FL with local alpha aggregation, DAC+FL with shared alpha aggregation and DAC without FL.

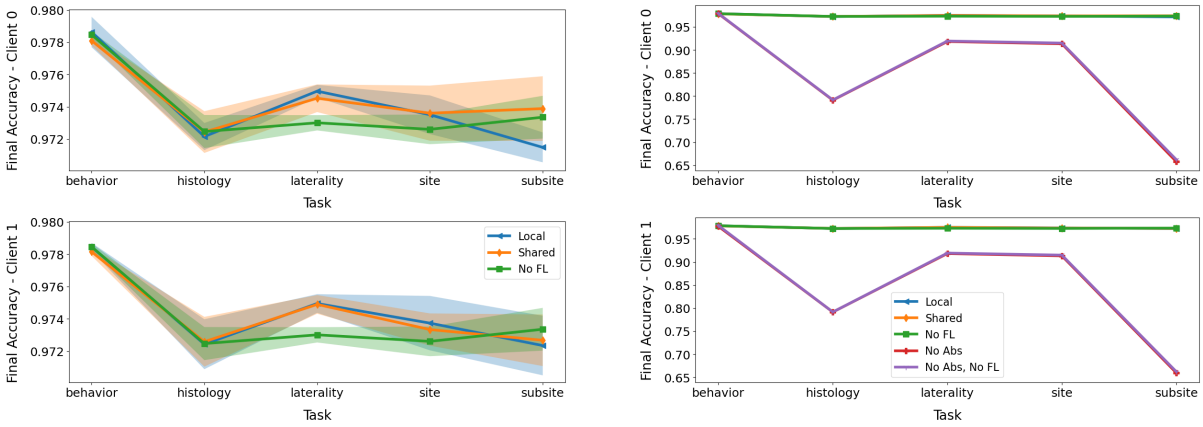


Figure 5: Accuracy at the end of training for the compared modalities. Plots correspond to the mean on 5 realizations and shaded region shows the standard deviation. **Left:** DAC+FL with local alpha aggregation, DAC+FL with shared alpha aggregation and DAC without FL. **Right:** includes also the plots for FL without abstention and for baseline without abstention or FL.

3.3 Heterogeneous Dataset Experiments

Having confirmed that homogeneous FL training essentially converges to the consolidated training result, we now address the more meaningful case of heterogeneous FL training. For these tests we use a combined dataset of Louisiana (LA) and Kentucky (KY). For the FL training, Client 0 uses the LA data and Client 1 the KY data. The KY dataset is comparable in size to LA (537,647 total) with a similar partitioning into train, validation and test sets.

We perform the same experiments as in the previous section, to confirm that the 'local' and 'shared' aggregation modalities for the α coefficients converge similarly and are comparable to the results for the consolidated training on the combined LAKY dataset.

Figure 7 shows the distribution of final α values for each task, across five training runs each for the 'local' and 'shared' FL training modes, and the same dataset trained as a single consolidated corpus. As with the homogeneous training, the values for the histology and subsite tasks are very tightly clustered across all modalities, reflecting the stringency required to obtain the desired accuracy bounds. Conversely, the values for behavior are quite disparate, since the

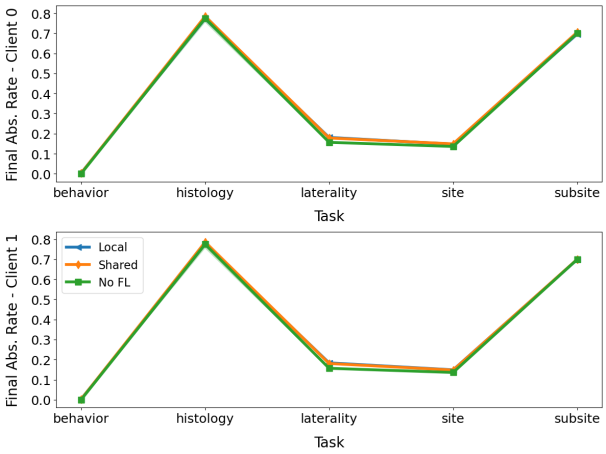


Figure 6: Abstention rate at the end of training for the compared modalities: DAC+FL with local alpha aggregation, DAC+FL with shared alpha aggregation and DAC without FL.

Task	DAC	DAC + FL Local	DAC + FL Shared
Behavior	0.9794±0.0001	0.9792±0.0009	0.9789±0.0003
		0.9791±0.0004	0.9790±0.0002
Histology	0.9750±0.0015	0.9736±0.0011	0.9751±0.0007
		0.9739±0.0019	0.9754±0.0013
Laterality	0.9737±0.0003	0.9756±0.0006	0.9756±0.0006
		0.9759±0.0007	0.9758±0.0006
Site	0.9756±0.0009	0.9763±0.0008	0.9764±0.0014
		0.9766±0.0015	0.9762±0.0007
Subsite	0.9766±0.0011	0.9738±0.0013	0.9757±0.0023
		0.9746±0.0020	0.9744±0.0017

Table 2: Mean accuracy ± standard deviation in test set per client and per task over the trained models.

accuracy target can be met with minimal or zero abstention (see Figures 8 and 9).

There is a visible difference in the final α values for laterality and site between client 0 and client 1, consistent across all training runs, indicating that the disparate silos require a slightly different tuning to achieve the same nominal targets. This is also visible in Figure 9 where clear differences in abstention rate are visible between clients. Notably the abstention rates for the FL case are higher, due to both noise introduced in the FL workflow and the need to satisfy accuracy targets on independent subsets of the data.

Overall, we demonstrate that the FL trained models converge to the consolidated training case, albeit with some slight variation between clients, primarily due to the data disparities and consequent divergence in the final parameters.

3.4 Anomaly Detection

A previously observed feature of the DAC is the ability to signal the prevalence of anomalous samples through changes in the abstention rate. As evidenced in the heterogeneous training example, the actual abstention rate for nominally identical models on datasets with differing composition will vary. In this section we briefly demonstrate the variability of abstention rate in

Task	DAC	DAC + FL Local	DAC + FL Shared
Behavior	0.0003±0.0002	0.0033±0.0016	0.0027±0.0013
		0.0029±0.0014	0.0025±0.0010
Histology	0.7732±0.0202	0.7708±0.0109	0.7822±0.0125
		0.7728±0.0105	0.7825±0.0143
Laterality	0.1567±0.0041	0.1808±0.0074	0.1782±0.0064
		0.1833±0.0075	0.1803±0.0057
Site	0.1329±0.0036	0.1454±0.0077	0.1464±0.0060
		0.1470±0.0081	0.1452±0.0043
Subsite	0.6975±0.0026	0.6949±0.0059	0.7036±0.0087
		0.6971±0.0059	0.6984±0.0065

Table 3: Mean abstention rate ± standard deviation in test set per client and per task over the trained models.

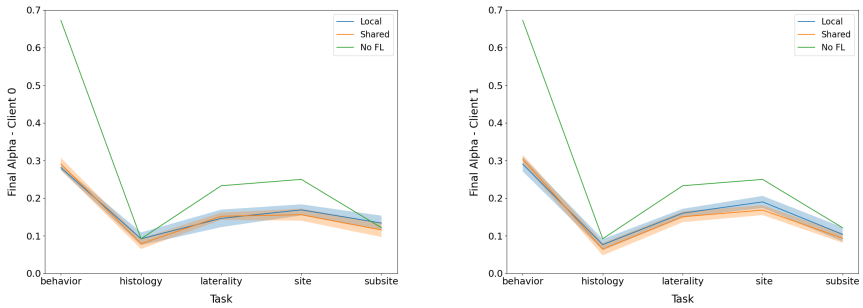


Figure 7: Values of alphas at the end of heterogeneous training for the compared modalities: DAC+FL with local alpha aggregation, DAC+FL with shared alpha aggregation, and consolidated training. Left is client 0 (LA), right is client 1 (KY). Alpha values converge to similar values across multiple tests, with consistent differences between client 0 and client 1.

two scenarios, cross registry testing and variations over time. This is preliminary study which will be more fully explored in upcoming work.

3.4.1 Cross-Registry Performance

For this test, we use the five models from the heterogeneous training, trained on LA+KY, and test the performance on the 'test' partition of five separate registries. In addition to the LA and KY datasets, we include registries from New Mexico (NM), Seattle (SE) and Utah (UT). These registries differ substantially in total size, as well potentially having different compositions compared to the LA, KY registries, including differing class imbalance, quality of annotation (ground truth) and a variety of other factors. The full analysis of the variations between registries and the reasons for abstention are beyond the scope of this paper.

Figure 10 shows the summary of the accuracy and abstention variation across tasks for each of the registries. For accuracy, the 97% is maintained or exceeded for all tasks except behavior, since that task has effectively no abstention pattern learned and simply behaves as a non-abstaining classifier in the presence of noise. The variation in abstention is obscured due to the large range of values in the plot so we display an alternate view in Figure 11, showing the variation of abstention across registries for each task. We exclude behavior since this remains

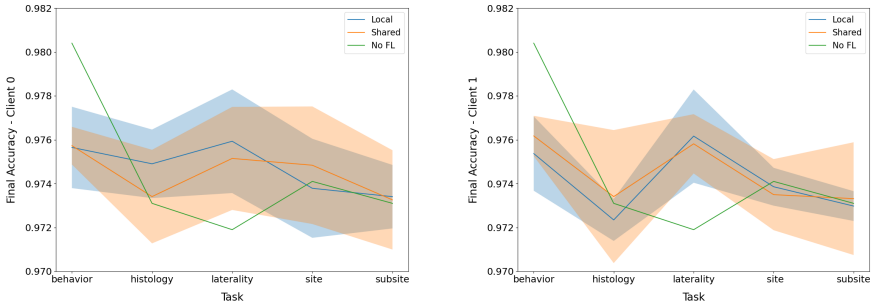


Figure 8: Accuracy at the end of training for the compared modalities: DAC+FL with local alpha aggregation, DAC+FL with shared alpha aggregation and DAC without FL. Left is client 0 (LA), right is client 1 (KY)

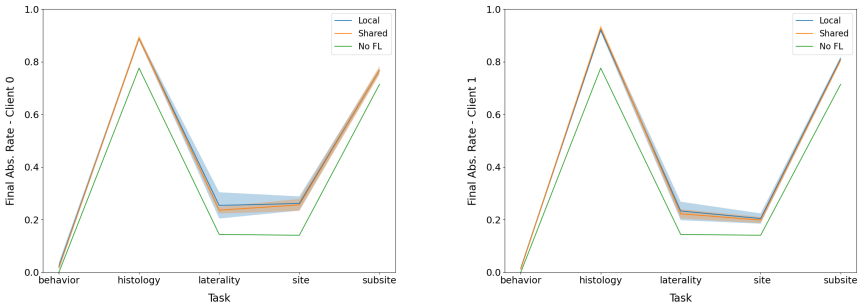


Figure 9: Abstention rate at the end of training for the compared modalities: DAC+FL with local alpha aggregation, DAC+FL with shared alpha aggregation and DAC without FL. Left is client 0 (LA), right is client 1 (KY)

effectively zero, as discussed above. Abstention rates vary across registries, but the trend is different for each task, reflecting the complexity of the dynamics and differences in number of classes, noise level and base abstention rates for each. Nevertheless, the ability of the DAC to signal variations in quality and the potential for abstention rates to vary, both positively and negatively, is demonstrated.

3.4.2 Time evolution inference tests

An alternate demonstration of the anomaly detection feature is to test the ability to discern changes in data over time. For this, we train the models on LA,KY data in an FL setting, but only using data up to and including the 2012 calendar year. The remaining LA,KY data, from 2013 to 2022, is partitioned into yearly increments and only the 'test' partition from each year is used.

Figures 12 and 13 show the abstention and accuracy performance for each client. Note that, as before, the trend for each task differs over time, but some consistent patterns emerge. Both clients display similar differences between LA and KY test data, despite being trained primarily on one or the other. For example, there is a clear separation between subsite accuracy rates, with LA (blue) always being more accurate than KY (orange), indicating a systematic difference in data quality between them. In contrast, the abstention rates for the same task vary more and converge over time.

The patterns learned by the DAC can be complex, and require an understanding of the details

Method	Repetition 1	Repetition 2	Repetition 3	Repetition 4	Repetition 5
DAC + FL Local	19, 1, 1, 1	21, 2, 6, 1	22, 9, 1, 2	24, 2, 2, 2	25, 9, 1, 1
	17, 2, 1, 1	17, 4, 1, 1	24, 2, 1, 1	16, 3, 3, 6	17, 14, 1, 1
DAC + FL Shared	25, 8, 2, 2	25, 1, 1, 1	23, 2, 2, 2	25, 10, 1, 3	23, 8, 2, 2
	21, 7, 2, 4	21, 2, 1, 7	17, 3, 1, 1	17, 3, 6, 2	17, 2, 7, 1

Table 4: Number of epochs in heterogeneous training per repetition. For each modality, top row is client 0 (LA) and bottom is client 1 (KY), epochs are reported by round. Note that in heterogeneous training the last round may require multiple epochs to converge, so that the final model state, including both α and model weights, may be slightly different for each silo.

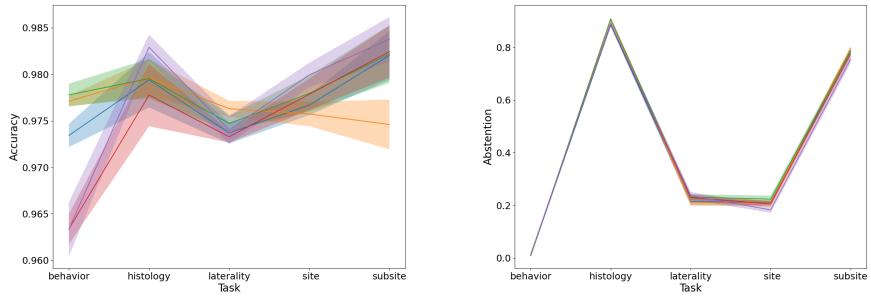


Figure 10: Performance summary, abstention and accuracy performance across registries. The models are trained on LA+KY data and tested on separate 'test' partitions from LA, KY, NM, SE, UT registries.

of the registry data. Over time, the cancer coding standards undergo periodic updates, and the quality of diagnostic tools may improve. Additional changes may be due to changes in population demographics within the catchment regions for the registries, due to a variety of factors. Again, the exact reasons for the variation are beyond the scope of this paper, where we only seek to demonstrate the potential for the DAC to signal (and separate, via the abstention flag) the presence of anomalous samples in the data.

4 CONCLUSION

We have demonstrated the ability to implement the accuracy-targeted DAC in a FL setting, and tested two different aggregation modes for the α coefficient which are an essential component of the models. The resulting FL trained models are shown to converge to the model obtained via consolidated training, with small differences due to the noise introduced in the FL process (to preserve privacy) and the need to enforce the accuracy target on independent subsets of the data. We then demonstrate the ability of these models to signal changes in data quality, via testing on data from alternate registries and from variations over time from the same registry. This provides a jumping off point for further work to characterize and improve the sensitivity of the DAC as an anomaly detecter for deployment in real-world distributed data collection environments.

5 Acknowledgments

The authors would like to thank Chris Stanley, John Gounley and Heidi Hanson from ORNL for their guidance on federated learning, the Flower Framework in particular, assistance with

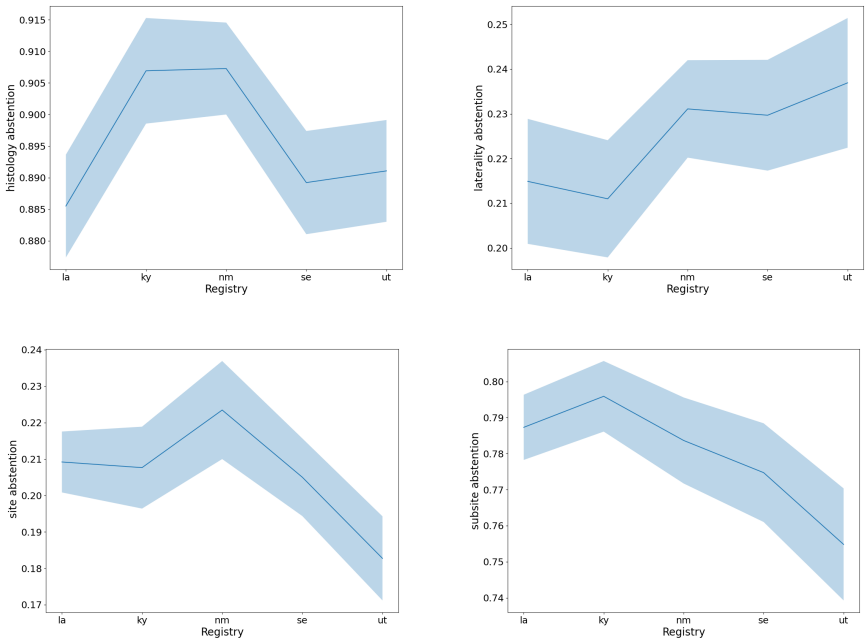


Figure 11: Abstention variation across registries for FL models trained on LA+KY. The behavior task is excluded since abstention remains effectively zero and the model has not learned an abstention pattern for this task.

data preparation and for valuable discussions.

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing under Award Number DE-SC-ERKJ422.

The authors would like to acknowledge the contribution to this study from other staff in the participating central cancer registries. These registries are supported by the National Cancer Institute's Surveillance, Epidemiology, and End Results (SEER) Program, the Centers for Disease Control and Prevention's National Program of Cancer Registries (NPCR), and/or state agencies, universities, and cancer centers.

The participating central cancer registries include the following:

- Fred Hutchinson Cancer Center working under contract numbers
  SEER: HHSN2612018000041
- Kentucky working under contract numbers
  SEER: HHSN261201800013I/HHSN26100001 and NPCR: NU58D007144
- Louisiana working under contract numbers
  SEER: HHSN261201800007I/HHSN26100002 and NPCR: NU58DP0063.
- New Mexico working under contract numbers
  SEER: HHSN2616018000141
- Utah working under contract numbers
  SEER: HHSN2612018000161 and NPCR: NU58DP007131

REFERENCES

[1] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017.

[2] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, R. G. L. D'Oliveira, H. Eichner, S. E. Rouayheb, D. Evans, and J. Gardner, "Advances and open problems in federated learning," in *Foundations and Trends in Machine Learning*, vol. 14, 2021.

[3] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, p. 1–19, 2019.

[4] C. Dwork, A. Roth, and et al., "The algorithmic foundations of differential privacy," *Foundations and Trends in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211 – 407, 2014.

[5] P. Bellavista, L. Foschini, and A. Mora, "Decentralised learning in federated deployment environments: A system-level survey," in *ACM Computing Surveys*, vol. 54, 2021.

[6] M. J. Sheller, B. Edwards, G. A. Reina, J. Martin, S. Pati, A. Kotrotsou, M. Milchenko, W. Xu, D. Marcus, R. R. Colen, and S. Bakas, "Federated learning in medicine: Facilitating multi-institutional collaborations without sharing patient data," *Scientific Reports*, vol. 10, no. 1, pp. 1 – 12, 2020.

[7] S. Warnat-Herresthal, H. Schultze, K. L. Shastry, S. Manamohan, S. Mukherjee, V. Garg, R. Sarveswara, K. Händler, P. Pickker, and A. N. Ahmad, "Swarm learning for decentralized and confidential clinical machine learning," *Nature*, vol. 594, no. 7862, pp. 265–270, 2021.

[8] G. Kaissis, A. Ziller, J. Passerat-Palmbach, T. Ryffel, D. Usynin, A. Trask, I. Lima, J. Mancuso, F. Jungmann, and M.-M. Steinborn, "End-to-end privacy preserving deep learning on multi-institutional medical imaging," *Nature Machine Intelligence*, vol. 3, no. 6, pp. 473 – 484, 2021.

[9] S. Thulasidasan, T. Bhattacharya, J. Bilmes, G. Chennupati, and J. Mohd-Yusof, "Combating label noise in deep learning using abstention," in *Proceedings International Conference on Machine Learning*, 2019.

[10] Flower-Labs-GmbH, "Flower framework." [Online]. Available: https://flower.ai/docs/framework/index.html

[11] "National Cancer Institute - Survellillance, Epidemiology, and End Results Program," https://seer.cancer.gov/.

[12] M. Alawad, S. Gao, J. X. Qiu, H. J. Yoon, J. Blair Christian, L. Penberthy, B. Mumphrey, X.-C. Wu, L. Coyle, and G. Tourassi, "Automatic extraction of cancer registry reportable information from free-text pathology reports using multitask convolutional neural networks," *Journal of the American Medical Informatics Association*, vol. 27, no. 1, pp. 89–98, 11 2019. [Online]. Available: https://doi.org/10.1093/jamia/ocz153

[13] S. Gao, M. Young, J. Qiu, H.-J. Yoon, J. Christian, P. Fearn, G. Tourassi, and A. Ramanthan, "Hierarchical attention networks for information extraction from cancer pathology reports," *Journal of the American Medical Informatics Association*, vol. 25, no. 3, pp. 321–330, 2018.

[14] S. Dhaubhadel, J. Mohd-Yusof, K. Ganguly, G. Chennupati, S. Thulasidasan, N. W. Hengartner, B. J. Mumphrey, E. B. Durbin, J. A. Doherty, M. Lemieux, N. Schaefferkoetter, G. Tourassi, L. Coyle, L. Penberthy, B. H. McMahon, and T. Bhattacharya, "Why I'm not answering: Understanding determinants of classification of an abstaining classifier for cancer pathology reports," *Arxiv*, 2022.

[15] A. Peluso, I. Danciu, H.-J. Yoon, J. Mohd-Yusof, T. Bhattacharya, A. Spannaus, N. Schaefferkoetter, E. Durbin, X.-C. Wu, A. Stroup, J. Doherty, S. Schwartz, C. Wiggins, L. Coyle, L. Penberthy, G. Tourassi, and S. Gao, "Deep learning uncertainty quantification for clinical text classification," *Journal of Biomedical Informatics*, vol. 149, p. 104576, 2024.

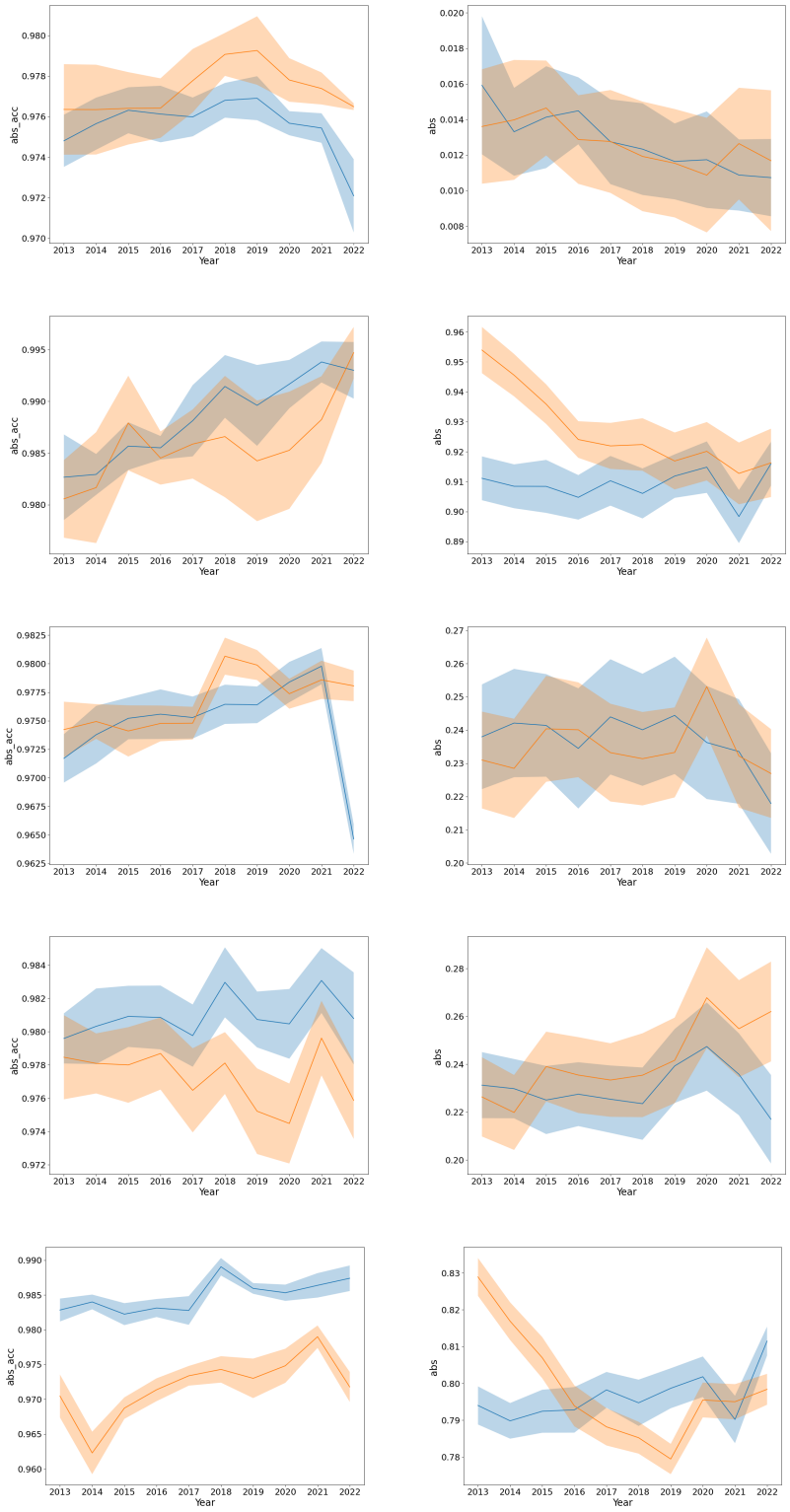


Figure 12: Inference performance of a DAC trained on pre-2012 data, tested on individual years from 2013 to 2022. Client 0 (trained on LA data in a heterogeneous LA+KY FL setting), tested on LA and KY data from 2013-2022. Tasks are (top to bottom) behavior, histology, laterality, site, subsite. Left column is accuracy, right column is abstention rate.

Figure 13: Inference performance of a DAC trained on pre-2012 data, tested on individual years from 2013 to 2022. Client 1 (trained on KY data in a heterogeneous LA+KY FL setting), tested on LA and KY data from 2013-2022. Tasks are (top to bottom) behavior, histology, laterality, site, subsite. Left column is accuracy, right column is abstention rate.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–18 June 2025

ADDRESSING EPISTEMIC UNCERTAINTY IN STATISTICAL TOLERANCE ANALYSIS: A NON-DETERMINISTIC APPROACH

Sonia C. Garcia[1], Pierre Beaurepaire[1], Nicolas Gayton[1], and Chahrazed Amrane[2]

[1]Université Clermont Auvergne, Clermont Auvergne INP, CNRS, Institut Pascal F-63000
Clermont-Ferrand, France
e-mail: {sonia.garcia_gomez,pierre.beaurepaire,nicolas.gayton}@sigma-clermont.fr

[2]Phimeca Engineering SA, Phimeca Engineering SA, France
e-mail: amrane@phimeca.com

Abstract. *Deviations in component dimensions are unavoidable due to manufacturing and measurement imperfections. Tolerance analysis evaluates how these variations impact the functionality of a mechanism by ensuring the stack-up of deviations remains within acceptable limits. The tolerance zone defines the permissible range of variations concerning the nominal dimension of the components, establishing boundaries for their acceptable deviations.*
In statistical tolerance analysis, the variability of components is represented using statistical distributions that model the spread and likelihood of component dimensions within a defined tolerance zone. The choice of distribution is crucial, as there may be an infinite number of probabilistic distributions that could fit the tolerance information. Traditional statistical tolerance methods often rely on precise probabilistic models that do not account for epistemic uncertainty due to a lack of knowledge about the exact distribution of deviations. Making arbitrary choices concerning the distribution of the deviations can affect the accuracy and the validity of the tolerance analysis.
The APTA method, developed in [5], is applicable to scenarios where variations in the mean and/or standard deviation need to be considered. This paper aims to determine the worst-case non-conformity rate by accounting for epistemic uncertainty to build the probability box (P-Box) that represents all the possible distributions of the cumulative stack-up of deviations for mechanical systems.
The method presented in this paper is applied to specific study cases, demonstrating its ability to provide a comprehensive understanding of the influence of epistemic uncertainty on tolerance analysis, and its capacity to provide a more general representation of the cumulative stack-up of deviations for practical engineering applications.

Keywords: Statistical tolerance analysis, non-conformity rate, epistemic uncertainty, imprecise probability.

1 INTRODUCTION

Dimensional and geometric deviations introduced during manufacturing are inevitable and directly impact the functional performance of assembled products. To control unintended part deviations, the designer must specify and allocate tolerances to guarantee that specific quality requirements are met [9]. Nevertheless, defining tolerances is a complex task because the assigned values not only affect the functionality and manufacturing cost of the design, but they are often interrelated and contribute to a given resultant tolerance [15].

The problem of tolerance design is traditionally approached in two different ways: tolerance analysis and tolerance synthesis [10]. Tolerance synthesis is often viewed as an optimization problem where the goal is to determine the set of tolerances that either maximizes or minimizes a cost, quality, or a combined cost-quality function. In contrast, tolerance analysis focuses on evaluating the compliance of a given set of tolerances with the specified functional requirements of the products [11].

Tolerance analysis, and tolerance design in general, can be classified into two main approaches: statistical tolerance analysis and worst-case analysis [8]. The worst-case analysis assumes that all dimensional deviations will occur at their maximum allowable limits, ensuring that even under the worst combination of deviations, the assembled product will still meet its functional requirements [3]. The primary benefit of this approach is its robustness–providing deterministic outcomes and guaranteeing product performance. However, it often leads to unnecessarily tight tolerances, as it does not account for the probabilistic nature of manufacturing variations. This can increase manufacturing costs, as tighter tolerances typically require more precise, and therefore more expensive, production processes.

In contrast, statistical tolerance analysis considers a probabilistic approach, treating dimensional variations as random variables that follow defined distributions (often normal). This method estimates the likelihood that an assembly will meet functional requirements by considering the combined effects of part deviations [16]. One of its main advantages is the ability to relax tolerances, compared to worst-case analysis, which can lead to cost savings in manufacturing. It also more accurately reflects the variation typically found in production environments.

However, statistical tolerance analysis has some limitations. It does not guarantee that every part will meet specifications, making it less suitable for high-reliability applications requiring strict functional compliance [14]. Additionally, its effectiveness depends on the accuracy of statistical data, such as part variation distributions and correlations, which can be difficult to define due to limited production data. Traditional methods, like the Root Sum Squared (RSS) model [6] and second-order tolerance-analysis techniques [1], assume precise defect distribution models, which may not always be available. To overcome these challenges, more advanced methods have been developed, focusing on defining the admissible conformity space using tolerance intervals and statistical characteristics. These approaches enable the simulation of a broader range of statistical cases, offering more robust solutions by better-incorporating uncertainty in process variability [5].

Managing uncertainty in process variability inherently requires addressing epistemic uncertainty. In this context, methods based on imprecise probabilities - such as interval theory, P-Boxes, and fuzzy set theory - have gained popularity [4]. These approaches incorporate the lack of precise knowledge about the system into the analysis, ensuring that uncertainty is systematically propagated to the results. These methods allow engineers to estimate upper and lower bounds on defect probabilities, leading to more robust and realistic assessments of design tolerances.

This paper presents an approach integrating epistemic uncertainty into statistical tolerance analysis using probability boxes. The proposed methodology estimates the worst-case non-conformity rate of a system, without requiring precise assumptions about defect distributions, making it more adaptable to real-world manufacturing variability. To demonstrate its practical benefits in mechanical design, the approach is applied to two real-world case studies, highlighting its effectiveness in addressing uncertainty and improving reliability.

The article is subdivided into three major parts. Firstly, Section 2 presents the framework of the statistical tolerance analysis problem, focusing on the propagation of variability through the stack-up (assembly response) function and the general calculation of the non-conformity rate. Then, Section 3 introduces imprecise probability analysis for obtaining the worst-case non-conformity rate for each single component and, in general, for an assembly response function. Section 4 illustrates the use of imprecise probability for calculating the worst-case non-conformity rate in two case studies and compares the results with those presented in [5]. Finally, the discussion and conclusions are presented. It should be noted that, unless otherwise specified, all dimensions are assumed to be in millimeters (mm).

2 FRAMEWORK OF STATISTICAL TOLERANCE ANALYSIS

From a mathematical point of view, the functional relationship between the tolerances of the individual components and the behaviour of the final assembly can be expressed as:

$$Y = f(X_1, X_2, \cdots, X_n) = f(\mathbf{X}) \tag{1}$$

where, Y is the assembly response function (also known as the performance characteristic), and $\mathbf{X}$ are random variables representing the individual dimensions or tolerances of the components. The values of each random variable X_i are typically derived from the drawing dimensions and lie within a specified range, bounded by a lower specification limit LSL_i and an upper specification limit USL_i. The assembly quality is determined by whether Y meets the functional condition, usually represented by functional variable bounds (LSL_Y, USL_Y). The non-conformity rate P_{nc}^Y represents the probability that Y falls outside its acceptable range, which depends on the variability of the random variables. To compute this non-conformity rate, it is necessary to know the probability density functions (PDFs) of each X_i. Since these distributions are not always readily available, certain assumptions must be made [13]. Some models that can be considered are: uniformly distributed statistical model, non-shifted distribution, shifted distribution and dynamic shifted distribution. These models are briefly explained in [5].

In most of the cases, each variable X_i is assumed to follow a normal distribution. Under the non-shifted distribution model, the nominal value corresponds to the mean, and the standard deviation is defined in terms of the component tolerance T_i as:

$$\sigma_i = \frac{T_i}{k} \tag{2}$$

where k is a constant associated with the quality, and its value influences the percentage of realizations within the specified tolerance zone. For instance, if it is set to 6, approximately 99.9% of the values of X_i will fall within the tolerance limits.

Manufacturing processes inherently exhibit natural variability, which significantly impacts the quality of a batch of components in mass production [12]. The natural process tolerance is generally defined as ± 3 times the standard deviation of the manufacturing process [13]. Consequently, the specified tolerance zone T_i for each component X_i must be smaller than the natural

tolerance t_i associated with its respective manufacturing process, hence the capability index of the manufacturing process $C_{p,i}$ can be calculated as follows:

$$C_{p,i} = \frac{\text{Tolerance width}}{\text{Process variation}} = \frac{T_i}{t_i} = \frac{T_i}{6\sigma_i} \tag{3}$$

where σ_i is the standard deviation of a batch of X_i. Due to the natural process variability, the mean of the batch can be shifted. This mean shift can be modelled through the constant $C_{pk,i}$ which also includes the effect of the absolute mean shift δ_i from the nominal value [7]:

$$C_{pk,i} = \frac{T_i/2 - |\delta_i|}{3\sigma_i} \tag{4}$$

Since the capability of a manufacturing process directly influences the variability of X_i, the minimum capability indices $C_{p,i}^{(r)}$ and $C_{pk,i}^{(r)}$ must be established as requirements. Using Equations (3)-(4), the conformity region for any X_i can be determined. Figure 1a illustrates the conformity region for a given variable X_i when the feasibility constraints are:

$$C_{p,i}^{(r)} = C_{pk,i}^{(r)}, \quad C_{p,i}(\sigma_i) = C_{p,i}^{(r)}, \quad C_{pk,i}(\sigma_i, \delta_i) = C_{pk,i}^{(r)}, \quad \text{and} \quad C_{p,i}^{(r)} < C_p^0 < \infty \tag{5}$$

It is important to note that while $C_{pk,i}^{(r)}$ is not required to be equal to $C_{p,i}^{(r)}$, it must not be greater than it. When $C_{pk,i}^{(r)} < C_{p,i}^{(r)}$, the representation of the conformity region is no longer a triangle but a trapezoid, see Figure 1b. Furthermore, as C_p^0 tends to infinity, the maximum mean shift $\delta_{\max}(\sigma_{\min})$ approaches $t/2$, while $\sigma_{\min}$ tends to zero. A batch of X_i is considered compliant if its σ_i and δ_i associated are inside the conformity region.

2.1 Propagation of variability and the non-conformity rate of Y

Once the conformity region for each of the individual process variables is determined, it is important to understand how the variability in these input variables propagates through the functional relationship. The distribution of Y depends on the nature of the function $f(\mathbf{X})$ and the statistical properties of the individual input variables.

For a linear relationship, where $Y = \sum_{i=1}^{n} X_i$, the resulting distribution of Y will also follow a normal distribution with the mean and variance determined by the mean and variance of each X_i:

$$\begin{aligned} \mu_Y &= \sum_{i=1}^{n} (\mu_i + \delta_i) \\ \sigma_Y^2 &= \sum_{i=1}^{n} \sigma_i^2 \end{aligned} \tag{6}$$

In the case of having a non-linear function, the distribution of Y cannot be easily derived, and methods such as Monte-Carlo simulation are often employed to estimate it.

The non-conformity rate P_{nc}^Y of Y is then defined as the probability that it falls outside the specified tolerance limits for the final product. Specifically, given the functional limits LSL_Y and USL_Y the non-conformity rate is calculated as:

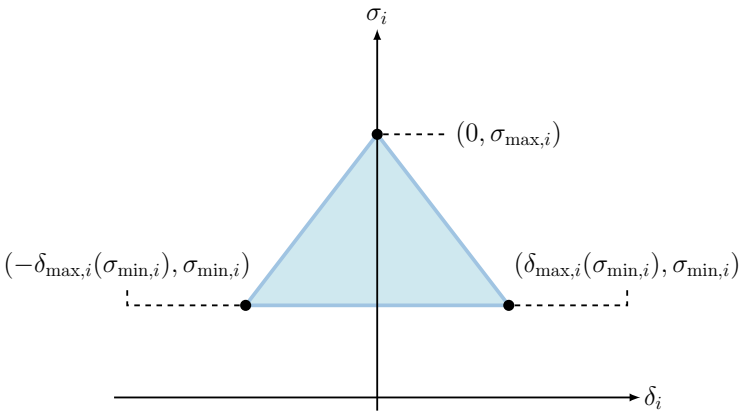
$$P_{\text{nc}}^Y = P(Y \leq LSL_Y) + P(Y \geq USL_Y) \tag{7}$$

To calculate this, the probability distribution of Y must be known. If Y is normally distributed, the non-conformity rate is given by:

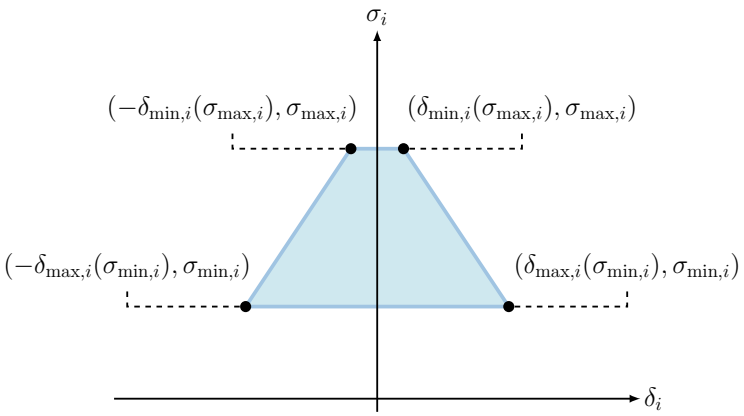
$$\begin{aligned} P(Y \leq LSL_Y) &= \Phi\left(\frac{LSL_Y - \mu_Y}{\sigma_Y}\right) \\ P(Y \geq USL_Y) &= 1 - \Phi\left(\frac{USL_Y - \mu_Y}{\sigma_Y}\right) \end{aligned} \tag{8}$$

where Φ is the Cumulative Density Function (CDF) of the normal distribution. These calculations allow for an estimation of the likelihood that the assembly will fail to meet the required specifications.

When Equation (1) takes a non-linear form, the distribution of Y is not normal. However, the non-conformity probability $P_{\text{nc}}^{Y}()$ can always be estimated using Monte Carlo simulations. While this method provides accurate results, it is computationally expensive. To address this, some authors suggest alternative approaches, such as the First-Order Reliability Method (FORM) [2] or the Second-Order Taylor Approximation (SOTA) method. The latter is based on a second-order Taylor expansion of Y and approximates its distribution using the generalized



(a) $C_{p,i}^{(r)} = C_{pk,i}^{(r)}$



(b) $C_{p,i}^{(r)} \geq C_{pk,i}^{(r)}$

Figure 1: Conformity region or Variability Domain (VD).

lambda distribution. Then, knowing the mean and standard deviation of each component, the calculation of P_{nc}^{Y} is performed using the cumulative density function of the generalized lambda distribution [1].

In [5], the authors propose a methodology to take into account the process variability in order to calculate the non-conformity rate P_{nc}^{Y}:

$$P_{\text{nc}}^{Y} = \int_{\boldsymbol{VD}} P_{\text{nc}|\boldsymbol{\delta},\boldsymbol{\sigma}}^{Y}(\boldsymbol{\sigma},\boldsymbol{\delta}) \prod_{i=1}^{n} h_{\delta,\sigma}(\delta_i,\sigma_i) d\delta_i d\sigma_i \tag{9}$$

where VD represents the variability domain within the conformity region (Figure 1), n is the number of variables X, and $h_{\delta,\sigma}(\delta_i,\sigma_i)$ denotes the joint probability density function of the mean shifts δ_i and standard deviations σ_i.

When there is not explicit knowledge of $h_{\delta,\sigma}(\delta_i,\sigma_i)$, due to dependencies on process conditions and potential correlations, it is still possible to calculate the upper bound of the non-conformity rate:

$$P_{\text{nc}}^{Y(U)} = \max_{(\boldsymbol{\delta},\boldsymbol{\sigma})\in VD} P_{\text{nc}|\boldsymbol{\delta},\boldsymbol{\sigma}}^{Y}(\boldsymbol{\sigma},\boldsymbol{\delta}) \tag{10}$$

The former equation particularly useful, as it allows to estimate the worst-case non-conformity rate without the need for precise modeling.

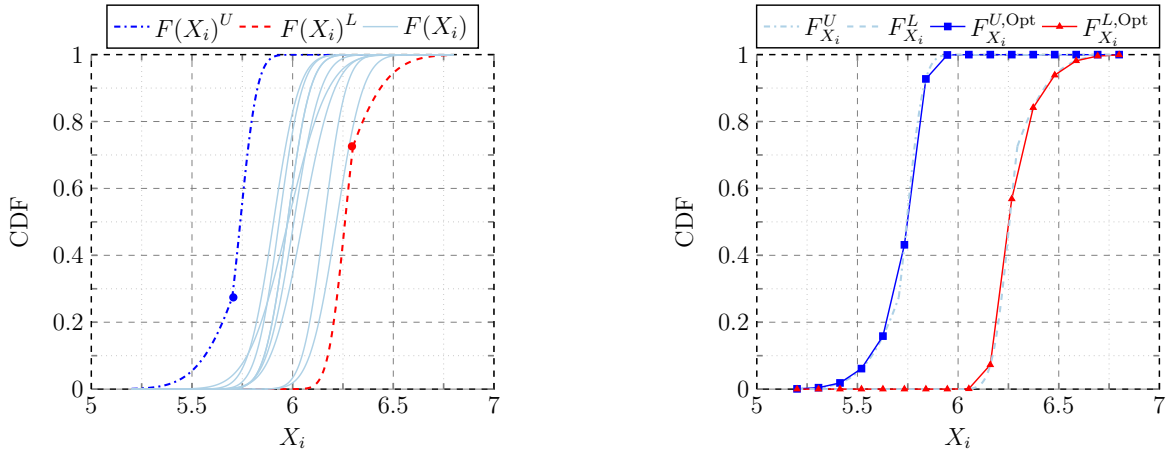
3 IMPRECISE PROBABILITY ANALYSIS

As already stated, due to the natural variability in manufacturing processes, the parameters that define each X_i cannot be known exactly, moreover, they can fluctuate. When the requirements for the capability indices, $C_{pk,i}^{(r)}$ and $C_{p,i}^{(r)}$, are set, and the maximum capability of the process, $C_{p,i}^{(0)}$, is known, it becomes possible to calculate the ranges of variation for each σ_i and δ_i. With this information it is possible to build the credal set that includes the unknown CDF of each X_i. In the same way, knowing the interaction between the X_i it is then possible to calculate the credal set of Y. P-Boxes are commonly used in engineering applications to represent a set of possible probability distributions for $F(X_i)$ bounded by a lower $F(X_i)^L$ and an upper $F(X_i)^U$ CDF. P-Boxes are computationally efficient, easy to build, and provide a simple graphical representation [4].

3.1 Modelling a single component

Consider a component X_i of dimension $6.\pm 0.3$, with $C_p^{(r)} = 0.5$, $C_{pk}^{(r)} = 0.2$ and $C_p^{(0)} = 1.5$. Since $C_p \geq C_{pk}$, the representation of the variability domain is a trapezoid, see Figure 1b. The P-Box that represents the credal set of X_i, Figure 2a, is bounded by the CDFs associated with the mean shift and standard deviations of the vertices, see Figure 3, in other terms, the the upper bound and the lower bound of the P-Box of X_i can be calculated as:

$$\begin{aligned}
F(X_i)^U &= \begin{cases} \Phi\left(\frac{x-(\mu_i-\delta_{\min,i}(\sigma_{\max,i}))}{\sigma_{\max,i}}\right), & \text{if } x < \mu_i - 3C_{p,i}^{(r)}\sigma_{\max,i} \\ \Phi\left(\frac{x-(\mu_i-\delta_{\max,i}(\sigma_{\min,i}))}{\sigma_{\min,i}}\right), & \text{if } x \geq \mu_i - 3C_{p,i}^{(r)}\sigma_{\max,i} \end{cases} \\
F(X_i)^L &= \begin{cases} \Phi\left(\frac{x-(\mu_i+\delta_{\max,i}(\sigma_{\min,i}))}{\sigma_{\min,i}}\right), & \text{if } x < \mu_i + 3C_{p,i}^{(r)}\sigma_{\max,i} \\ \Phi\left(\frac{x-(\mu_i+\delta_{\min,i}(\sigma_{\max,i}))}{\sigma_{\max,i}}\right), & \text{if } x \geq \mu_i + 3C_{p,i}^{(r)}\sigma_{\max,i} \end{cases}
\end{aligned} \tag{11}$$



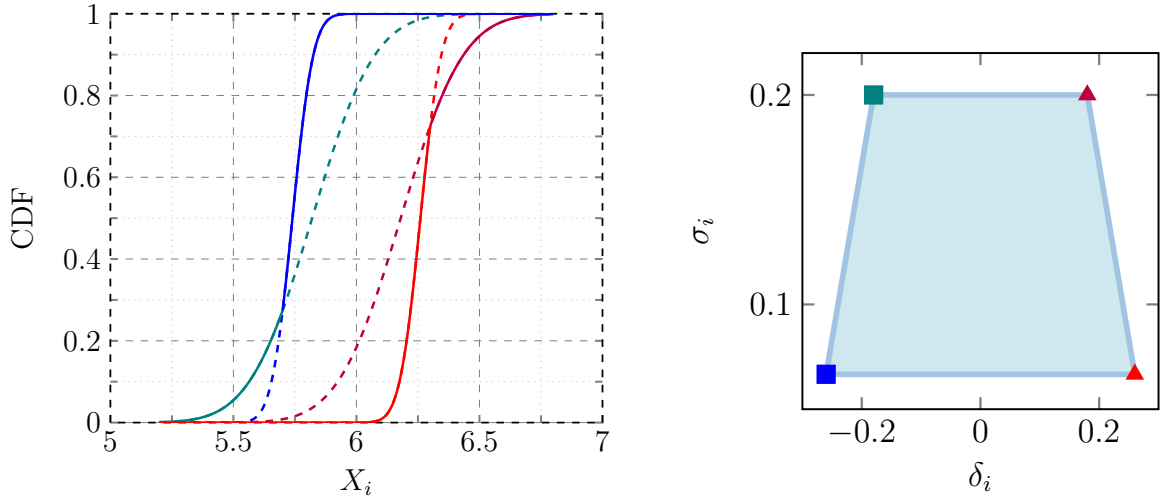
(a) Analytical P-Box

(b) Optimized P-Box

Figure 2: P-Box for a vairable X_i representing all the possible CDF within the variability domain for $C_{p,i}^{(r)} = 0.5$, $C_{pk,i}^{(r)} = 0.2$ and $C_{p,i}^{(0)} = 1.5$. In (b), the square markers represent the results of the maximization of the CDF, while the triangle markers represent the results of the minimization.

when $C_{pk,i}^{(r)} = C_{p,i}^{(r)}$ the former set of equations is simplified because $\delta_{\min,i}(\sigma_{\max,i}) = 0$.

When the calculation of the P-Box is not straightforward, numerical methods for propagating P-Boxes can be used [4]. In our case, we use the double-loop approach, in which $F_{X_i}^U$ and $F_{X_i}^L$ are obtained by solving an optimization problem that maximizes and minimizes the CDF of X_i, respectively:



(a) Analytical P-Box

(b) Variability Domain

Figure 3: Association of the CDFs that constitute the P-Box for a variable X_i, with their respective mean shifts and standard deviations from the Variability Domain (the square and triangular markers are associated with the upper bound and the lower bound, respectively).

$$
\begin{aligned}
F_{X_i}^{U,\text{opt}} &= \max\ F_{X_i}(\sigma_i, \delta_i) \\
&\text{subject to:} \quad \sigma_{\text{min},i} \leq \sigma_i \leq \sigma_{\text{max},i}, \\
&\qquad\qquad -\delta_{\text{max},i}(\sigma_i) \leq \delta_i \leq \delta_{\text{max},i}(\sigma_i)
\end{aligned} \tag{12}
$$

$$
\begin{aligned}
F_{X_i}^{L,\text{opt}} &= \min\ F_{X_i}(\sigma_i, \delta_i) \\
&\text{subject to:} \quad \sigma_{\text{min},i} \leq \sigma_i \leq \sigma_{\text{max},i}, \\
&\qquad\qquad -\delta_{\text{max},i}(\sigma_i) \leq \delta_i \leq \delta_{\text{max},i}(\sigma_i)
\end{aligned} \tag{13}
$$

When calculating the P-Box via optimization for the previous case, we obtain a good approximation to the analytical solution (see Figure 2b).

3.2 Illustrative example: Two-part assembly

The computation the P-Box of Y for any assembly response function can be done using the double loop approach. To illustrate the process, consider the mechanical assembly of two parts shown in Figure 4. In this case, the stack-up function of the system can be written as:

$$Y = X_1 + X_2 \tag{14}$$

where the nominal dimensions of the components X_1 and X_2 are 6 and 4, respectively. The tolerances of the components are equal, with $t_1 = t_2 = 0.6$. Their capability requirements are also equal, set as $C_{p1}^{(r)} = C_{p2}^{(r)} = 1$, $C_{pk1}^{(r)} = C_{pk2}^{(r)} = 1$ and $C_{p1}^{0} = C_{p2}^{0} = 1.5$. The mechanical requirement for the assembly can be written as:

$$Y = X_1 + X_2 \in [9.5, 10.5] \tag{15}$$

where 9.5 and 10.5 respresent the functional variable bounds, LSL_Y and USL_Y, respectively. Figure 5a shows the calculated P-Box for Y given the tolerances and the capability requirements. The worst-case non-conformity rate can be calculated by means of an optimization process

$$
\begin{aligned}
P_{\text{nc}}^{Y(U)} &= \max\ \left(F_Y(\sigma_i, \delta_i, LSL_Y) + (1 - F_Y(\sigma_i, \delta_i, USL_Y))\right) \\
&\text{subject to:} \quad \sigma_{\text{min},i} \leq \sigma_i \leq \sigma_{\text{max},i}, \\
&\qquad\qquad -\delta_{\text{max},i}(\sigma_i) \leq \delta_i \leq \delta_{\text{max},i}(\sigma_i)
\end{aligned} \tag{16}
$$

It is important to note that, in this case, the previous optimization process, Equation (16) differs from the optimization used to generate the P-Box of the stack-up function (Equations (12)-(13)). As seen in Equation (15), the functional condition includes both a lower and an upper

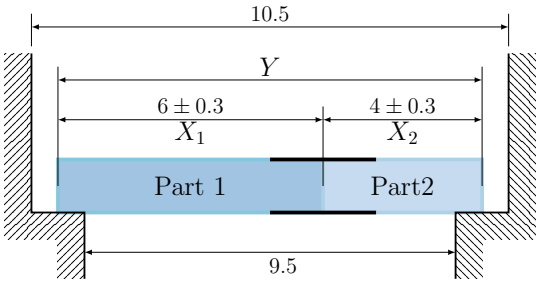


Figure 4: Simple mechanical assembly of two parts in a serial configuration

bound, meaning the worst-case non-conformity rate is influenced by both boundaries of the P-Box.

For the functional condition, Equation (15), the worst-case non-conformity rate $P_{\text{nc}}^{Y(U)}$ is approximately 731 ppm. In this case, the mean shift and standard deviation for both variables associated with the calculation of $P_{\text{nc}}^{Y(U)}$ are presented in Figure 6.

Figure 5b illustrates the evolution of the maximum non-conformity rate as the functional variable bounds, LSL_Y and USL_Y, vary—expanding the range of the functional condition. Specifically, LSL_Y ranges from $[9.6, \ldots, 9.2]$ and USL_Y from $[10.4, \ldots, 10.8]$, resulting in a tolerance zone for Y of $[0.8, \ldots, 1.6]$, computed as $S = USL_Y - LSL_Y$.

In Figure 6, we observe how the mean shift and standard deviation associated with the maximum non-conformity rate evolve, shifting toward the upper vertex of the triangle when the tolerance zone is less than 1.1. This is noteworthy because it suggests that the worst-case non-conformity rate is not always linked to the minimum standard deviation and an extreme mean shift. Instead, an alternative configuration may yield an even higher non-conformity rate, indicating that the optimization problem becomes nontrivial in more complex systems.

4 CASE STUDIES

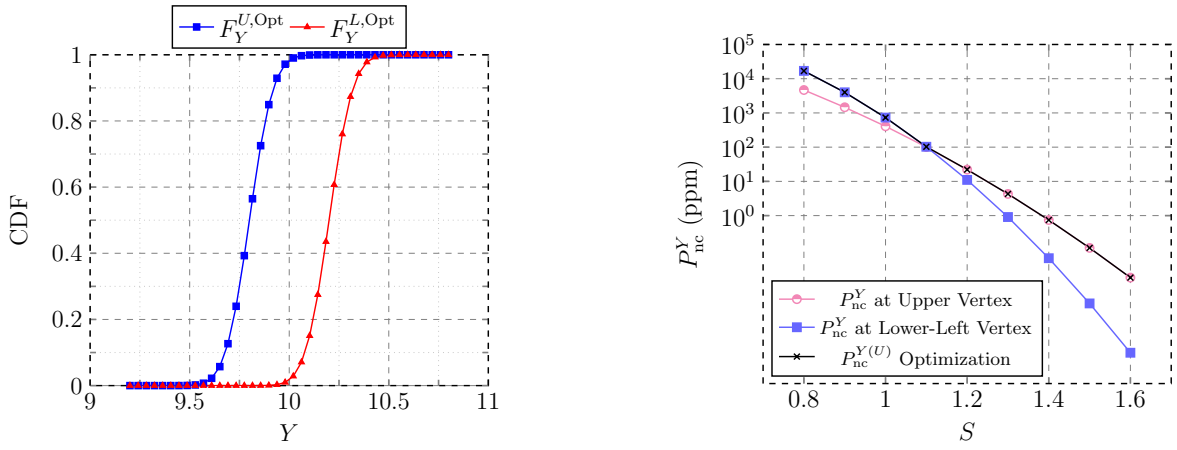
Let's consider the two industrial applications presented in [5]. Both industrial cases are provided by RADIALL; the first one is a linear application, while the second one is non-linear.

4.1 Linear industrial application

This example consists of an electrical pin whose functionality depends on a variable Y that represents the contact length of the pin in its socket. Y depends on 17 dimensions linked together by the following stack-up function:

$$Y = f(X_i) = \sum_{i=1}^{17} k_i X_i, \quad k_i = \pm 1 \tag{17}$$

The functional requirement of the system is to ensure that the contact length of the pin (Y)



(a) P-Box of Y

(b) Non-conformity rate

Figure 5: (a) P-Box of Y for the simple mechanical assembly of two parts in a serial configuration, and (b) sensitivity of the non-conformity rate regarding the functional condition S.

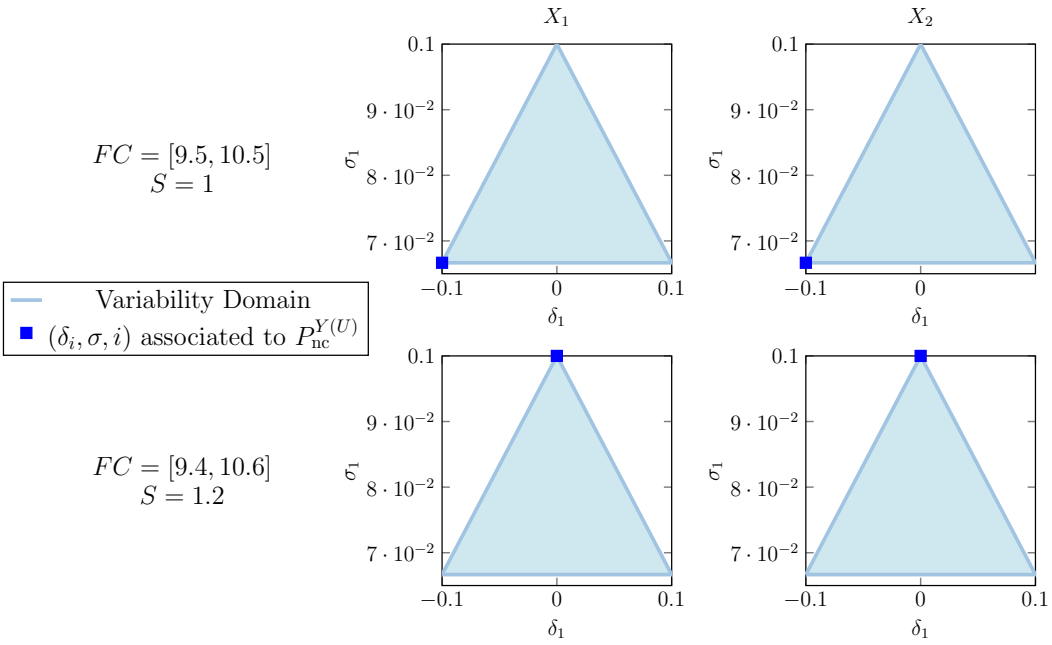


Figure 6: Values of mean shift and standard deviation of each variable that generates the $P_{\text{nc}}^{Y(U)}$ for different functional conditions.

Dim.	k_i	Nominal Dimension	t_i	$C_{p,i}^{(r)} = C_{pk,i}^{(r)}$	$C_{p,i}^{0}$
1	-1	0.750	0.20	1.8	2.6
2	1	10.710	0.16	1.8	2.6
3	1	3.750	0.10	1.8	2.6
4	-1	5.125	0.05	n.a	n.a
5	-1	3.125	0.05	n.a	n.a
6	1	4.505	0.03	1.7	1.94
7	1	5.985	0.09	1.8	2.8
8	-1	21.535	0.11	0.38	1.32
9	-1	8.195	0.11	0.38	1.32
10	1	5.985	0.09	1.8	2.8
11	1	12.305	0.03	1.7	1.94
12	-1	10.325	0.05	n.a	n.a
13	-1	5.125	0.05	n.a	n.a
14	1	3.750	0.10	1.8	2.6
15	1	0.825	0.05	1.05	1.85
16	1	8.800	0.10	1.8	2.6
17	-1	0.400	0.20	1.8	2.6

Table 1: RADIALL Linear application: Dimension details, from [5].

is greater than or equal to a given threshold S, $Y \geq S$. This requirement is considered the minimum contact length needed to ensure good conductivity in an electrical circuit. The values of k_i, the nominal dimension of each X, their tolerances, and the capability requirements are given in Table 1. The parts X_i for $i = [4, 5, 12, 13]$ are considered difficult to manufacture

within tolerance; hence, each dimension is verified after fabrication. This consideration is taken into account in the optimization process by setting $\sigma_i = 0$ and $\delta_i \in [-t_i/2, t_i/2]$ for these four variables.

The P-Box of Y can be calculated through the double-loop optimization, as shown in Figure 7a. Since the functional condition limits only the lower bound of Y, we can calculate the maximum non-conformity rate by considering only the upper bound of the P-Box:

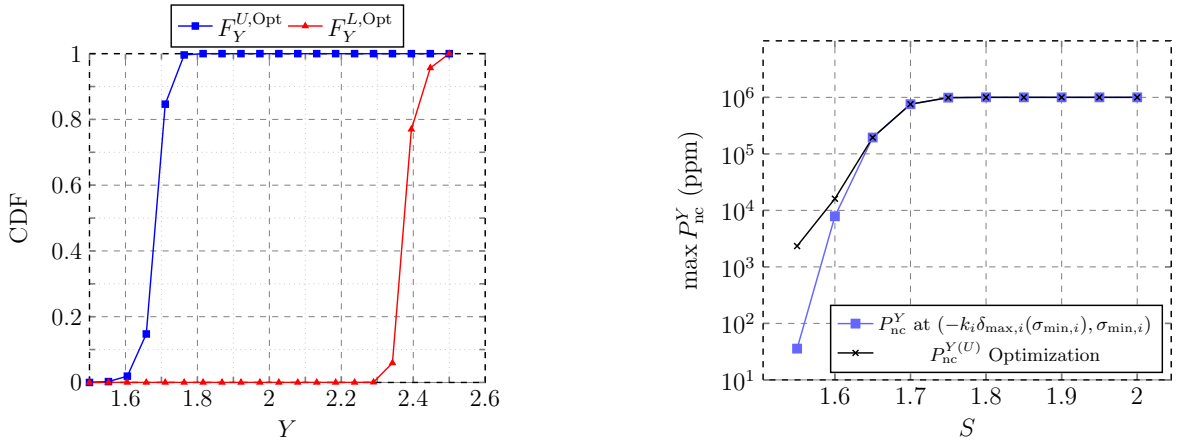
$$P_{\text{nc}}^{Y(U)} = F_Y^{U,\text{Opt}}(S) \tag{18}$$

To perform this optimization, we employ a local optimization algorithm initialized at 100 different random starting points. This approach helps mitigate the risk of converging to a local minimum. For $S = 1.75$, the maximum non-conformity rate is approximately 98.7%. This corresponds to the mean shift and standard deviations presented in Figure 8a. These results are consistent with those presented in [5], with slight differences potentially arising from the optimization process.

Additionally, Figure 7b illustrates the evolution of the maximum non-conformity rate as a function of S, highlighting its sensitivity to this parameter. For $S < 1.65$, the maximum non-conformity rate values are higher than those reported in [5]. This discrepancy arises because that study assumes the worst-case scenario always occurs at the lower vertex of the variability domain (smallest standard deviation and for the mean shift $-k_i\delta_{\max,i}(\sigma_{\min,i})$). However, our results demonstrate that this is not always the case. As shown in Figure 8b, for $S < 1.65$, the mean shift and standard deviation that yield the worst-case value can sometimes be associated with the upper vertex of the variability domain (components X_8, X_9).

4.2 Non-linear industrial application

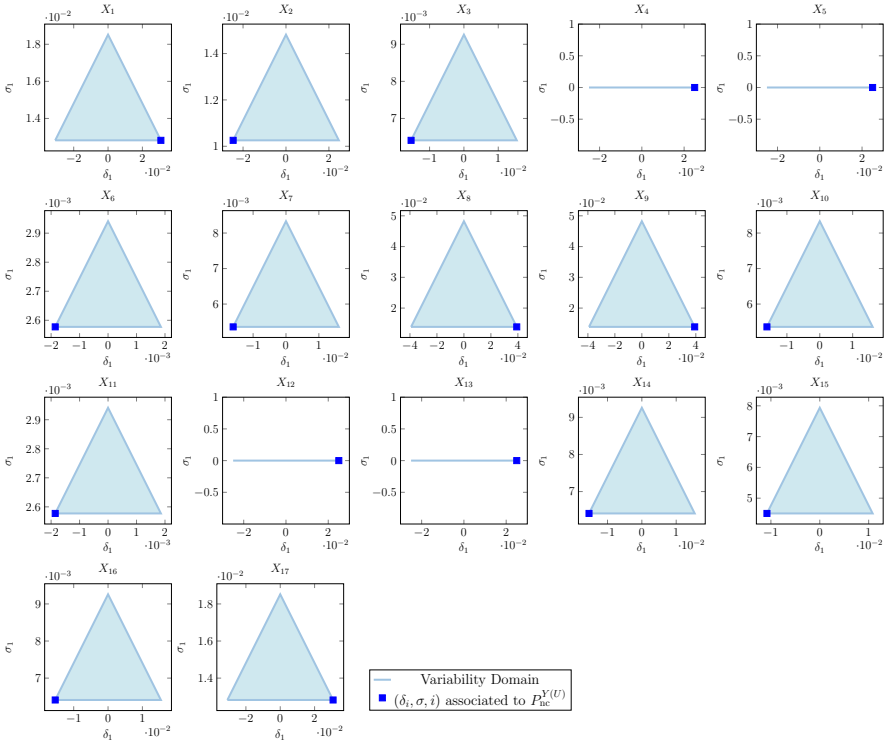
This application involves modeling the amplitude of a pin's extremity, which can lead to assembly issues in an electrical system, see Figure 9. The function describing this variation depends on 14 dimensions and follows the relationships provided below:



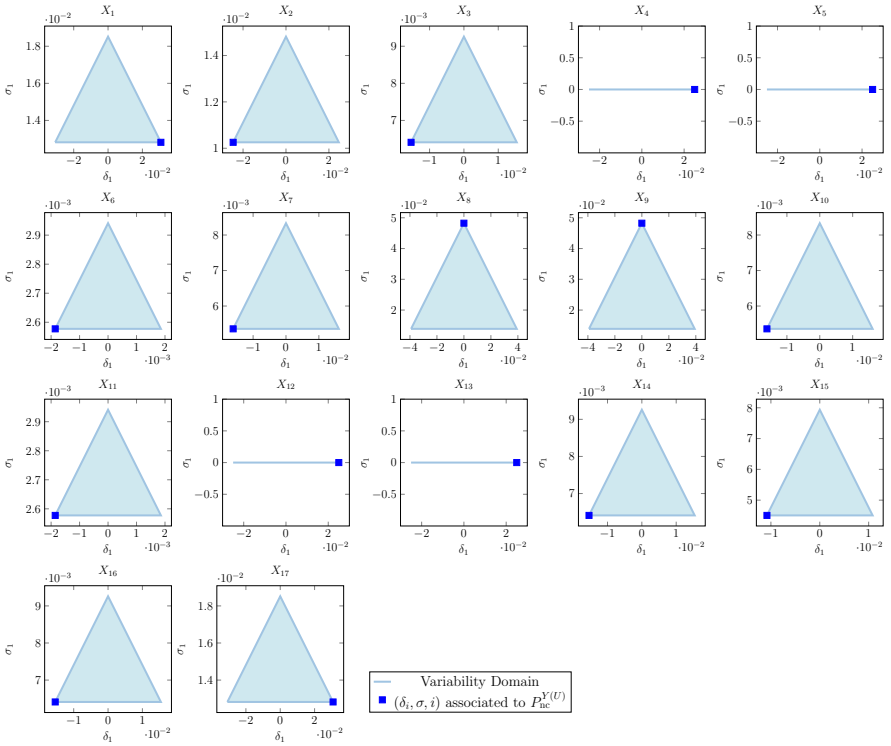
(a) P-Box of Y

(b) Non-conformity rate

Figure 7: (a) P-Box of Y for the linear case representing all the possible CDF taking into account the variability of all the X_i, and (b) the sensitivity of the non-conformity rate regarding the value of S.



(a) $S > 1.65$



(b) $S < 1.65$

Figure 8: Mean shift and standard deviation values that generate the maximum non-conformity rate ($P_{\text{nc}}^{Y(U)}$) for different values of S.

$$
\begin{aligned}
c &= \frac{cm_4 + cm_8 + cm_7}{2} \\
i &= \frac{ima_{17} + ima_{19} + ima_{20}}{2} \\
h &= ima_2 + ima_3 \\
\alpha &= \arccos\left(\frac{c}{\sqrt{i^2+h^2}}\right) - \arccos\left(\frac{i}{\sqrt{i^2+h^2}}\right) \\
r2 &= \frac{1}{\tan(\alpha)}\left(\frac{ima_{19}+ima_{20}}{2} - \frac{cm_8+cm_7}{2\cos(\alpha)}\right) \\
z &= \frac{r_2}{\cos(\alpha)} + \left(\frac{cm_9+cm_{10}}{2} + \frac{cm_7}{4}\right)\tan(\alpha) \\
J1 &= (cm_1 - cm_3 - z)\sin(\alpha) \\
J2 &= \frac{cm_7}{4}\cos(\alpha) \\
J3 &= \frac{cm_5+cm_6}{2}\cos(\alpha) \\
Y &= J1 + J2 + J3
\end{aligned}
$$

The information regarding component dimensions, tolerances, and process capability indices is presented in Table 2. The functional requirement of the system is to ensure that the amplitude of the pin's extremity is below a threshold S, $Y < S$.

Since the stack-up function of the system is nonlinear, we will employ the SOTA methodology to construct the P-Box and compute the non-conformity rate. Figure 10a illustrates the P-Box of the system. Given that the controlled bound in the functional condition is the upper bound of Y, the worst-case non-conformity rate must be calculated using the lower bound of the P-Box. Thus:

$$P_{\text{nc}}^{Y(U)} = 1 - F_Y^{L,\text{Opt}}(S) \tag{19}$$

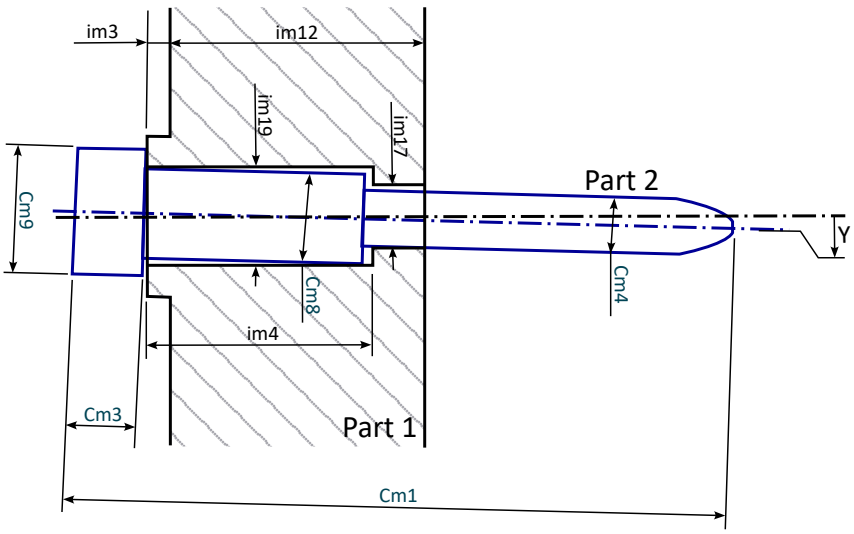
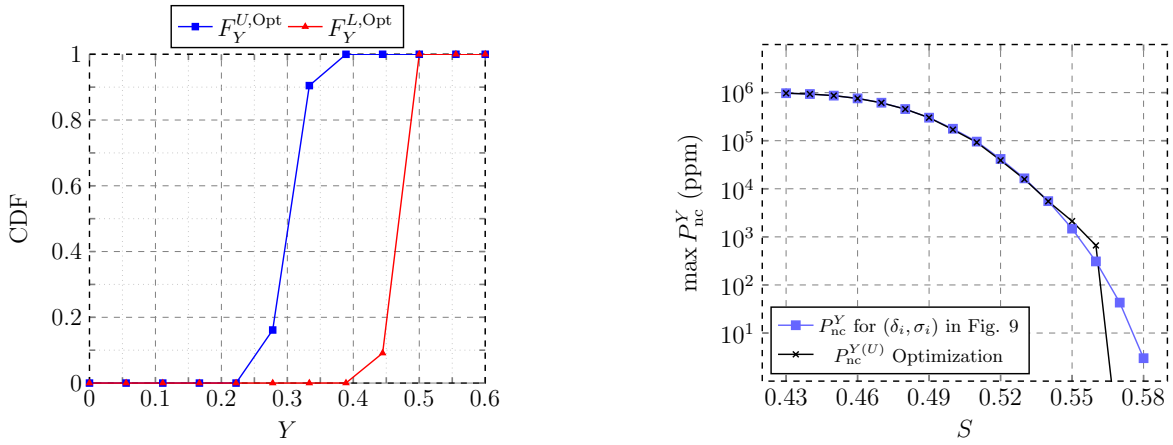


Figure 9: Non-linear industrial case, reproduced from [5].

Dim.	Nominal Dimension	t_i	$C_{p,i}^{(r)} = C_{pk,i}^{(r)}$	$C_{p,i}^{0}$
cm1	10.53	0.2	1.1	1.6
cm3	0.75	0.04	1.1	1.6
cm4	0.643	0.015	1.1	1.6
cm5	0.1	0.2	1.1	1.6
cm6	0	0.06	1.1	1.6
cm7	0	0.2	1.1	1.6
cm8	0.72	0.04	1.1	1.6
cm9	1.325	0.05	1.1	1.6
cm10	0	0.04	1.1	1.6
ima2	3.02	0.06	0.86	1.36
ima3	0.4	0.06	0.86	1.36
ima17	0.72	0.04	0.86	1.36
ima19	0.97	0.04	0.86	1.36
ima20	0	0.04	0.86	1.36

Table 2: RADIALL Non-Linear application: component dimensions and process capability indices, from [5].



(a) P-Box of Y

(b) Non-conformity rate

Figure 10: (a) P-Box of Y for the non-linear case representing all the possible CDF taking into account the variability of all the X_i, and (b) the sensitivity of the non-conformity rate regarding the value of S.

For $S = 0.52$, the maximum non-conformity rate is approximately 4.1%, corresponding to the mean shifts and standard deviations presented in Figure 11. Figure 10b illustrates the evolution of the maximum non-conformity rate as a function of S, showing that it decreases as S increases. Since this is a non-linear case, the cumulative distribution function (CDF) is computed through the SOTA methodology, using a generalized lambda distribution (GLD). The resulting CDF serves as input for our optimization algorithm—a local optimization approach initialized at 100 different random starting points. However, this method becomes computationally expensive, primarily due to the number of points required for GLD estimation. In our case, 2^{14} samples are generated using a Sobol sequence for the 14 variables.Additionally, as

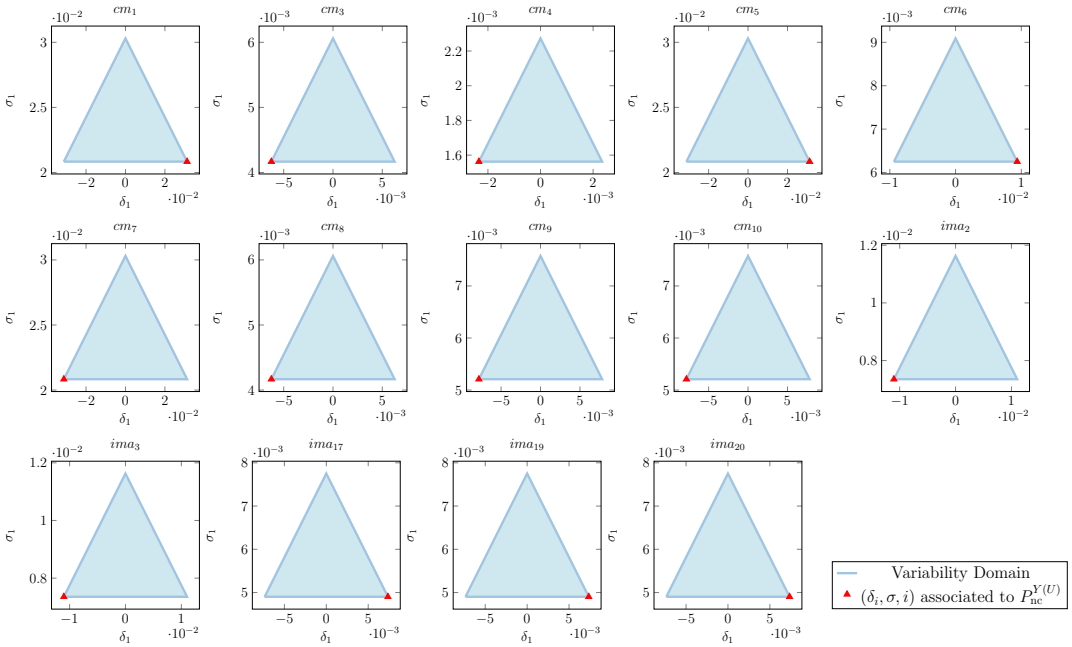


Figure 11: Values of mean shift and standard deviation of each variable that generates the $P_{\text{nc}}^{Y(U)}$ for different values of S in the non-linear case.

we approach the tail of the distribution, the optimization algorithm tends to get trapped in local minima, making it increasingly difficult to determine the worst-case non-conformity rate beyond $S = 0.55$.

If we assume the worst-case configuration to be the one presented in Figure 11 for all values of S, we obtain results that align with our optimization outcomes (up to $S = 0.55$) and with those reported in [5]. Slight differences from [5] may arise because we employ the SOTA methodology, whereas the authors in [5] used the FORM approach to calculate the non-conformity rate, along with different numerical approximations.

5 DISCUSSION

From the illustrative example of the two-part assembly, we observe that determining the worst-case non-conformity rate of a system is not necessarily straightforward. Unlike previous studies, which assumed a fixed relationship between standard deviations and mean shifts, our results demonstrate that these parameters can vary depending on the system's functional condition, as shown in Figure 6.

This observation is further reinforced in the linear industrial application. As we move further into the tail of the stack-up function's distribution, the mean shift and standard deviation of certain parts transition from the lower-left vertex to the upper vertex of the variability domain, as shown in Figure 8. This shift highlights the complexity of accurately identifying worst-case scenarios.

In the non-linear case study, we successfully computed the worst-case non-conformity rate for certain values of S. However, as we progress further into the tail of the distribution, the optimization algorithm encounters difficulties in identifying the optimal solution, Figure 10b. This occurs because the algorithm is more likely to become trapped in local minima.

To validate our approach, we compared our results with those reported in [5] for the same

case studies. These comparisons indicate that our methodology has significant potential. However, to improve convergence in the non-linear case, it would be necessary to implement a global search algorithm—such as simulated annealing, genetic algorithms, or other heuristic methods—to enhance the optimization process and address these challenges effectively.

6 CONCLUSIONS

In this work, we introduced an approach that integrates epistemic uncertainty into statistical tolerance analysis using P-Boxes. The methodology enables the estimation of worst-case non-conformity rates without requiring precise assumptions about defect distributions.

As an initial step, we applied the approach to a basic serial system consisting of two parts. We computed the corresponding P-Box and analyzed the maximum non-conformity rate under different functional conditions. This analysis revealed that the worst-case non-conformity rate does not always correspond to the same combination of standard deviation and mean shift for the components. This insight suggests that, in more complex systems, the optimization process will play an even more critical role in accurately identifying worst-case scenarios.

Furthermore, in both the linear and non-linear case studies reproduced from [5], we observed some differences in the maximum non-conformity rate compared to their results. This may be due to the fact that, when calculating the maximum non-conformity rate, the previous study considered the worst-case to be the smallest standard deviation with the largest mean shift. However, our analysis shows that this is not always the case.

The results from the case studies show that the pairs of mean shifts and standard deviations associated with each component X_i, that generate the maximum non-conformity rate are located at a vertex of the variability domain for each component. However, we cannot ensure that this holds for more complex systems, particularly parallel or over-constrained systems, where the worst-case configurations may follow a different pattern. Investigating this further would be an interesting topic for future research, as it could provide deeper insights into worst-case non-conformity rate behavior in other architectures.

ACKNOWLEDGMENTS

This work was supported by the French National Research Agency (Agence Nationale de la Recherche, ANR) under project number ANR-21-CE46-0009.

REFERENCES

[1] *A Second-Order Method for Assembly Tolerance Analysis*, volume Volume 1: 25th Design Automation Conference of *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 09 1999.

[2] Karl Breitung. Asymptotic approximations for multinormal integrals. *Journal of Engineering Mechanics*, 110(3):357–366, 1984.

[3] Vincent Delos, Denis Teissandier, Alexander Malyshev, Grégory Nuel, and Sonia García. Polyhedral-based Modelling and Algorithms for Tolerancing Analysis. *Computer-Aided Design*, page 103071, Jul 2021.

[4] Matthias G.R. Faes, Marco Daub, Stefano Marelli, Edoardo Patelli, and Michael Beer. Engineering analysis with probability boxes: A review on computational methods. *Structural Safety*, 93:102092, 2021.

[5] Gayton, Nicolas, Beaucaire, Paul, Bourinet, Jean-Marc, Duc, Emmanuel, Lemaire, Maurice, and Gauvrit, Laurent. Apta: advanced probability-based tolerance analysis of products. *Mécanique & Industries*, 12(2):71–85, 2011.

[6] W. H. Greenwood and K. W. Chase. Root sum squares tolerance analysis with nonlinear problems. *Journal of Engineering for Industry*, 112(4):382–384, 11 1990.

[7] Mikel J Harry and Reigle Stewart. *Six sigma mechanical design tolerancing*. Motorola University Press Schaumburg, IL, 1988.

[8] Y.S. Hong and T.C. Chang. A comprehensive review of tolerancing research. *International Journal of Production Research*, 40(11):2425–2459, 2002.

[9] Zbigniew Humienny. State of art in standardization in the geometrical product specification area a decade later. *CIRP Journal of Manufacturing Science and Technology*, 33:42–51, 2021.

[10] Sandipan Karmakar and Jhareswar Maiti. A review on dimensional tolerance synthesis: Paradigm shift from product to process. *Assembly Automation*, 32:373–388, 2012.

[11] R. Sampath Kumar, V. Soundararajan, and N. Alagumurthi. Review of tolerance analysis, allocation and constraints in manufacturing. *Journal for Manufacturing Science and Production*, 10(1):1–16, 2009.

[12] E. M. Mansoor. The application of probability to tolerances used in engineering designs. *Proceedings of the Institution of Mechanical Engineers*, 178(1):29–39, 1963.

[13] Swami D Nigam and Joshua U Turner. Review of statistical approaches to tolerance analysis. *Computer-Aided Design*, 27(1):6–15, 1995.

[14] Jami J. Shah, Gaurav Ameta, Zhengshu Shen, and Joseph Davidson. Navigating the tolerance analysis maze. *Computer-Aided Design and Applications*, 4(5):705–718, 2007.

[15] Michael Walter and Sandro Wartzack. Statistical tolerance-cost-optimization of systems in motion taking into account different kinds of deviations. In Michael Abramovici and Rainer Stark, editors, *Smart Product Engineering*, pages 705–714, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[16] D.E. Whitney, O.L. Gilbert, and M. Jastrzebski. Representation of geometric variations using matrix transforms for statistical tolerance analysis in assemblies. *Research in Engineering Design*, 6:191–210, 1994.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

NAVIGATING UNCERTAINTY IN IMAGE-BASED STRUCTURAL HEALTH MONITORING: THE ROLE OF FALSE POSITIVES, FALSE NEGATIVES, AND BASE RATE BIAS

Vagelis Plevris[1] **and Alejandro Jiménez Rios**[2]

[1] Department of Civil and Environmental Engineering, College of Engineering, Qatar University
P.O. Box: 2713, Doha, Qatar
e-mail: vplevris@qu.edu.qa

[2] Department of Architecture & Civil Engineering, University of Bath
6 East, University of Bath, Claverton Down, Bath, BA2 7AY, UK
e-mail: ajr225@bath.ac.uk

Abstract

This study investigates the practical limitations and reliability challenges of image-based structural health monitoring (SHM) systems that utilize machine learning and computer vision to detect damage in infrastructure. While such systems offer scalable and efficient alternatives to manual inspections, their performance is significantly affected by false positives, false negatives, environmental variability, and statistical biases – particularly base rate bias. When actual damage is rare, even highly accurate models may produce an overwhelming number of false alarms, undermining trust in automated detection and leading to inefficient allocation of inspection resources. Using numerical simulations, the paper illustrates how base rate bias reduces precision in low-damage environments and explores the broader implications for system design, operational risk, and decision-making. Several mitigation strategies are presented, including threshold adjustments based on structural criticality, human-in-the-loop verification, hybrid monitoring approaches, continuous model updating, and risk-based decision frameworks. These findings emphasize the importance of context-aware deployment and uncertainty management in ensuring the effectiveness and real-world applicability of image-based SHM systems.

Keywords: Structural Health Monitoring (SHM), Image-Based Damage Detection, Base Rate Bias, Risk-Based Decision Making, Uncertainty in SHM Systems.

1 INTRODUCTION

Structural health monitoring (SHM) is essential for maintaining the safety and longevity of infrastructure systems, enabling early detection of damage and supporting timely maintenance decisions [1]. Various artificial intelligence (AI) and machine learning (ML) techniques have been successfully applied in civil engineering contexts, demonstrating their potential to revolutionize traditional inspection processes [2]. Recent studies have explored the growing influence of AI on SHM, emphasizing its role in enhancing infrastructure maintenance and safety [3-5]. Advances in ML and computer vision have led to the rise of image-based SHM techniques, which promise to transform traditional inspection processes by offering faster, scalable, and more cost-effective assessments [6-8]. Using drones, high-resolution cameras, and deep learning models, these methods automate the collection and analysis of visual data, reducing reliance on manual inspections that are often slow, labor-intensive, and subject to human variability [6, 8].

Beyond efficiency, image-based techniques provide significant practical advantages. They allow for continuous monitoring through real-time image analysis, enabling early detection of structural issues before they escalate [9, 10]. Drones and remote sensing technologies extend inspection capabilities to hard-to-reach or hazardous areas [8], improving both safety and coverage. These features make image-based methods particularly attractive for large-scale infrastructure networks, where rapid, comprehensive inspections are critical for effective asset management.

However, as image-based SHM methods move from controlled research environments to real-world applications, significant challenges have emerged. Although theoretical frameworks that account for both epistemic and aleatory uncertainties have been proposed [11, 12], practical deployments reveal vulnerabilities to false positives, where undamaged structures are mistakenly flagged as damaged, and false negatives, where actual damage goes undetected [9, 10]. Furthermore, environmental factors such as lighting variability, weather conditions, and image noise further complicate accurate damage detection [13]. These issues are not merely technical imperfections but have direct consequences for public safety, resource allocation, and decision-making.

Compounding these concerns is the often-overlooked influence of *base rate bias* [14], also known as *base rate neglect* [15]. In many practical cases, structural damage events are rare compared to the large number of inspected elements. This low base rate leads to a statistical phenomenon where even highly accurate models can produce a large number of misleading positive results, undermining the perceived reliability of automated systems [14, 16]. Without proper consideration of base rates, the outputs of sophisticated AI systems can be misinterpreted, resulting in inefficient maintenance efforts or overlooked critical risks [17].

This paper aims to explore the practical implications of uncertainty in image-based SHM by focusing on the dual impact of classification errors and statistical biases. In particular, we highlight how false positives, false negatives, and base rate bias interact to affect the trustworthiness of automated monitoring systems. Examples and numerical investigations are presented to demonstrate these effects in realistic scenarios. Strategies for mitigating uncertainty through improved system design, expert validation, and risk-based decision frameworks are also discussed, offering a practical perspective for engineers and practitioners engaged in infrastructure monitoring.

The remainder of this paper is organized as follows: Section 2 discusses key sources of uncertainty in image-based SHM systems; Section 3 explores the impact of base rate bias through illustrative examples; Section 4 presents numerical simulations; Section 5 outlines

practical strategies to mitigate uncertainty; and Section 6 concludes with a summary of findings and implications for real-world deployment.

2 TYPES OF UNCERTAINTY IN IMAGE-BASED SHM

Despite the impressive advances in ML and computer vision, image-based SHM systems remain affected by both uncertainties and classification errors. These factors can significantly impact system performance, especially when models are deployed outside controlled experimental settings [17, 18]. A key source of uncertainty is environmental variability, which may include changes in lighting, weather, or surface conditions – factors that contribute to epistemic uncertainty and can be partially reduced through better data collection and model adaptation.

In addition to such uncertainties, practical deployments are also hindered by classification errors, particularly false positives and false negatives. These outcomes result from the limitations of the model's decision boundaries and play a critical role in shaping the reliability of damage detection. Together, uncertainty and error introduce risk to automated SHM and have important implications for both safety and maintenance efficiency.

2.1 Environmental Variability

Environmental conditions introduce a further layer of uncertainty. Changes in lighting, weather, surface moisture, and temperature can affect image quality and damage visibility, thus influencing the detection accuracy of ML models [13]. Unlike laboratory datasets, real-world conditions are highly variable, and models trained on idealized images often struggle to maintain performance in the field.

For instance, images taken under direct sunlight may exaggerate surface shadows, resembling cracks, whereas cloudy conditions may soften genuine crack patterns, making them less detectable [7, 13]. Seasonal variations, such as frost or plant growth near structures, can further complicate image interpretation and degrade detection reliability during routine monitoring.

2.2 Sensor and Acquisition-Related Uncertainty

In addition to environmental effects, image-based SHM systems are susceptible to uncertainties arising from sensor hardware and image acquisition processes. Poor camera calibration, lens distortion, and inconsistent imaging resolution can introduce systematic errors that degrade detection accuracy [19]. Occlusions – such as vehicles, vegetation, or structural components obstructing the camera's field of view – may partially or entirely conceal damage indicators.

Furthermore, extreme or varying viewing angles can cause perspective distortion, making it difficult for models to reliably recognize features such as cracks or spalling. These acquisition-related issues are often persistent and may not fluctuate with environmental conditions, making them harder to detect and correct. Addressing this type of uncertainty requires careful sensor placement, routine calibration, and the use of preprocessing techniques that normalize or correct image quality before analysis.

2.3 False positives and false negatives

A *false positive* occurs when the system incorrectly identifies damage where none exists. While a conservative bias toward detecting potential damage may seem prudent, high false positive rates can quickly erode trust in the system. Engineers may be overwhelmed with false alarms, leading to unnecessary inspections, wasted resources, and elevated operational costs

[9]. For example, minor surface stains or harmless material discolorations may be misclassified as cracks or spalling under varying lighting conditions, triggering false alerts during routine bridge inspections [8]. False positives are particularly problematic when the cost of verification is high or when frequent false alarms cause maintenance teams to deprioritize real warnings – a phenomenon known as "alarm fatigue" [10].

In contrast, a *false negative* happens when the system fails to detect actual structural damage. False negatives pose a more serious safety risk because undetected damage may propagate, potentially leading to catastrophic structural failures. For critical infrastructure such as highway bridges or tunnels, the consequences of missing early-stage damage can be severe, both in terms of human life and economic loss [7]. Certain conditions increase the likelihood of false negatives. For instance, faint cracks may not be visible under poor lighting, fog, or oblique camera angles. In post-earthquake assessments, dust and debris may obscure true damage, causing automated image-based systems to misclassify affected regions as undamaged [6, 8].

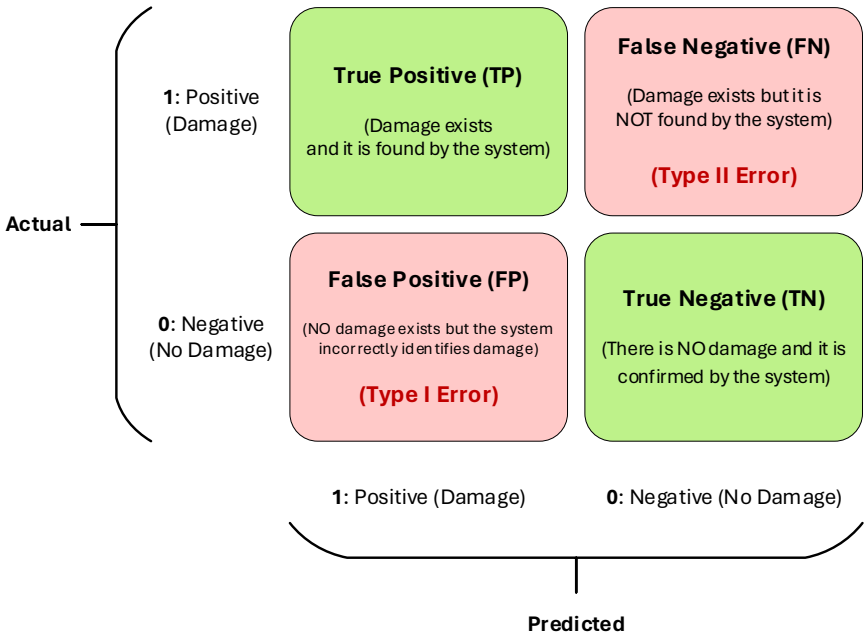


Figure 1. Confusion matrix illustrating true and false classifications in SHM damage detection.

These two types of errors are fundamental to understanding the performance of damage detection systems and are clearly illustrated in the confusion matrix shown in Figure 1, which categorizes all possible outcomes of binary classification in SHM applications.

To evaluate the performance of damage detection systems, a range of standard classification metrics can be derived from the confusion matrix shown in Figure 1. These include *Precision*, which measures the proportion of correctly identified positive cases out of all positive predictions (Eq. (1)), and *Negative predictive value*, which assesses the proportion of correctly identified negatives among all negative predictions (Eq. (2)). *Sensitivity* or *Recall* (Eq. (3)) reflects the system's ability to correctly detect actual damage, while *Specificity* (Eq. (4)) measures the correct identification of undamaged cases. *Accuracy* (Eq. (5)) gives the overall proportion of correct classifications. Additionally, the *F1-score* (Eq. (6)) provides a harmonic mean of precision and recall, balancing both metrics into a single performance indicator. These formulas are fundamental for interpreting classification results and for comparing the performance of different SHM models in a consistent and quantitative way.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Negative\ Predictive\ Value = \frac{TN}{TN + FN} \tag{2}$$

$$Sensitivity\ (Recall) = \frac{TP}{TP + FN} \tag{3}$$

$$Specificity = \frac{TN}{TN + FP} \tag{4}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

$$F1\text{-}score = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}} = \frac{2 \cdot Recall \cdot Precision}{Recall + Precision} \tag{6}$$

2.4 Combined Impact and Practical Implications

In practice, these uncertainties do not occur in isolation. A single inspection campaign may simultaneously experience environmental noise, sensor-related distortions, and produce both false positives and false negatives – while operating in a context where actual damage is rare. The combined effect can greatly diminish the value of automated assessments if not properly addressed [9].

Understanding the various sources of uncertainty is critical for designing monitoring strategies that reflect the real-world limitations of image-based systems. As discussed in the following sections, these challenges become even more pronounced when considering the role of base rate bias, where the rarity of true damage events further distorts the interpretation of positive results.

3 BASE RATE BIAS: A PRACTICAL PROBLEM

Beyond the well-recognized challenges of false positives and false negatives, a more subtle but equally critical issue in image-based SHM is the *base rate bias* [14, 15]. This statistical phenomenon can severely distort the interpretation of model outputs, even when the underlying classification models perform with high nominal accuracy. The base rate refers to the true underlying probability of an event occurring – in this case, the probability that a structure actually has damage. In most real-world monitoring scenarios, especially in routine inspections of large infrastructure networks, the occurrence of damage is relatively rare. For example, during regular bridge inspections, it is common to find that only a small fraction of inspected elements exhibit any significant structural damage [20].

When the base rate of damage is low, the performance of even highly accurate systems can become counterintuitive. Specifically, the probability that a positive detection truly indicates damage – the precision of the system – can be much lower than expected. This is because false positives, even if occurring at a low percentage, vastly outnumber true positives simply due to the sheer volume of intact structures [14, 16].

3.1 An Illustrative Example from Facial Recognition Safety Systems

To better illustrate the impact of base rate bias in detection systems, we turn to a real-world example outside of SHM: the use of facial recognition technology for security purposes. While this example touches on a sensitive application, we set aside any ethical concerns or societal controversies surrounding the deployment of such systems. Our sole intention is to present a simplified and relatable scenario that clearly demonstrates how statistical reasoning can be affected when rare events are involved.

Consider a city with a population of 1,000,000 people. Intelligence reports suggest that approximately 100 individuals in the city are classified as terrorists, representing only 0.01% of the total population. To assist in identifying these individuals, the authorities deploy an automated facial recognition system that scans people in public spaces and issues an alert when a match is found. The system's reported performance is as follows:

- It correctly identifies a terrorist 99% of the time (true positive rate, *TPR*=99%).
- It incorrectly flags an innocent person 2% of the time (false positive rate, *FPR*=2%).

At first glance, these figures suggest a highly reliable system. However, when examining the real-world implications, a different picture emerges. Among the 1,000,000 people:

- There are 100 actual terrorists.
- There are 999,900 innocent individuals.

Applying the system's accuracy:

- 99 of the 100 terrorists will be correctly flagged (99% true positives, *TP*=99).
- 19,998 innocent people will be incorrectly flagged as threats (2% false positives, *FP*=19,998).

Thus, the total number of individuals flagged by the system will be:

$$\text{Total flagged} = TP + FP = 99 \text{ (true positives)} + 19,998 \text{ (false positives)} = 20,097 \text{ people} \tag{7}$$

The probability that a person flagged by the system is actually a terrorist is expressed by the *Precision* value, and it can be calculated as:

$$Precision = \frac{TP}{TP + FP} = \frac{99}{20,097} \approx 0.0049 = 0.49\% \tag{8}$$

In other words, despite the facial recognition system having an impressive individual accuracy, a person flagged by the system is, in fact, highly unlikely – less than 1 in 200 – to be a true terrorist. This example highlights a crucial lesson: when the base rate of the event being detected is extremely low, even highly accurate systems produce a disproportionate number of false positives. The overwhelming number of false alarms can undermine the practical utility of the system, leading to inefficient resource allocation, public distrust, and operational paralysis.

As we will see in the next section, similar statistical dynamics can arise in image-based SHM, particularly during routine monitoring where the occurrence of actual structural damage is rare. This example highlights the importance of considering base rates when interpreting positive detections – whether in facial recognition or SHM – because the same logic applies: when damage is rare, even accurate models can mislead decision-makers if base rate effects are ignored.

3.2 Broader Implications

The base rate bias has critical implications for infrastructure monitoring and the practical deployment of image-based SHM systems. One major consequence is its effect on resource allocation. A large number of false positives forces inspection teams to spend considerable time and financial resources verifying structures that are ultimately sound, leading to inefficiencies and diverting attention from areas that might genuinely require intervention.

Another important concern is the risk of complacency. When operators are repeatedly exposed to false alarms, they may begin to discount warnings altogether, increasing the danger of missing real signs of structural deterioration. Over time, this erosion of trust in the system's outputs can weaken the effectiveness of monitoring programs and delay necessary maintenance actions.

Finally, public safety and confidence are directly at stake. If damage detection systems are perceived as unreliable due to frequent false alerts or overlooked true positives, trust in automated infrastructure monitoring can decline. This not only undermines investment in new technologies but also creates potential risks by slowing down critical responses to actual damage.

In systems where positive findings can trigger costly inspections, evacuations, or repairs, understanding the influence of the base rate is not optional – it is essential for responsible decision-making. In the next section, we will present a series of numerical simulations that further explore how different combinations of system performance metrics and base rates affect the practical utility of image-based SHM systems.

4 NUMERICAL SIMULATIONS IN SHM

4.1 Simulation Setup

To investigate how detection performance is influenced by base rate bias and classification errors, we simulate the behavior of an image-based SHM system. This hypothetical system is designed to analyze visual data – such as photographs captured by drones or surveillance cameras – and determine whether structural damage is present in buildings or infrastructure elements. The system uses a binary classification model that labels each structure as either "damaged" or "undamaged" based on its analysis.

For all simulations, we assume a system with the following baseline characteristics unless otherwise specified:

- **True Positive Rate (TPR)**: 98% (i.e., the probability of detecting damage when it exists)
- **False Positive Rate (FPR)**: 5% (i.e., the probability of incorrectly flagging undamaged structures)

		Predicted		
		1. Positive (Damage)	**0. Negative (No damage)**	**SUM**
Actual	**1. Positive (Damage)**	*TPR*=98%	*FNR*=2%	**100%**
	0. Negative (No damage)	*FPR*=5%	*TNR*=95%	**100%**

Table 1: Normalized confusion matrix of the hypothetical SHM system, using percentage-based rates.

To explore how the system behaves under different conditions, we vary the **base rate** – the actual proportion of damaged structures in the monitored population – across seven values: 0.1%, 0.5%, 2%, 5%, 10%, 20% and 50%.

Each scenario simulates a population of 100,000 structures. For each case, we compute the number of true positives (*TP*), which represent correctly identified damaged structures, and the number of false positives (*FP*), which represent undamaged structures incorrectly flagged as damaged. Using these values, we calculate the *Precision* of the system, defined as the proportion of positive results that are correct.

4.2 Results and Observations

Table 2 summarizes the results of this simulation for a population of 100,000 structures

Base Rate (*BR*)	**Damaged structures (*DS*)**	**Not damaged structures (*NDS*)**	**True Positives (*TP=DS×TPR*)**	**False Positives (*FP=NDS×FPR*)**	**Precision = *TP*/(*TP*+*FP*)**
0.1%	100	99,900	98	4,995	1.92%
0.5%	500	99,500	490	4,975	8.97%
2%	2,000	98,000	1,960	4,900	28.57%
5%	5,000	95,000	4,900	4,750	50.78%
10%	10,000	90,000	9,800	4,500	68.53%
20%	20,000	80,000	19,600	4,000	83.05%
50%	50,000	50,000	49,000	4,500	95.15%

Table 2: Effect of varying base rates on true positives, false positives, and precision in a simulated SHM system monitoring 100,000 structures.

Figure 2 shows a simple plot of precision versus base rate for the assumed system parameters, where the true positive rate is fixed at 98% and the false positive rate at 5%. The graph clearly shows that precision increases significantly as the base rate of actual damage rises. When the base rate is low – as in routine monitoring scenarios – precision remains below 10%, meaning that the vast majority of positive detections are false alarms. As the base rate increases, particularly beyond 10%, the precision improves rapidly, reaching over 95% at a 50% base rate. This trend highlights the strong influence of base rate bias: even a high-performing system can produce unreliable outputs in environments where actual damage is rare. The figure underscores the need to consider the underlying base rate when interpreting detection results and evaluating the practical reliability of image-based SHM systems.

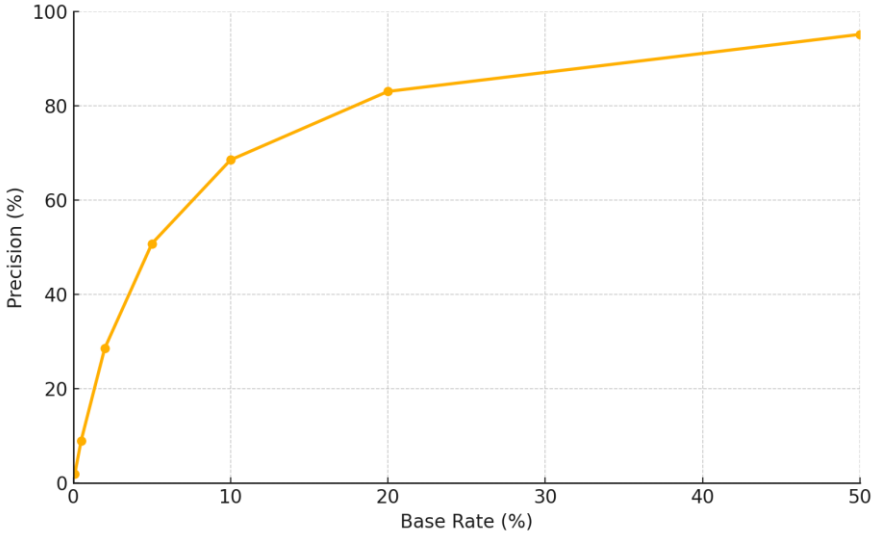


Figure 2. Precision of damage detection as a function of base rate, assuming fixed true positive (TPR=98%) and false positive (FPR=5%) rates.

4.3 Practical Takeaways

The results of the simulations reveal several important insights. First, when the base rate of actual damage is low, the impact of false positives becomes greatly amplified, making positive detections significantly less reliable. In such scenarios, even a modest false positive rate can overwhelm true detections, leading to a high number of false alarms. Second, it becomes clear that precision cannot be meaningfully interpreted in isolation; its value depends heavily on the underlying base rate of the monitored environment. Without this context, performance metrics may be misleading. Finally, for systems deployed in environments where damage is rare, achieving reliable results requires either an exceptionally low false positive rate – much lower than 5% – or the integration of additional validation mechanisms, such as human review or cross-checking with other data sources. In the next section, we examine several practical strategies designed to address these uncertainties and enhance the overall reliability of image-based SHM systems.

5 STRATEGIES TO MITIGATE UNCERTAINTY

The numerical simulations presented in the previous section clearly show that false positives, false negatives, and base rate bias can significantly impact the practical reliability of image-based SHM systems. To ensure that these systems contribute meaningfully to infrastructure management, it is essential to implement strategies that address these challenges. This section outlines practical approaches to mitigate uncertainty and improve the trustworthiness of image-based SHM in real-world applications.

5.1 Threshold Adjustment Based on Application Criticality

One effective strategy is to adjust the decision thresholds of ML models based on the criticality of the monitored structure. Many models provide a confidence score associated with each prediction. By setting a higher threshold for positive classifications, the number of false positives can be reduced – at the cost of potentially increasing false negatives.

For high-risk structures such as suspension bridges, critical tunnels, or dams, a more conservative threshold (favoring sensitivity) might be appropriate to avoid missing true damage.

Conversely, for less critical structures like minor pedestrian bridges, a higher precision threshold can help minimize unnecessary inspections.

Adaptive thresholding allows practitioners to balance risk and cost according to the specific context of each monitoring campaign.

5.2 Human-in-the-Loop Systems

Integrating human expertise into the decision-making process, commonly referred to as a human-in-the-loop (HITL) approach, offers a valuable strategy for mitigating the limitations of fully automated image-based SHM systems [21]. In this framework, positive detections generated by the system are subject to review by trained engineers before maintenance actions are taken. The human expert serves as a secondary layer of validation, using contextual judgment to confirm or override the model's predictions. This approach reduces the likelihood of false positives, strengthens confidence in system outputs, and allows for more nuanced interpretation in ambiguous cases, such as distinguishing between surface discolorations and genuine structural damage.

However, while effective in improving reliability, human-in-the-loop methods face challenges when applied at scale. For large infrastructure networks where thousands of images are processed regularly, it may be impractical to involve a human reviewer for every positive case. In such contexts, the reliance on human validation must be balanced against the need for efficiency. Selective human review, limited to high-risk detections or cases with low model confidence, may offer a compromise, preserving scalability while still addressing uncertainty in critical instances. Ultimately, the integration of human judgment should be applied strategically, complementing rather than replacing the strengths of automated SHM systems.

5.3 Hybrid Monitoring Approaches

Relying solely on image-based data can be risky, especially for detecting subtle or internal damage not visible on the surface. Combining multiple sources of information, such as vibration monitoring, strain gauges, or acoustic emission sensors, can create a hybrid monitoring system that compensates for the limitations of each method [7, 9]. For example, a positive detection from an image analysis model can be cross-validated with abnormal vibration patterns recorded by sensors. Recent work has shown that integrating artificial intelligence with traditional sensing techniques significantly enhances the robustness and responsiveness of SHM systems, particularly in the context of earthquake engineering and structural control [22]. Broader applications of AI across civil engineering, including hybrid SHM, have demonstrated the value of combining intelligent data analysis with diverse sensing strategies to improve damage detection and overall system reliability [23]. Such fusion of data streams improves overall confidence and reduces the probability of both false positives and false negatives.

5.4 Continuous Model Updating and Retraining

Image-based SHM systems should not be static. Regular retraining of ML models using updated datasets, especially with new examples of both damage and non-damage under diverse environmental conditions, can significantly improve performance over time [24]. Including real-world false positives and false negatives encountered during previous inspections into the training process allows the system to learn from its past mistakes. Adaptive models that evolve with time are better suited to cope with the variability and complexity of real-world environments.

5.5 Risk-Based Decision Frameworks

Ultimately, uncertainty in SHM should be addressed within a risk-based decision-making framework. Rather than treating all positive detections as equally important, decision-makers can assign varying levels of response based on several factors, including the criticality of the structure, the severity and confidence level associated with the detected damage, and the potential consequences of failing to identify actual damage. For instance, a low-confidence detection in a minor structure may only warrant a scheduled follow-up inspection, whereas the same detection in a major highway bridge could justify immediate on-site verification. This kind of tiered response strategy allows for more efficient use of resources by ensuring that the highest level of attention is directed toward the most critical and high-risk situations [25-27].

6 CONCLUSIONS

Image-based SHM systems, powered by advances in machine learning and computer vision, offer significant potential for enhancing the efficiency, scalability, and reach of infrastructure inspection and maintenance. However, this study demonstrates that practical deployment of such systems involves far more complexity than algorithmic accuracy alone can address. The challenges posed by false positives, false negatives, and particularly base rate bias are substantial and cannot be ignored in real-world applications.

Our simulations and examples show that even high-performing models can yield a large number of false alarms when actual damage is rare – leading to wasted resources, unnecessary inspections, and reduced confidence in system outputs. Conversely, overlooking false negatives can compromise structural safety, especially in critical infrastructure. These findings highlight the importance of understanding uncertainty not as an exception but as a core feature of SHM system design and interpretation.

To manage these limitations, SHM systems must be adapted to their operational context. This includes tuning thresholds based on structure criticality, incorporating human-in-the-loop validation, and leveraging additional data sources for cross-verification. No single configuration or approach fits all scenarios. Instead, systems must be context-aware and capable of balancing risk, cost, and reliability.

Rather than replacing manual inspection methods outright, image-based SHM should be viewed as a decision-support tool – one that assists engineers and asset managers in making informed, risk-based decisions. Effective deployment requires smarter integration of technology with human oversight, especially in low-damage environments where false alarms can erode trust.

Future work should focus on improving system robustness to environmental variability, expanding hybrid and multimodal monitoring solutions, and enhancing the interpretability of detection results through clear communication of uncertainty. By embracing the complexities of uncertainty and designing systems around them, image-based SHM can play a transformative role in supporting safer, more resilient, and more cost-effective infrastructure management.

REFERENCES

[1] Wang, G. and J. Ke, *Literature Review on the Structural Health Monitoring (SHM) of Sustainable Civil Infrastructure: An Analysis of Influencing Factors in the Implementation.* Buildings, 2024. **14**(2): p. 402 DOI: https://doi.org/10.3390/buildings14020402.

[2] Plevris, V., A. Ahmad, and N.D. Lagaros, eds. *Artificial Intelligence and Machine Learning Techniques for Civil Engineering*. 2023, IGI Global. DOI https://doi.org/10.4018/978-1-6684-5643-9.

[3] Plevris, V. and G. Papazafeiropoulos, *AI in Structural Health Monitoring for Infrastructure Maintenance and Safety.* Infrastructures, 2024. **9**(12) DOI: https://doi.org/10.3390/infrastructures9120225.

[4] Plevris, V., *AI-Driven Innovations in Earthquake Risk Mitigation: A Future-Focused Perspective.* Geosciences, 2024. **14**(9): p. 244 DOI: https://doi.org/10.3390/geosciences14090244.

[5] Jiménez Rios, A., V. Plevris, and M. Nogal, *Bridge management through digital twin-based anomaly detection systems: A systematic review.* Frontiers in Built Environment, 2023. **9**(1176621) DOI: https://doi.org/10.3389/fbuil.2023.1176621.

[6] Kim, J.-W., H.-W. Choi, S.-K. Kim, and W.S. Na, *Review of Image-Processing-Based Technology for Structural Health Monitoring of Civil Infrastructures.* Journal of Imaging, 2024. **10**(4): p. 93 DOI: https://doi.org/10.3390/jimaging10040093.

[7] Spencer, B.F., V. Hoskere, and Y. Narazaki, *Advances in Computer Vision-Based Civil Infrastructure Inspection and Monitoring.* Engineering, 2019. **5**(2): p. 199-222 DOI: https://doi.org/https://doi.org/10.1016/j.eng.2018.11.030.

[8] Mandirola, M., et al., *Use of UAS for damage inspection and assessment of bridge infrastructures.* International Journal of Disaster Risk Reduction, 2022. **72**: p. 102824 DOI: https://doi.org/10.1016/j.ijdrr.2022.102824.

[9] Fan, C.-L., *Deep neural networks for automated damage classification in image-based visual data of reinforced concrete structures.* Heliyon, 2024. **10**(19): p. e38104 DOI: https://doi.org/10.1016/j.heliyon.2024.e38104.

[10] Poudel, U.P., G. Fu, and J. Ye, *Structural damage detection using digital video imaging technique and wavelet transformation.* Journal of Sound and Vibration, 2005. **286**(4): p. 869-895 DOI: https://doi.org/10.1016/j.jsv.2004.10.043.

[11] Jiménez Rios, A., V. Plevris, and M. Nogal, *Synthetic Data Generation for the Creation of Bridge Digital Twins What-if Scenarios*, in *9th International Conference on Computational Methods in Structural Dynamics and Earthquake Engineering (COMPDYN 2023)*. 2023: Athens, Greece. p. 4801-4809. DOI: https://doi.org/10.7712/120123.10760.21262.

[12] Jiménez Rios, A., V. Plevris, and M. Nogal, *Uncertainties in the Synthetic Data Generation for the Creation of Bridge Digital Twins*, in *5th International Conference on Uncertainty Quantification in Computational Science and Engineering (UNCECOMP 2023)*. 2023: Athens, Greece. p. 39-47. DOI: https://doi.org/10.7712/120223.10323.20020.

[13] Torzoni, M., L. Rosafalco, A. Manzoni, S. Mariani, and A. Corigliano, *SHM under varying environmental conditions: an approach based on model order reduction and deep learning.* Computers & Structures, 2022. **266**: p. 106790 DOI: https://doi.org/10.1016/j.compstruc.2022.106790.

[14] Bar-Hillel, M., *The base-rate fallacy in probability judgments.* Acta Psychologica, 1980. **44**(3): p. 211-233 DOI: https://doi.org/10.1016/0001-6918(80)90046-3.

[15] Stengård, E., P. Juslin, U. Hahn, and R. van den Berg, *On the generality and cognitive basis of base-rate neglect.* Cognition, 2022. **226**: p. 105160 DOI: https://doi.org/10.1016/j.cognition.2022.105160.
[16] Welsh, M.B. and D.J. Navarro, *Seeing is believing: Priors, trust, and base rate neglect.* Organizational Behavior and Human Decision Processes, 2012. **119**(1): p. 1-14 DOI: https://doi.org/10.1016/j.obhdp.2012.04.001.
[17] Plevris, V., *Assessing uncertainty in image-based monitoring: addressing false positives, false negatives, and base rate bias in structural health evaluation.* Stochastic Environmental Research and Risk Assessment, 2025. **39**: p. 959-972 DOI: https://doi.org/10.1007/s00477-024-02898-7.
[18] Plevris, V., *Addressing the Pitfalls of Image-Based Structural Health Monitoring: A Focus on False Positives, False Negatives, and Base Rate Bias.* ArXiv e-prints, 2024(arXiv:2410.20384) DOI: https://doi.org/10.48550/arXiv.2410.20384.
[19] Yin, W., X. Zang, L. Wu, X. Zhang, and J. Zhao, *A Distortion Correction Method Based on Actual Camera Imaging Principles.* Sensors, 2024. **24**(8): p. 2406 DOI: https://doi.org/10.3390/s24082406.
[20] Fard, F. and F. Sadeghi Naieni Fard, *Development and Utilization of Bridge Data of the United States for Predicting Deck Condition Rating Using Random Forest, XGBoost, and Artificial Neural Network.* Remote Sensing, 2024. **16**(2): p. 367 DOI: https://doi.org/10.3390/rs16020367.
[21] Mosqueira-Rey, E., E. Hernández-Pereira, D. Alonso-Ríos, J. Bobes-Bascarán, and Á. Fernández-Leal, *Human-in-the-loop machine learning: a state of the art.* Artificial Intelligence Review, 2023. **56**(4): p. 3005-3054 DOI: https://doi.org/10.1007/s10462-022-10246-w.
[22] Lu, X., V. Plevris, G. Tsiatas, and D. De Domenico, *Editorial: Artificial Intelligence-Powered Methodologies and Applications in Earthquake and Structural Engineering.* Frontiers in Built Environment, 2022. **8** DOI: https://doi.org/10.3389/fbuil.2022.876077.
[23] Lagaros, N.D. and V. Plevris, *Artificial Intelligence (AI) Applied in Civil Engineering.* Applied Sciences, 2022. **12**(15) DOI: https://doi.org/10.3390/app12157595.
[24] Alzubaidi, L., et al., *A survey on deep learning tools dealing with data scarcity: definitions, challenges, solutions, tips, and applications.* Journal of Big Data, 2023. **10**(1): p. 46 DOI: https://doi.org/10.1186/s40537-023-00727-2.
[25] Hughes, A.J., R.J. Barthorpe, N. Dervilis, C.R. Farrar, and K. Worden, *A probabilistic risk-based decision framework for structural health monitoring.* Mechanical Systems and Signal Processing, 2021. **150**: p. 107339 DOI: https://doi.org/10.1016/j.ymssp.2020.107339.
[26] Feghhi, M., S.S. Khedmatgozar Dolati, A. Mehrabi, P. Malla, and D. Tabiatnejad, *A New Framework for Condition and Risk Assessment for Sustainable Management of PT Bridges.* Sustainability, 2024. **16**(22): p. 9703 DOI: https://doi.org/10.3390/su16229703.
[27] Hughes, A.J., et al., *On risk-based active learning for structural health monitoring.* Mechanical Systems and Signal Processing, 2022. **167**: p. 108569 DOI: https://doi.org/10.1016/j.ymssp.2021.108569.

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

IDENTIFICATION OF THE LATERAL TRACK IRREGULARITIES FROM THE BEHAVIOR OF THE TRAIN'S AXLES ON A SURROGATE RAILWAY DYNAMICS MODEL

M.Chihaoui[1,2], D. Duhamel[1], C.Funfschilling[2] and G. Perrin[3]

[1]Laboratoire Navier, École nationale des ponts et chaussées
6 and 8 Avenue Blaise-Pascal, 77455 Champs-sur-Marne, France
e-mail: {malek.chihaoui,denis.duhamel}@enpc.fr

[2] SNCF, Direction Technologies, Innovation et Projets Groupe
1 to 9 Avenue François Mitterrand, 93210 Saint-Denis, France
e-mail: {malek.chihaoui,christine.funfschilling}@sncf.fr

[3]COSYS, Université Gustave Eiffel, 5 Boulevard Descartes, 77420 Champs-sur-Marne, France
e-mail: {guillaume.perrin}@univ-eiffel.fr

Abstract. *This work aims at the reverse identification of the lateral offset irregularity from the kinematics of the axle on a simplified train surrogate model. The fundamental goal of this study is to have a better understanding of the dynamics of the train which would ultimately allow to identify the track irregularities from axle-boxes acceleration measurements on commercial trains. First, a known surrogate model of the dynamics of the axle from the literature is adapted to include a train carbody and the railway track irregularities and then the inversion of the system is achieved so that a relationship between the kinematics that would be measured on the axle and the lateral track irregularity is established.*

Keywords: track irregularities, lateral offset, railway maintenance, inverse problem, linear estimation, train accelerations.

1 INTRODUCTION

Irregularities in railway tracks are critical factors that must be routinely monitored to maintain comfortable, safe, and punctual train operations. These geometrical irregularities and their classification is represented on figure 1. Additionally, understanding axle-rail contact forces and track irregularities is vital for analyzing the dynamic behavior and mechanical degradation of both trains and tracks. This data serves as a key input for various computational models, including those addressing rail and wheel damage, track settlement, and wheel wear.

Currently, specialized trains are used to measure and record track geometry, requiring dedi-

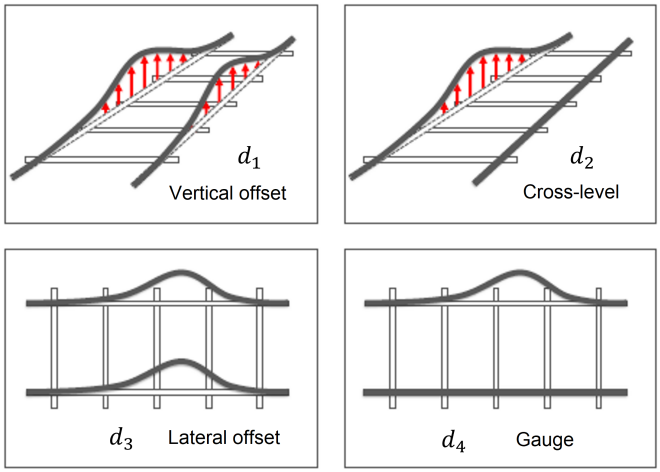


Figure 1: Classification of track geometric irregularities

cated personnel and equipment. Installing accelerometers on commercial trains and using the recorded accelerations to assess track geometry could enable more frequent and cost-effective monitoring, improving the organization and optimization of maintenance operations.

Recent studies [1, 2, 3, 4], as well as a patent [5] have explored the reconstruction of vertical track irregularities using on-board accelerometers. This involves double integration of the measured accelerations, allowing the vertical track offset of each rail to be estimated from the vertical position of the corresponding wheel. However, this method cannot be directly applied to lateral track irregularities. This is because lateral movement of the train's axles is subject to a clearance (or play) in the wheel-rail contact, enabling the axles to shift laterally between the rails. Consequently, while lateral axle position is strongly influenced by lateral irregularities, it cannot be directly inferred from them as it can for vertical irregularities. The axle's lateral dynamics between the tracks are complex, depending heavily on the train's mechanical properties and speed. Additionally, the train-track system exhibits significant non-linearity and is influenced by various uncertainties, both internal (e.g., mass, suspension characteristics) and external (e.g., wind, rain), which further complicates the analysis.

In this work, the Klingel formulas [6], which describe the natural oscillations of axles on rails, are utilized and adapted to develop a surrogate model for railway dynamics. A linear surrogate model is constructed, incorporating the train carbody and track geometry, including certain irregularities. Then, an analytical inversion of the system is performed, allowing the deduction of the lateral offset irregularity from both the axle's kinematics and the known track layout parameters. The relationship is also identified using linear regression, and both estimators are then compared.

2 SURROGATE MODEL CREATION

2.1 Lone axle's dynamics on straight railway tracks

To begin, we consider a perfectly straight track with no geometrical defects. We focus on the kinematics of a single axle of mass m and inertia j along $\overrightarrow{z}$ with a conical wheel profile of constant conicity c, which is defined from the wheel conicity angle α as $c = tan\alpha$ as shown on both figure 2 and 3. The conical profile of the wheels means that the rolling radius is not constant and changes according to the lateral position taken by the axle, as detailed in figure 2. When centered, the rolling radius is r_0. Moreover, the axle is subjected to a constant longitudinal speed

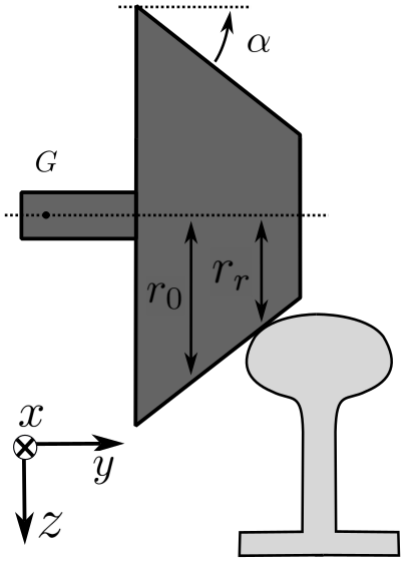


Figure 2: Rolling radius of the wheel

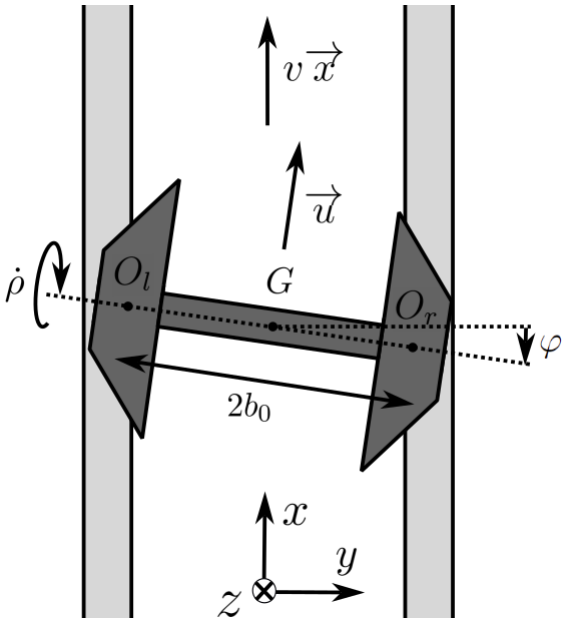


Figure 3: Diagram of the motion of the axle

$v\overrightarrow{x}$ and a constant wheel rotation speed $\dot{\rho} = \frac{v}{r_0}$. In order to take into account the interaction between the axle and the rail, it is necessary to calculate the friction forces and therefore identify the slip speed at each of the right and left wheels $\overrightarrow{v}_{slip,r/l}$. The full proof of how the creep force is obtained will not be detailed here, as it is well-documented in the literature [7, 8, 9]. However, a brief overview is provided. The relationship between the radius of each wheel and its lateral position enable to calculate the circumferential or rotation-induced speed of the contact point on the wheel at the contact interface $\overrightarrow{v}_{circ,r/l}$.

$$\overrightarrow{v}_{circ,r/l} = v(1 \pm \frac{c}{r_0}y)(\overrightarrow{x} + \varphi\overrightarrow{y}) \tag{1}$$

Whereas the translational speed at each contact interface $\overrightarrow{v}_{transl,r/l}$ is calculated using the kinematics of the axle, which is assumed not to roll but to slide perfectly on the rail.

$$\overrightarrow{v}_{transl,r/l} = (v \mp b_0\dot{\varphi})\overrightarrow{x} + \dot{y}\overrightarrow{y} \tag{2}$$

The slip speed is deduced from the difference between these two speeds at each wheel/rail interface points O_r and O_l which will be considered to be at a distance of b_0 from the axle's center of mass.

$$\overrightarrow{v}_{slip,r/l} = \overrightarrow{v}_{transl,r/l} - \overrightarrow{v}_{circ,r/l} = \mp(b_0\dot{\varphi} + \frac{vc}{r_0}y)\overrightarrow{x} + (\dot{y} - \varphi v)\overrightarrow{y} \tag{3}$$

The creep forces are computed on each wheel directly from the slip speeds using the linear Kalker creep law. κ_{11} and κ_{22} are respectively the longitudinal and lateral creep coefficients.

$$\overrightarrow{F}_{creep,r/l} = -\frac{1}{v}\begin{bmatrix}\kappa_{11} & 0\\ 0 & \kappa_{22}\end{bmatrix}.\overrightarrow{v}_{slip,r/l} \tag{4}$$

These forces at each wheel create a lateral creep force as well as a momentum along the vertical axis.

$$\begin{bmatrix}F_{creep}\\ M_{creep}\end{bmatrix} = -\begin{bmatrix}\frac{2\kappa_{22}}{v} & 0\\ 0 & \frac{2b_0^2\kappa_{11}}{v}\end{bmatrix}\begin{bmatrix}\dot{y}\\ \dot{\varphi}\end{bmatrix} - \begin{bmatrix}0 & -2\kappa_{22}\\ 2\kappa_{11}\frac{b_0 c}{r_0} & 0\end{bmatrix}\begin{bmatrix}y\\ \varphi\end{bmatrix} \tag{5}$$

Using Newton's second law of motion and this force, the dynamics equations governing a lone axle's movement can be deduced. This lateral dynamics surrogate model has been widely used in the literature, particularly for studying stability and calculating the critical speeds of trains. Typically, links to a virtual, motionless carbody centered between the rails are added, incorporating stiffness and damping elements. Indeed, with this model and at speeds below this critical speed, the axle tends to behave like a damped oscillator, swinging sideways before stabilizing in a centered lateral position.

2.2 Addition of the carbody, track parameters and irregularities

In our case, a free carbody with a lateral position of y_c and a yaw angle of φ_c of mass m_c and inertia j_c along $\overrightarrow{z}$ is added to the surrogate model. It is linked to the axle at each of the points $O_{r/l}$ with a longitudinal and a lateral stiffness K_x and K_y as well as a longitudinal and lateral damping C_x and C_y. These will create a lateral force and a torque along the vertical axis acting on both the axle and the carbody that can be written as follows :

$$\begin{bmatrix}F_{carbody/axle}\\ M_{carbody/axle}\end{bmatrix} = -\begin{bmatrix}F_{axle/carbody}\\ M_{axle/carbody}\end{bmatrix} = \begin{bmatrix}2C_y & 0\\ 0 & 2b_0^2C_x\end{bmatrix}\begin{bmatrix}\dot{y}-\dot{y}_c\\ \dot{\varphi}-\dot{\varphi}_c\end{bmatrix} + \begin{bmatrix}2K_y & 0\\ 0 & 2b_0^2K_x\end{bmatrix}\begin{bmatrix}y-y_c\\ \varphi-\varphi_c\end{bmatrix} \tag{6}$$

The next step involves incorporating the track layout parameters and irregularities into the model. These parameters will cause changes in the position of the rails along the curvilinear abscissa, affecting the creep forces. The relative lateral position of the axle in relation to the rail is denoted as y_r, which depends on the absolute lateral position of the axle y, the lateral offset irregularity d_3, and the effective roll of the rails θ_d, as shown in Figure 4.

$$y \simeq y_r + d_3 + r_0\theta_d \tag{7}$$

The track's effective cross-level angle θ_d, which incorporates both the track layout cross-level c_{track} and the irregularity d_2, is defined by $\theta_d = \tan^{-1}(\frac{c_{track}+d_2}{2b_0})$. Since we are focusing on modeling only the lateral behavior of the train, the vertical offset and its irregularity d_1 will be ignored because their impact on the dynamics is purely vertical. Additionally, gauge irregularity d_4 will be disregarded for now, as it affects the system only through non-linearities [8]. The creep forces will then be computed using y_r instead of y in Equation 5.

$$\begin{bmatrix}F_{creep/axle}\\ M_{creep/axle}\end{bmatrix} = -\begin{bmatrix}\frac{2\kappa_{22}}{v} & 0\\ 0 & \frac{2b_0^2\kappa_{11}}{v}\end{bmatrix}\begin{bmatrix}\dot{y}_r\\ \dot{\varphi}\end{bmatrix} - \begin{bmatrix}0 & -2\kappa_{22}\\ 2\kappa_{11}\frac{b_0 c}{r_0} & 0\end{bmatrix}\begin{bmatrix}y_r\\ \varphi\end{bmatrix} \tag{8}$$

The last railways parameter to take into consideration is the curvature of the track γ. Since, the frame we are computing the inertias in is non-Galilean, the centrifugal force needs to be taken

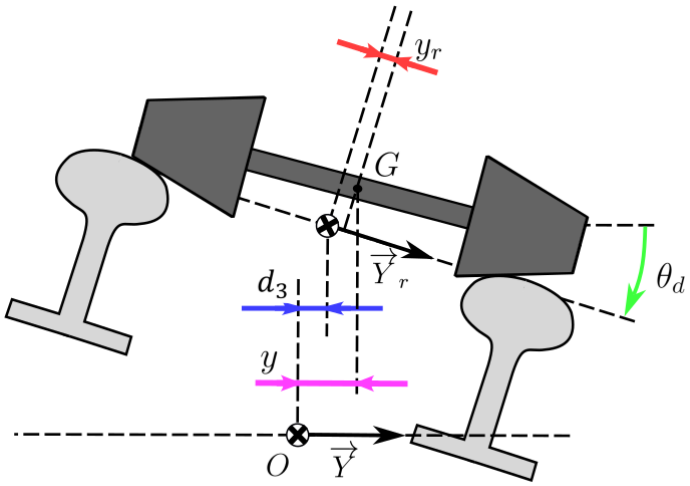


Figure 4: Diagram of the lateral motion of the axle

into consideration for both the axle and the carbody.

$$\begin{bmatrix} F_{centr/axle} \\ M_{centr/axle} \end{bmatrix} = \begin{bmatrix} -mv^2\gamma \\ -jv\dot{\gamma} \end{bmatrix}, \begin{bmatrix} F_{centr/carbody} \\ M_{centr/carbody} \end{bmatrix} = \begin{bmatrix} -m_c v^2\gamma \\ -j_c v\dot{\gamma} \end{bmatrix} \tag{9}$$

Adding up all the mentioned forces in equations 6, 8, and 9, a 4 degrees of freedom model is obtained :

$$\underline{\underline{\boldsymbol{M}}}\ddot{\boldsymbol{U}} + \underline{\underline{\boldsymbol{C}}}\dot{\boldsymbol{U}} + \underline{\underline{\boldsymbol{K}}}\boldsymbol{U} = \underline{\underline{\boldsymbol{B}}}_2\dot{\boldsymbol{E}} + \underline{\underline{\boldsymbol{B}}}_1\boldsymbol{E}, \quad \boldsymbol{U} = \begin{bmatrix} y \\ y_c \\ \varphi \\ \varphi_c \end{bmatrix}, \quad \boldsymbol{E} = \begin{bmatrix} d_3 \\ \theta_d \\ \gamma \end{bmatrix} \tag{10}$$

The mass, damping, stiffness and forcing matrices are defined as follows :

$$\underline{\underline{\boldsymbol{M}}} = \frac{1}{2}\begin{bmatrix} m & 0 & 0 & 0 \\ 0 & m_c & 0 & 0 \\ 0 & 0 & j & 0 \\ 0 & 0 & 0 & j_c \end{bmatrix}, \quad \underline{\underline{\boldsymbol{C}}} = \begin{bmatrix} C_y + \frac{\kappa_{22}}{v} & -C_y & 0 & 0 \\ -C_y & C_y & 0 & 0 \\ 0 & 0 & b_0^2(C_x + \frac{\kappa_{11}}{v}) & -b_0^2 C_x \\ 0 & 0 & -b_0^2 C_x & b_0^2 C_x \end{bmatrix} \tag{11}$$

$$\underline{\underline{\boldsymbol{K}}} = \begin{bmatrix} K_y & -K_y & -\kappa_{22} & 0 \\ -K_y & K_y & 0 & 0 \\ \kappa_{11}\frac{b_0 c}{r_0} & 0 & b_0^2 K_x & -b_0^2 K_x \\ 0 & 0 & -b_0^2 K_x & b_0^2 K_x \end{bmatrix}, \tag{12}$$

$$\underline{\underline{\boldsymbol{B}}}_1 = \begin{bmatrix} 0 & 0 & -\frac{1}{2}mv^2 \\ 0 & 0 & -\frac{1}{2}m_c v^2 \\ \kappa_{11}\frac{b_0 c}{r_0} & \kappa_{11} b_0 c & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \underline{\underline{\boldsymbol{B}}}_2 = \begin{bmatrix} \frac{\kappa_{22}}{v} & \frac{\kappa_{22} r_0}{v} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{2}jv \\ 0 & 0 & -\frac{1}{2}j_c v \end{bmatrix} \tag{13}$$

3 KINEMATICS AND INVERSION OF THE SYSTEM

3.1 Measurable kinematical quantities

The resulting system has 4 degrees of freedom and 3 forcing parameters. The whole purpose of this study is to identify the lateral offset irregularity d_3 which is one of the forcing parameters

or inputs to the system from axlebox acceleration measurements. Furthermore, an analysis of the identification of the axle's kinematics shows that the lateral position y and the roll of the axle θ can be deduced respectively from the lateral accelerations and the difference of vertical accelerations. However, identifying the axle's yaw angle φ using the difference in longitudinal accelerations is far more challenging, as its amplitude is very small and easily obscured by noise and sensor movements. In addition, the curvature γ will be considered known along the entire track. The axle's roll θ although not directly appearing in equation 10 can be used since it can be expressed using other parameters if the dynamical deformation of the rail is neglected. It is composed by the roll of the rail θ_d that incudes both the cross-level layout and the irregularity as well as the roll of the axle relative to the rail θ_r.

$$\theta = \theta_d + \theta_r \tag{14}$$

The relative roll θ_r as demonstrated in figure 5 is linked to the relative lateral displacement y_r through the following relation.

$$\theta_r = -\frac{c}{b_0} y_r \tag{15}$$

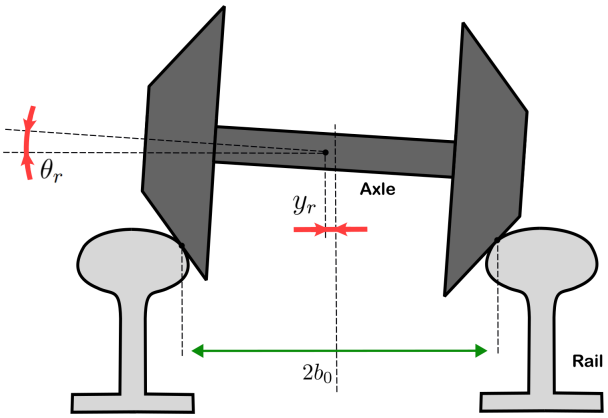


Figure 5: Diagram showing the isolated relative lateral displacement

3.2 System inversion

Let ω be the angular frequency and $\underline{\hat{\boldsymbol{U}}}$ the Fourier transform of $\underline{\boldsymbol{U}}$. With the use of the Fourier transform on equation 10 and with a matrix inversion, it can be written as follows :

$$\underline{\hat{\boldsymbol{U}}} = (\underline{\underline{\boldsymbol{K}}} - \omega^2 \underline{\underline{\boldsymbol{M}}} + i\omega \underline{\underline{\boldsymbol{C}}})^{-1} (\underline{\underline{\boldsymbol{B}}}_1 + i\omega \underline{\underline{\boldsymbol{B}}}_2) \underline{\hat{\boldsymbol{E}}} \tag{16}$$

Extracting the first line from this equation and noting $\underline{\underline{\boldsymbol{A}}} = (\underline{\underline{\boldsymbol{K}}} - \omega^2 \underline{\underline{\boldsymbol{M}}} + i\omega \underline{\underline{\boldsymbol{C}}})^{-1} (\underline{\underline{\boldsymbol{B}}}_1 + i\omega \underline{\underline{\boldsymbol{B}}}_2)$, The lateral position $\hat{y}$ can be written as such :

$$\hat{y} = \underline{\underline{\boldsymbol{A}}}(1,1)\hat{d}_3 + \underline{\underline{\boldsymbol{A}}}(1,2)\hat{\theta}_d + \underline{\underline{\boldsymbol{A}}}(1,3)\hat{\gamma} \tag{17}$$

Using equations 7, 14and 15, θ_d is formulated as :

$$\theta_d = \frac{1}{1 + \frac{r_0 c}{b_0}}\theta + \frac{\frac{c}{b_0}}{1 + \frac{r_0 c}{b_0}} y - \frac{\frac{c}{b_0}}{1 + \frac{r_0 c}{b_0}} d_3 \tag{18}$$

Then, 17 and 18 are combined and $\hat{d_3}$ is isolated. The transfer functions obtained are shown on figure 6.

$$\hat{d_3} = \hat{H}_y.\hat{y} + \hat{H}_\theta.\hat{\theta} + \hat{H}_\gamma.\hat{\gamma} \tag{19}$$

The first thing that stands out is the spikes that both $\hat{H}_y$ and $\hat{H}_\gamma$ reach around the wavelength $\lambda = 40m$. These spikes are linked to a known resonance phenomenon of trains which is the haunting oscillation. Then, one can notice that $\hat{H}_\theta$ has a constant norm and phase of π. This is due to the fact that d_3 and θ_d intervene in the same way in the equations and that $\underline{\underline{\boldsymbol{A}}}(1,1)$ and $\underline{\underline{\boldsymbol{A}}}(1,2)$ are the same up to a scaling factor.

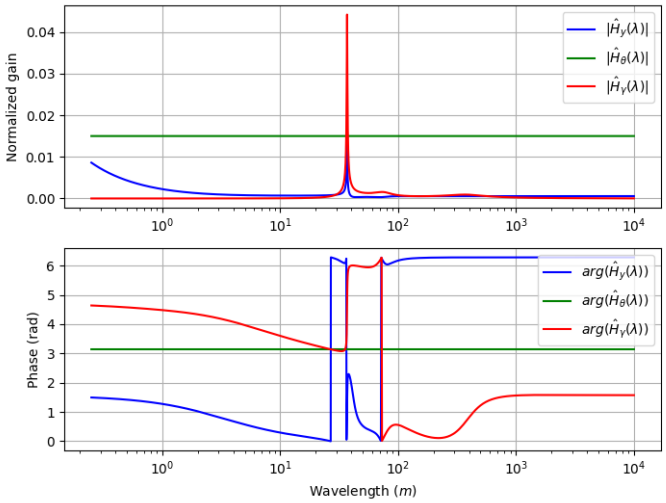


Figure 6: Theoretical transfer functions obtained by inverting the system

With the use of a reverse Fourier transform, the lateral offset irregularity d_3 at each point of the curvilinear abscissa s can be deduced by convolution from the two measured kinematics quantities y and θ as well as the known curvature γ and their corresponding kernels H_y, H_θ and H_γ which are the reverse Fourier transforms of the transfer functions $\hat{H}_y$, $\hat{H}_\theta$ and $\hat{H}_\gamma$. These convolution kernels are displayed on figure 7.

$$d_3 = H_y * y + H_\theta * \theta + H_\gamma * \gamma \tag{20}$$

The obtained kernels are very different from each others. As expected H_θ is a Dirac delta function. H_y and H_γ are however very different. H_y is discontinuous at the $s = 0$ and very limited in space as it gets close to null for $|s| > 50m$. This is the complete opposite for H_γ whose value diminishes very slowly as s decreases. This means that for these theoretical kernels to be used, they need to be very wide.

4 LATERAL TRACK IRREGULARITY IDENTIFICATION ON A NUMERICAL EXPERIMENT

A numerical experience is carried out where the reaction of the surrogate model to a real measured forcing is computed. Measured track irregularities d_2 and d_3 and track designs c_{track} and γ on the french railway network as well as realistic high speed train suspensions parameters, conicity and creep coefficients are employed. Then, the kernels derived from the system

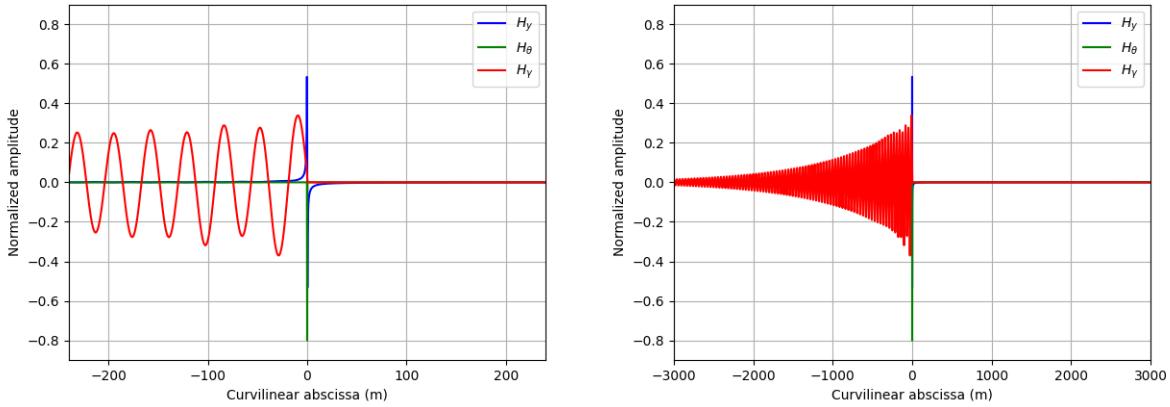


Figure 7: Theoretical kernels obtained by inverting the system

inversion are used to try to estimate the lateral offset irregularity d_3. These kernels were found to give a precise estimation only if their half width l is large enough as the lateral offset identification requires the knowledge of the track curvature γ for a long distance as it appears on 7. For half widths of $300m$ and $1000m$, respective RMS errors of $3.4mm$ and $1.9mm$ are found. An identification error of $\epsilon_{d_3} = 0.25mm$ is only obtained for kernels with a half width of at least $3000m$. These errors are summarized on table 1.
On the other hand, the kernels can also be identified using a linear regression algorithm. For this purpose, ridge regression with a regularization parameter α has been chosen as it gives the most accurate results. let $X = (y, \theta, \gamma)$. The ridge kernels are obtained as follows :

$$H_{ridge} = argmin_H \left\{ \sum_s |d_3(s) - H.X[s-l, s+l]|^2 + \alpha |H|^2 \right\} \tag{21}$$

The kernels given by this method give good estimations of the lateral offset irregularity d_3 with much smaller half widths. with $l = 300m$ and $1000m$ we obtain respective RMS identification errors of $0.1mm$ and $0.03mm$ which should be compared to the RMS value of the lateral offset irregularity in the database used $RMS(d_3) = 2mm$. The ones with a half-width of $l = 1000m$ are shown on 8 and the identification of the lateral offset irregularity is on figure 9. The obtained kernels closely resemble the theoretical ones, but they appear as if noise has been added. In fact, since they are constructed based on statistical correlation, they depend not only on the dynamics equation but also on the characteristics of the given input signals. Unlike the theoretical kernels, they are also adapted to the chosen half-width and with $l = 1000m$ the identification of d_3 is practically perfect.

Kernels used	Theoretical	Theoretical	Theoretical	Ridge regression	Ridge regression
Half width $l(m)$	300	1000	3000	300	1000
RMSE $\epsilon_{d_3}(mm)$	3.4	1.9	0.24	0.10	0.03

Table 1: Summary of the RMS identification errors of d_3

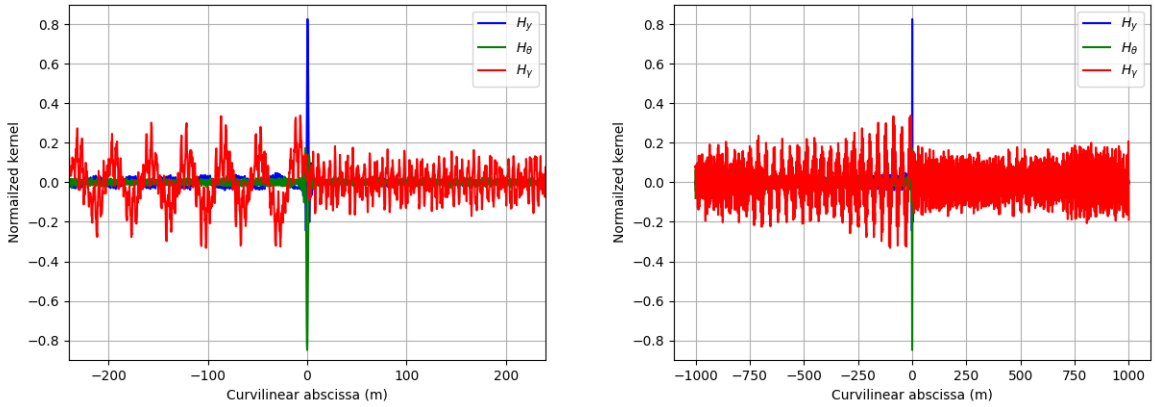


Figure 8: Empirical kernels obtained by Ridge regression

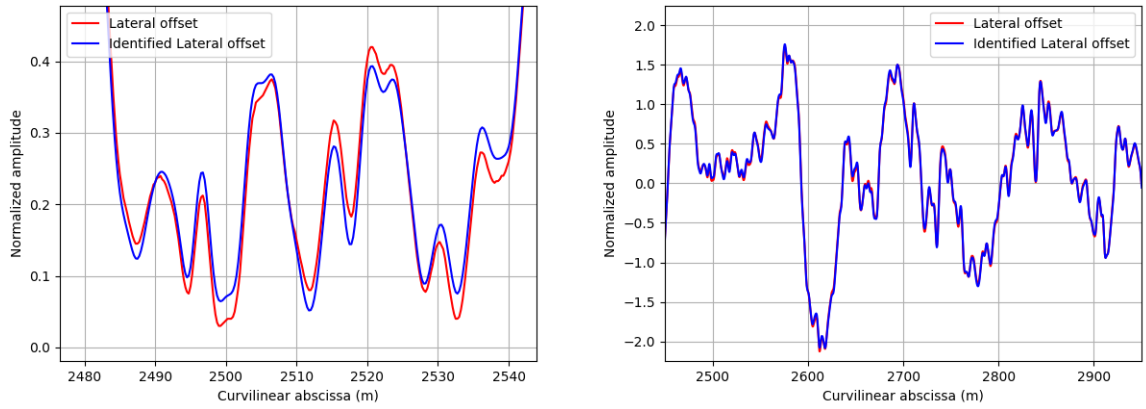


Figure 9: Lateral offset identification using the ridge identified kernel with a half-width of $l = 1000m$

5 CONCLUSIONS

In conclusion, based on the well-known single-axle lateral dynamics model, we have developed a railway dynamics surrogate model that accounts for the presence of a carbody, the track layout, and geometrical irregularities. The resulting system was inverted, and the kinematics of the axle were analyzed, leading to the conclusion that the lateral track irregularity d_3 can be identified solely using the lateral position of the axle y and its roll θ, both of which can be reconstructed from axle-box accelerations, along with the known curvature of the track γ. Subsequently, the use of these theoretically identified kernels was compared to kernels derived through linear regression. The next step in this work is to study the stability and sensitivity of the system. Following this, realistic non-linearities will be introduced incrementally, and their impact on the identification of the lateral offset irregularity will be analyzed.

REFERENCES

[1] X. Wei, F. Liu, and L. Jia, "Urban rail track condition monitoring based on in-service vehicle acceleration measurements," *Measurement*, vol. 80, pp. 217–228, 2016.

[2] H. Tsunashima, H. Ono, T. Takata, and S. Ogata, "Development and operation of track condition monitoring system using in-service train," *Applied Sciences*, vol. 13, no. 6, 2023.

[3] H. Tsunashima, Y. Naganuma, and T. Kobayashi, "Track geometry estimation from car-body vibration," *Vehicle System Dynamics*, vol. 52, pp. 207–219, 05 2014.

[4] J. S. Lee, S. Choi, S.-S. Kim, C. Park, and Y. G. Kim, "A mixed filtering approach for track condition monitoring using accelerometers on the axle box and bogie," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 3, pp. 749–758, 2012.

[5] E. Bongini and C. Funfschilling, "Procédé et installation d'analyse automatique de l'évolution de la géométrie de voies de circulation," July 2017.

[6] S. Iwnicki, *Handbook of Railway Vehicle Dynamics*. CRC Press, 2006.

[7] V. K. Garg and R. V. Dukkipati, *Dynamics of Railway Vehicle Systems*. Academic Press, 1984.

[8] K. Knothe and S. Stichel, *Rail Vehicle Dynamics*. 01 2017.

[9] A. H. Wickens, "The dynamics of railway vehicles—from stephenson to carter," *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, vol. 212, no. 3, pp. 209–217, 1998.

UNCECOMP 2025

Proceedings of the
6th International Conference on
Uncertainty Quantification in Computational Science and Engineering

M. Papadrakakis, V. Papadopoulos, G. Stefanou (Eds.)

First Edition, November 2025

ISBN: **978-618-5827-03-8**

Institute of Structural Analysis and Antiseismic Research
National Technical University of Athens, Greece