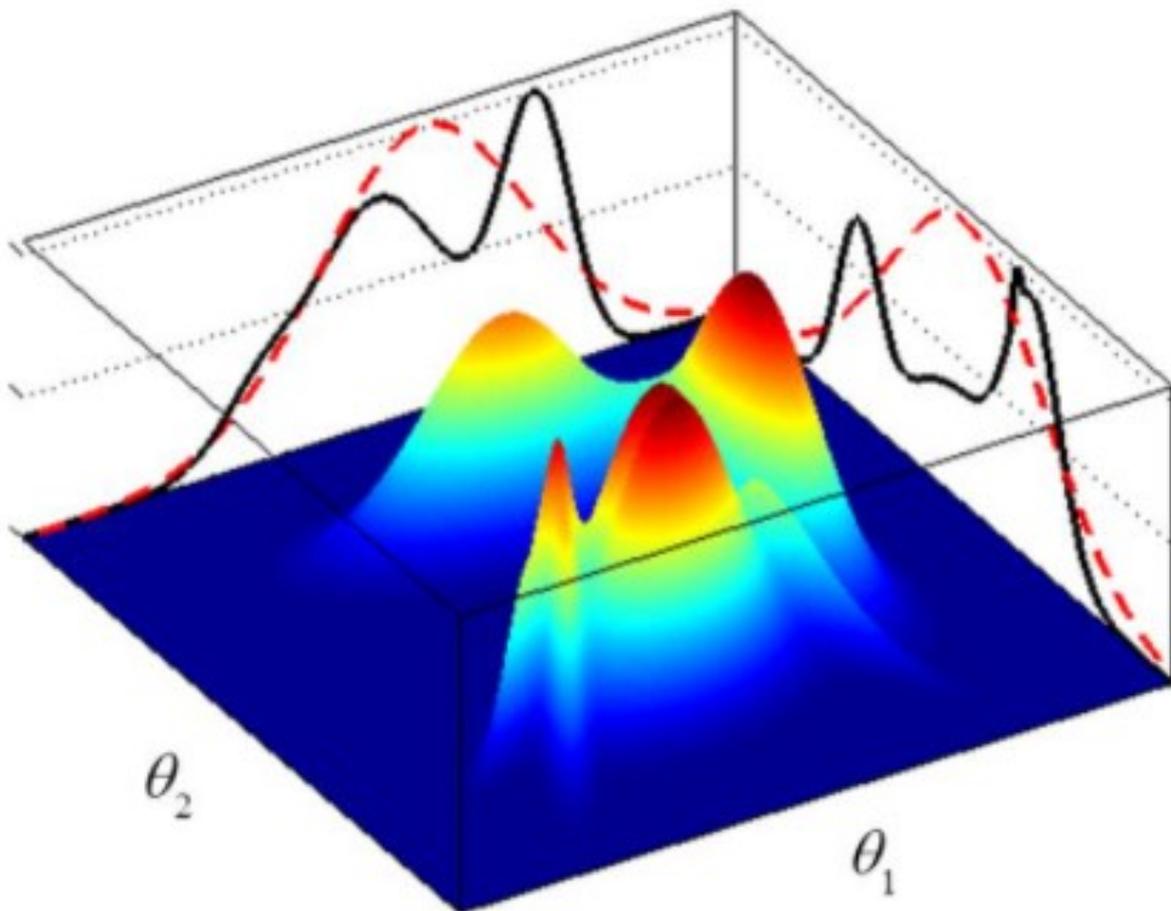


UNCECOMP 2021

*4th International Conference
on Uncertainty Quantification in Computational
Sciences and Engineering*

PROCEEDINGS

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



UNCECOMP 2021

Uncertainty Quantification in Computational Sciences and Engineering

Proceedings of the 4th International Conference on Uncertainty
Quantification in Computational Sciences and Engineering
Streamed from Athens, Greece
28-30 June 2021

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 2021

Uncertainty Quantification in Computational Sciences and Engineering

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

First Edition, September 2021

© The authors

ISBN: **978-618-85072-6-5**

PREFACE

This volume contains the full-length papers presented in the International Conference on Uncertainty Quantification in Computational Sciences and Engineering (UNCECOMP 2021) that was streamed from Athens, Greece on June 28-30, 2021.

UNCECOMP 2021 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 2021 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.

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 “4th International Conference on Uncertainty Quantification in Computational Sciences and Engineering”, to the following organizations:

- European Community on Computational Methods in Applied Sciences (ECCOMAS)
- 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 Speakers: George Karniadakis, Petros Koumoutsakos, Bruno Sudret

Semi-Plenary Speakers: Alireza Doostan, Hector Jensen, Chao Jiang, David Moens, Fabio Nobile, Raul Tempone

MS Organizers: Alberto Figueroa Alvarez, George Arampatzis, Miguel Bessa, Jean-Marc Bourinet, Eleni Chatzi, Manolis Chatzis, Alice Cicirello, Luca Dede, Vasilis Dertimanis, Matthias Faes, Krishna Garikipati, Roger Ghanem, Dimitrios G. Giovanis, Wolfgang Graf, Michael Hanss, Dionissios Hristopoulos, Michael Kaliske, Evangelia Kalligiannaki, Ioannis Kalogeris, Petros Koumoutsakos, Sigrid Leyendecker, Alexander Litvinenko, Geert Lombaert, Eliz-Mari Lourens, Sankaran Mahadevan, Andrea Manzoni, Stefano Marelli, David Moens, Costas Papadimitriou, Vissarion Papadopoulos, Edoardo Patelli, Paris Perdikaris, Dirk Pflüger, Marco Pingaro, Dmytro Pivovarov, Stefanie Reese, Bojana Rosic, Dimitrios Savvas, Alba Sofi, Christian Soize, George Stefanou, Paul Steinmann, James Stewart, Bruno Sudret, Alexandros Taflanidis, Patrizia Trovalusci, Ivi Tsantili

SUMMARY

Preface.....	iii
Acknowledgements.....	iv
Contents.....	vi

Minisymposia

MS 1: UNCERTAINTY QUANTIFICATION IN VIBRATION BASED MONITORING AND STRUCTURAL DYNAMICS SIMULATIONS	1
<i>Organized by Eleni Chatzi, Manolis Chatzis, Vasilis Dertimanis, Geert Lombaert, Costas Papadimitriou</i>	
MS 2: UNCERTAINTY QUANTIFICATION UNDER LIMITED DATA	26
<i>Organized by Michael Hanss, David Moens, Matthias Faes, Edoardo Patelli, Alba Sofi</i>	
MS 3: NON-DETERMINISTIC COMPUTATIONAL HOMOGENIZATION OF HETEROGENEOUS MATERIALS	100
<i>Organized by Paul Steinmann, Dmytro Pivovarov</i>	
MS 4: ADVANCES IN DATA-DRIVEN MODELING AND APPLICATIONS	112
<i>Organized by Ivi Tsantili, Evangelia Kalligiannaki, Dionissios Hristopoulos</i>	
MS 8: DATA-DRIVEN UNCERTAINTY QUANTIFICATION AND DATA ASSIMILATION USING MANIFOLD LEARNING WITH SPARSE AND LOW-RANK REPRESENTATIONS	129
<i>Organized by Dimitrios G. Giovanis, Alexander Litvinenko, Bojana Rosic</i>	
MS 11: SENSORS PLACEMENT UNDER UNCERTAIN INFORMATION	145
<i>Organized by Eliz-Mari Lourens, Alice Cicirello</i>	
MS 12: SOFTWARE FOR UNCERTAINTY QUANTIFICATION AND METHODS FOR IMPROVING THE EFFICIENCY OF MONTE CARLO SIMULATION	194
<i>Organized by Stefano Marelli, Edoardo Patelli, Dirk Pflüger</i>	

Thematic Sessions

TS 21: UNCERTAINTY QUANTIFICATION, SYSTEM RELIABILITY ANALYSIS AND RISK ASSESSMENT	232
---	-----

CONTENTS

Minisymposia

MS 1: UNCERTAINTY QUANTIFICATION IN VIBRATION BASED MONITORING AND STRUCTURAL DYNAMICS SIMULATIONS

NONLINEAR GAUSSIAN PROCESS LATENT FORCE MODELS FOR INPUT ESTIMATION IN HYSTERETIC SYSTEMS 1
Timothy J. Rogers, Joe D. Longbottom, Keith Worden, Elizabeth J. Cross

INFLUENCE OF DIFFERENT FULLY NON-STATIONARY ARTIFICIAL TIME HISTORIES GENERATION METHODS
ON THE SEISMIC RESPONSE OF FREQUENCY-DEPENDENT STRUCTURES 15
Federica Genovesi, Giuseppe Muscolino, Alessandro Palmeri

MS 2: UNCERTAINTY QUANTIFICATION UNDER LIMITED DATA

LOW-COMPLEXITY ZONOTOPES CAN ENHANCE UNCERTAINTY QUANTIFICATION (UQ) 26
Olga Kosheleva, Vladik Kreinovich

BOUNDS OF RELIABILITY FUNCTION FOR STRUCTURAL SYSTEMS SUBJECTED TO A SET OF RECORDED
ACCELEROGRAMS 35
Federica Genovesi, Giuseppe Muscolino, Alba Sofi

STRUCTURAL RELIABILITY ESTIMATION OF STEEL MAST EXHIBITING RANDOM MECHANICAL AND
ENVIRONMENTAL PARAMETERS 49
Rafał Bredow, Marcin Kamiński

INVERSE PROBLEMS FOR STOCHASTIC NEUTRONICS 63
Corentin Houpert, Josselin Garnier, Philippe Humbert

MACHINE LEARNING AIDED STOCHASTIC SLOPE STABILITY ANALYSIS 75
Zhanpeng Liu, Di Wu, Daichao Sheng, Behzad Fatahi, Hadi Khabbaz

LIMIT REPRESENTATIONS OF IMPRECISE RANDOM FIELDS 82
Mona M. Dannert, Johannes L. Häufler, Udo Nackenhorst

MS 3: NON-DETERMINISTIC COMPUTATIONAL HOMOGENIZATION OF HETEROGENEOUS MATERIALS

NUMERICAL SIMULATION FOR 3D PRINTED WALL STRUCTURE DURING THE PROCESS OF PRINTING
CONSIDERING UNCERTAINTY 100
Meron Wondafrash, Albrecht Schmidt, Luise Göbel, Tom Lahmer, Carsten Könke

MS 4: ADVANCES IN DATA-DRIVEN MODELING AND APPLICATIONS

EFFICIENT DISCRIMINATION BETWEEN BIOLOGICAL POPULATIONS VIA NEURAL-BASED ESTIMATION OF RÉNYI DIVERGENCE	112
<i>Anastasios Tsourtis, Georgios Papoutsoglou, Yannis Pantazis</i>	

MS 8: DATA-DRIVEN UNCERTAINTY QUANTIFICATION AND DATA ASSIMILATION USING MANIFOLD LEARNING WITH SPARSE AND LOW-RANK REPRESENTATIONS

IDENTIFICATION OF UNKNOWN PARAMETERS AND PREDICTION WITH HIERARCHICAL MATRICES	129
<i>Alexander Litvinenko, Ronald Kriemann, Vladimir Berikov</i>	

MS 11: SENSORS PLACEMENT UNDER UNCERTAIN INFORMATION

OPTIMAL SELECTION OF BAYESIAN VIRTUAL SENSORS FOR DAMAGE DETECTION UNDER VARIABLE ENVIRONMENTAL CONDITIONS	145
<i>Jyrki Kullaa</i>	

SENSOR FAULT IDENTIFICATION FOR ROBUST STRUCTURAL HEALTH MONITORING	159
<i>Andreea-Maria Oncescu, Alice Cicirello</i>	

ON THE INVESTIGATION OF THE EFFECT OF POPULATION UNCERTAINTY ON OPTIMAL SENSOR LOCATIONS	168
<i>Felipe Igea, Manolis N. Chatzis, Alice Cicirello</i>	

OPTIMAL SENSOR PLACEMENT IN DISTRICT HEATING NETWORKS FOR BAYESIAN INFERENCE OF UNCERTAIN DEMANDS	178
<i>Alexander Matej, Andreas Bott, Lea Rehlich, Florian Steinke, Stefan Ulbrich</i>	

MS 12: SOFTWARE FOR UNCERTAINTY QUANTIFICATION AND METHODS FOR IMPROVING THE EFFICIENCY OF MONTE CARLO SIMULATION

IMPROVING THE RATE OF CONVERGENCE OF THE QUASI-MONTE CARLO METHOD IN ESTIMATING EXPECTATIONS ON A GEOTECHNICAL SLOPE STABILITY PROBLEM	194
<i>Philippe Blondeel, Pieterjan Robbe, Dirk Nuyens, Geert Lombaert, Stefan Vandewalle</i>	

UNCERTAINTY QUANTIFICATION IN THE CLOUD WITH UQCLOUD	209
<i>Christos Lataniotis, Stefano Marelli, Bruno Sudret</i>	

OTBENCHMARK: AN OPEN SOURCE PYTHON PACKAGE FOR BENCHMARKING AND VALIDATING UNCERTAINTY QUANTIFICATION ALGORITHMS	218
<i>Elias Fekhari, Michaël Baudin, Vincent Chabridon, Youssef Jebroun</i>	

Thematic Sessions

TS 21: UNCERTAINTY QUANTIFICATION, SYSTEM RELIABILITY ANALYSIS AND RISK ASSESSMENT

ANALYTICAL MODEL FOR FRACTURE IN RANDOM QUASIBRITTLE MEDIA BASED ON EXTREMES OF THE AVERAGING PROCESS	232
<i>Miroslav Vořechovský</i>	
A SEQUENTIAL MULTI-POINT SAMPLING PROCEDURE FOR SURROGATE MODELS	246
<i>Matthias Fischer, Carsten Proppe</i>	
CALIBRATION OF MATERIAL MODEL PARAMETERS USING MIXED-EFFECTS MODEL	258
<i>Clément Laboulfie, Matthieu Balesdent, Loïc Brevault, Sébastien Da Veiga, François-Xavier Irisarri, Rodolphe Le Riche, Jean-François Maire</i>	
ADAPTIVE SEQUENTIAL SAMPLING FOR POLYNOMIAL CHAOS EXPANSION	296
<i>Lukáš Novák, Miroslav Vořechovský, Václav Sadílek</i>	
AN EXPERIMENTAL STUDY OF VARIABILITY IN DAMPING, FREQUENCY RESPONSE AND MODAL DATA	302
<i>Asish Kumar Panda, Subodh V. Modak</i>	
EFFECTIVENESS OF THE PROBABILITY DENSITY EVOLUTION METHOD FOR DYNAMIC AND RELIABILITY ANALYSES OF MASONRY STRUCTURES	313
<i>Massimiliano Lucchesi, Barbara Pintucchi, Nicola Zani</i>	
FEM SHAKEDOWN ANALYSIS OF KIRCHOFF-LOVE PLATES UNDER UNCERTAINTY OF STRENGTH	323
<i>Ngọc Trình Trần, Manfred Staat</i>	
LINEAR ALGEBRA OF LINEAR AND NONLINEAR BAYESIAN CALIBRATION	339
<i>Michaël Baudin, Régis Lebrun</i>	
ASSESSMENT OF VARIANTS OF THE METHOD OF MOMENTS AND POLYNOMIAL CHAOS APPROACHES TO AERODYNAMIC UNCERTAINTY QUANTIFICATION	354
<i>Evangelos Papoutsis-Kiachagias, Varvara Asouti, Kyriakos Giannakoglou</i>	
SOFTWARE FOR UNCERTAINTY PROPOGATION AND RELIABILITY ASSESSMENT OF INELASTIC WIND EXCITED SYSTEMS	371
<i>Wei-Chu Chuang, Seymour MJ Spence</i>	
UNCERTAINTY QUANTIFICATION FOR DEEP LEARNING REGRESSION MODELS IN THE LOW DATA LIMIT	379
<i>Cristina Garcia-Cardona, Yen Ting Lin, Tanmoy Bhattacharya</i>	

NONLINEAR GAUSSIAN PROCESS LATENT FORCE MODELS FOR INPUT ESTIMATION IN HYSTERETIC SYSTEMS

T.J. Rogers¹, J. D. Longbottom¹, K. Worden¹ & E. J. Cross¹

¹ Dynamics Research Group
Department of Mechanical Engineering, University of Sheffield
Mappin Street, Sheffield
S1 3JD, UK

e-mail: {tim.rogers, jdlongbottom1, k.worden, e.j.cross}@sheffield.ac.uk

Keywords: Bayesian methods, Particle filtering, Monte Carlo methods, Grey-box modelling.

Abstract. *One outstanding challenge in structural dynamics is lack of access to measurements while systems are in operation. For example, this could be wind or wave loads on offshore structures, contact forces between a vehicle's wheels and a roadway, the action of an earthquake on a tall building. In view of this, one specific and important problem is to infer both the unknown inputs to a dynamic system and its internal states. The difficulty of this task is linked to the inherent uncertainty in the system, which has two key sources. The first issue is the presence of noise on the measurements or uncertainty in the dynamics of the system (process noise); the second, is that it may not be possible to define a priori a functional form for the loading signal. The Gaussian process latent force model is a tool to address both of these tasks, it couples a known dynamic system with a flexible non-parametric representation of the unknown forcing. This representation is chosen to be a Gaussian process in time, which is used to estimate the distribution of possible functions which could have been the unmeasured inputs to the system. One key benefit of this approach is that it provides closed-form expressions for the process noise in the system, based on the characteristics of this function. This paper extends the range of systems for which this type of model can be applied to those exhibiting hysteresis. These systems represent a significant increase in difficulty; not only do they introduce a nonlinearity into the system parameters, but this nonlinearity requires an additional hidden state. It will be shown in this paper how a particle Gibbs approach allows Bayesian inference over the intractable state-space model that describes these systems. This approach allows joint input-state inference to be performed even in these highly nonlinear systems.*

1 INTRODUCTION

This paper will address the problem where output measurements from a nonlinear dynamic system are available, from which the practitioner would like to infer the internal (hidden) states of that system and its unmeasured inputs. A motivating example may be the situation where a suspension system for a vehicle is identified in a laboratory environment, then in use, one wishes to investigate the displacement and velocity of that dynamic element and the unmeasured excitation from the road, based on measurements of the acceleration in the vehicle. One important class of systems for which input estimation is particularly difficult is *hysteretic* systems. Hysteresis, informally, introduces a memory effect into the system which can greatly complicate identification, including input-state estimation. The contribution of this work is to extend previous work on nonlinear input-state estimation [1] to the case of a hysteretic nonlinearity in order to demonstrate the general and powerful approach of nonlinear Gaussian process latent force modelling.

One sensible framework for attempting the challenge of input (load) estimation is that of a Bayesian state-space model (SSM) [2]. If measurements of the input to the system are available, recovering the hidden states of the system is achieved by inferring the filtering or smoothing distribution of the model. However, in the situation being considered here, these inputs are unknown. It would be possible to assume that the system was excited with a white-noise; however, this is often not true. It is necessary then to also model the unknown inputs to the system with a view to inferring them. The challenge becomes that of making an appropriate choice of tool to model these unknown inputs and how this may be incorporated into the model of the dynamical system.

It is desirable that the model chosen to represent the forcing should be flexible enough to accommodate the diversity of functions which may represent these unknown inputs. For this reason, a white-noise assumption on the input may be too restrictive. Instead, it may be appropriate to describe some of the characteristics of the function which may have generated the forcing, for instance how many times differentiable it is. Such a tool exists in the Gaussian process (GP), a flexible Bayesian nonparametric regression model (for an introduction see [3] or [4]). Given the existence of such an approach, a system can be imagined which has some defined dynamical structure where the input to that system is modelled as a GP in time which provides a richer class of possible forcing signals.

Alvarez et al. [5] proposed such an approach for linear systems which could be modelled as a dynamical system forced by a GP; they term this the *Latent Force Model*. By introducing this first or second-order dynamic component to the data-modelling process, they increased the range of scenarios which could be effectively modelled using the Gaussian process. However, the implementation and training of such models can be difficult and time consuming. A significant reduction in this complexity can be achieved given work by Hartikainen and Särkkä [6], which shows that a temporal GP can be written as an equivalent linear Gaussian state-space model (LGSSM) and an identical solution to the full GP is recovered by application of a Kalman filter and Rauch-Tung-Striebel (RTS) smoother. It is then possible to transform the GP latent force model of Alvarez et al. [5] into an LGSSM, where the states represent the dynamic system augmented by one or more additional states which are equivalent to the GP input to the system. This link was noted in [7]; however, one limitation remains — the dynamic model is restricted to linear systems. This paper will present a methodology for removing this limitation such that a nonlinear latent force model can be learnt, also within the state-space framework. In particular, the extension to more difficult hysteretic nonlinearities is attempted. The main

question being investigated is: "Does the introduction of additional dynamic states, because of hysteresis, impede joint input-state estimation with nonlinear Gaussian process latent force models?"

2 NONLINEAR LATENT FORCE MODELS

Beginning with the case of a linear latent-force model, for a second-order dynamic system, the model being considered is,

$$M_s \ddot{\mathbf{q}} + C_s \dot{\mathbf{q}} + K_s \mathbf{q} = U, \quad U \sim \mathcal{GP}(0, k(t, t')) \quad (1)$$

where M_s , C_s , and K_s are the mass, damping and stiffness matrices of a second order system which is forced by an unknown input U ; the subscript s is used to indicate that these are the system matrices. The unknown input U is modelled as a Gaussian process with a zero mean and a covariance kernel (function) in time, $k(t, t')$. Defining the states of the model $\mathbf{x} = [\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u}]^T$, with \mathbf{u} being the augmented states which correspond to the GP, this model has an equivalent continuous-time state-space representation,

$$\dot{\mathbf{x}}(t) = F\mathbf{x}(t) + v(t) \quad (2)$$

$$\mathbf{y}(t) = H\mathbf{x}(t) + w(t) \quad (3)$$

with H being the matrix which defines the observation of the states; for example, in the case of observing the acceleration of the system, $\ddot{\mathbf{q}}$, $H = [-M_s^{-1}K_s, -M_s^{-1}C_s, M_s^{-1}, 0]$. This observation is subject to white noise $w(t)$ on the measurements. The dynamics of the process are captured in the matrix F which can be considered to be made up of four block matrices,

$$F = \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix} = \begin{bmatrix} 0 & \mathbb{I} & \vdots & 0 \\ -M_s^{-1}K_s & -M_s^{-1}C_s & \vdots & M_s^{-1} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \vdots & F_{GP} \end{bmatrix} \quad (4)$$

In the top left, F_{11} is a block of the matrix which corresponds to the linear second-order dynamics of the system; in the top right, F_{12} relates the forcing states \mathbf{u} to this dynamical system. The F_{21} block is zero, as the dynamics are assumed not to affect the forcing evolution in time, i.e. the applied forces are independent of the dynamics. Finally, the matrix F_{22} contains the state-space representation of the Gaussian process in time. How this matrix F_{22} is formed will be briefly reviewed now and it will be seen how it, along with the process noise $v(t)$ is fully defined by the GP over U .

Following [6], for a GP with a stationary covariance function $k(t, t')$, it is possible to consider the power spectral density of that covariance. Taking the popular Matérn 3/2 kernel as an example, with $r = |t - t'|$, one has,

$$k(r) = \sigma_f^2 \left(1 + \frac{\sqrt{3}r}{\ell}\right) \exp\left\{-\frac{\sqrt{3}r}{\ell}\right\} \quad (5)$$

which is governed by two *hyperparameters*; the length scale ℓ and the signal variance σ_f^2 . Taking the Fourier transform gives the spectral density of the process as,

$$S(\omega) = 4\sigma_f^2 \lambda^3 (\lambda^2 + \omega^2)^{-2} \quad (6)$$

where $\lambda = \sqrt{3}/\ell$. Observing that this is a rational function with a denominator which is a polynomial in ω^2 , Hartikainen and Särkkä [6] apply a spectral factorisation on this density to show

that this is equivalent to a stochastic differential equation (SDE) of order p , if the denominator is a polynomial of order $(\omega^2)^p$. More intuitively, the mode can be seen as an ordinary differential equation (ODE) of order p forced by a white noise. Interestingly, this formulation links back to the interpretation of a Gaussian process as a linear filter on a continuous white noise sequence. For the Matérn 3/2 kernel being used as an example, this gives the SDE,

$$\frac{d^2u}{dt^2} - 2\lambda \frac{du}{dt} - \lambda^2 = \nu(t) \quad (7)$$

where $\nu(t)$ is a white noise process with spectral density q ,

$$q = \frac{12\sqrt{3}\sigma_f^2}{\ell^3} \quad (8)$$

It should be noted at this point, that this formulation gives the state-space form for a single output GP; however, it is trivial to extend this to a number of independent GPs acting on the different degrees of freedom in equation (1).

Therefore, for a single-degree-of-freedom harmonic oscillator forced by a GP with a Matérn 3/2 covariance function, the equation of motion,

$$m\ddot{q} + c\dot{q} + kq = u, \quad u \sim \mathcal{GP}(0, k(t, t')) \quad (9)$$

has an equivalent state-space form,

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -k/m & -c/m & 1/m & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -2\lambda & -\lambda^2 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \nu(t) \quad (10)$$

$$\mathbf{y}(t) = [-k/m \quad -c/m \quad 1/m \quad 0] \mathbf{x}(t) + w(t) \quad (11)$$

when one is observing acceleration and if the augmented state vector is $\mathbf{x} = [q, \dot{q}, u, \dot{u}]$. The solution to this system can then be found by discretising and applying the Kalman filter and RTS smoother, since this is a LGSSM. For examples of the linear case, see [8], where inference over the model is extended to include the hidden states of the oscillator, the unknown input U and the parameters of the model M_s, C_s, K_s .

This paper however, is concerned with a more general case where the ODE which describes the dynamic system is nonlinear. Considering a single-degree-of-freedom system, this means inferring the displacement and velocity of an oscillator with an unknown input which can be expressed as,

$$m\ddot{q} + f(q, \dot{q}) = u, \quad u \sim \mathcal{GP}(0, k(t, t')) \quad (12)$$

under the same assumption that the unmeasured inputs can be modelled as a GP in time with zero mean and some stationary covariance $k(t, t')$. Here $f(q, \dot{q})$ is some function of the displacement and velocity of the oscillator, it is possible to recover the linear case as a subset of this model by setting $f(q, \dot{q}) = c\dot{q} + kq$.

It will now be seen how a similar procedure as for the linear case can be followed to develop a *nonlinear latent force model*. This approach is discussed in further detail in [8] where a similar methodology is applied to a Duffing oscillator. Inspecting the structure of the SSM in equations (10) and (11), it is clear that the state vector \mathbf{x} can be viewed in two parts. The states related to

the “physical” dynamical system (states related to q and \dot{q}) and the augmented states which are the GP modelling the forcing (in the case of the Matérn 3/2 kernel, u and \dot{u}). This interpretation gives some indication of how a nonlinear equivalent may be structured. The change in the system equation to include nonlinear dynamics will result in the system states related to q and its derivatives becoming nonlinear; however, the representation of the GP remains unchanged.

It is now possible to construct a nonlinear SSM with a similar structure to that seen in the linear case,

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} \dot{q} \\ \frac{u}{m} - \frac{1}{m}f(q, \dot{q}) \\ \dot{u} \\ -2\lambda u - \lambda^2 \dot{u} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \nu(t) \quad (13)$$

$$y(t) = \frac{u}{m} - \frac{1}{m}f(q, \dot{q}) + w(t) \quad (14)$$

This form covers many commonly-encountered nonlinearities; however, for certain types — notably hysteretic systems — additional states also need to be included in the model. For the purposes of this paper, the important point is that to apply the GP latent force approach, it is possible to augment the state vector of the system with a number of states which are a direct equivalent to assuming a GP in time as the input signal. The challenge then is to perform inference over this extended state-space model, to recover the smoothing distribution; in the vast majority of cases, this distribution will not be available in closed form and some numerical estimation must be employed. As a point of interest that will not be covered in any more depth, it would also be possible to accommodate inputs to the system which appear in a nonlinear manner as opposed to a simple additive forcing.

2.1 Inference

While the modification to the linear version of the GP latent force model may seem minor, it unfortunately severely complicates the inference procedure. Remembering that the quantities of interest are the hidden states of the model \mathbf{x} , which include the internal states of the oscillator and those related to the GP, the task to be solved (from a Bayesian filtering perspective) is one of inferring the smoothing distribution of this state-space model. That is to recover the distribution¹ $p(\mathbf{x}_{1:T}|\mathbf{y}_{1:T})$. Since the model is no longer linear with Gaussian noise, it is not (generally) possible to recover this distribution exactly. Instead some approximation must be made.

The problem being addressed is one of estimating a high-dimensional intractable posterior distribution, which is the smoothing distribution of the nonlinear filter. This challenge will be addressed by the use of a particle filter. The particle filter can be used to form an efficient approximation to the filtering distribution of a nonlinear state-space model in a Monte Carlo manner. A number of weighted point masses (or particles) propagate through time and by repeated proposal, weighting and resampling, they form an importance sampling approximation to the filtering distribution at every time step; an introduction is given in Doucet and Johansen [9]. In general, algorithms of this type are referred to as *Sequential Monte Carlo* (SMC) methods and their applicability has been shown to extend beyond inference of the filtering distributions

¹The notation $a : b$ is adopted to indicate the discrete index from time a to time b inclusively, e.g. $\mathbf{x}_{1:T}$ would denote the state vector \mathbf{x} at all time points from step 1 to step T .

of nonlinear state-space models; for example, Andersson Naesseth et al. [10] show how this approach can be applied to probabilistic graphical models.

However, in this case, the object of interest is the smoothing distribution of the state-space model. A naïve approach to determining this distribution would be to apply a Markov Chain Monte Carlo (MCMC) approach; there are two major limitations to this, firstly, the likelihood of the system is not available in closed form. Secondly, the smoothing distribution can be very high-dimensional and it becomes far harder to define efficient proposals within standard MCMC schemes such as the Metropolis-Hastings algorithm. Andrieu et al. [11] show how the SMC approach can be combined with an MCMC framework, to allow more efficient inference over high dimensional distributions encountered in nonlinear state-space models. By showing that the estimate of the marginal likelihood of the filter provided by the SMC algorithm $\hat{p}_\theta(\mathbf{x}_{1:T}|\mathbf{y}_{1:T})$ can be used within a Metropolis-Hastings step, they create a valid MCMC algorithm. It is also shown that it is possible to construct an equivalent to the Gibbs sampling procedure, an approach which will be used in this paper.

In the case considered here, it will be possible to employ a slightly more sophisticated approach, where multiple variables are sampled at once, specifically sampling all of the states together $\mathbf{x}_{1:T}$. A form of particle filter over the nonlinear state-space model will be able to provide an efficient proposal for the high-dimensional distribution being considered; in fact, it will be possible to ensure that the proposal generates a valid sample from the smoothing distribution on every iteration. In order to ensure the ergodicity of the Markov kernel which samples these states they are conditioned upon the previous sample, this gives rise to the *Conditional Particle Filter* (CPF) [12]. In a CPF, one state trajectory is fixed *a priori* which is included in the weighting and resampling steps. Conceptually, one could think of this as “anchoring” the particle filter to the previously-sampled state trajectory and ensuring it does not drift too far in a single iteration.

One challenge encountered when learning the smoothing distribution in an MCMC manner with a CPF is poor mixing in the chain because of path degeneracy, a phenomenon where resampling can mean particles all share a few common ancestors. This poor mixing more acutely affects samples of the states close to the beginning of the time series. To overcome this shortcoming a simple yet powerful modification to the CPF approach was made by Lindsten et al. [13]. The contribution of that work is to include an *ancestor sampling* step in the CPF algorithm. At every point in time a new ancestor for the reference trajectory is sampled. Before presenting this algorithmically, it is worth considering what is being asked in the ancestor sampling step: “Given the location of the reference trajectory at time t , which of the particles at time $t - 1$ could have generated this sample?”. This possible ancestor for the reference trajectory at time t is then sampled based on those proposal probabilities.

The algorithm for inferring the smoothing distribution using an MCMC approach incorporating the CPF with ancestor sample will now be given, a more thorough review and comparison with an alternative method can be found in [12]. It should also be noted that, should the (hyper)parameters of the system need to be inferred this can be done as part of a full Particle Gibbs with Ancestor Sampling [13] scheme; an example of this approach in the context of a GP latent force model can be found in [1].

The algorithm for sampling P state trajectories using this approach can be found in Algorithm 1. Starting from some initial trajectory $X[0]$, each trajectory is drawn conditioned on the previous sample $X[p - 1]$ by means of a CPF with ancestor sampling. The CPF runs as in the literature, a particle system is proposed from a prior $q(\mathbf{x}_1)$, except the final particle which is assigned the reference trajectory value; all particles are assigned equal weights. These particles are then weighted by some operation $W_{\theta,1}(\mathbf{x}_1^i)$; in the case of the bootstrap filter, this is the

Algorithm 1 MCMC Smoothing From a Conditional Particle Filter with Ancestor Sampling

```

1: Set  $X[0]$  ▷ Initial reference trajectory.
2: for  $p=1, \dots, P$  do
3:   Sample  $\mathbf{x}_1^i \sim q(\mathbf{x}_1)$  for  $i = 1, \dots, N - 1$ 
4:   Set  $\mathbf{x}_1^N = X[p - 1]$ 
5:   Calculate  $w_1^i = W_{\theta,1}(\mathbf{x}_1^i)$  for  $i = 1, \dots, N$ 
6:   for  $t = 2, \dots, T$  do
7:     Sample  $\{a_t^i, \mathbf{x}_t^i\}_{i=1}^N \sim M_{\theta,t}(a_t, \mathbf{x}_t)$ 
8:     Calculate  $\{\tilde{w}_{t-1|T}^i\}_{i=1}^N$ 
9:     Sample  $a_t^N$  with  $\mathbb{P}(a_t^N = i) \propto \tilde{w}_{t-1|T}^i$ 
10:    Set  $\mathbf{x}_t^N = \mathbf{x}_t'$ 
11:    Set  $\mathbf{x}_{1:t}^i \leftarrow (\mathbf{x}_{1:t-1}^{a_t^i}, \mathbf{x}_t^i)$  for  $i = 1, \dots, N$ 
12:    Calculate  $w_t^i = W_{\theta,t}(x_{1:t}^i)$  for  $i = 1, \dots, N$ 
13:   end for
14:   Draw  $k$  with  $\mathbb{P}(k = 1) \propto w_T^i$ 
15:   return  $X[p] = \mathbf{x}_{1:T}^{(k)}$ 
16: end for

```

observation likelihood. Then, sequentially for every time point, the whole particle system is moved from $t - 1$ to t by means of a move kernel $M_{\theta,t}(a_t, \mathbf{x}_t)$ which describes the resampling and proposal steps. At this point the ancestor sampling operation takes place, the ancestral weights of the reference $\{\tilde{w}_{t-1|T}^i\}_{i=1}^N$, are given by $\tilde{w}_{t-1|T}^i = p_\theta(\mathbf{x}_t^N | \mathbf{x}_{t-1}^i)$ the probability of the value of the reference trajectory at time t given all the particles (including the reference) at time $t - 1$. The ancestor for this point on the reference trajectory at time t can then be sampled with probability proportional to these ancestral weights. The N^{th} particle at this time step t is then replaced with the reference, and the ancestral paths of the particle system are updated. Finally, the weights of the particles at time t are calculated, $W_{\theta,t}(x_{1:t}^i)$. This pattern continues up to the end of the available time data, i.e. for $t = 1, \dots, T$. Once the CPF has completed this forward pass, a path can be sampled with probability proportional to the weights at the final time step, $\mathbb{P}(k = 1) \propto w_T^i$; this is then assigned as the p^{th} sample $X[p]$. These P samples then provide a Monte Carlo approximation to the smoothing distribution of the state-space model.

Remembering that the aim is to recover the distribution over the unknown internal states and inputs to the nonlinear system, given the augmentation of the states with the state-space representation of the GP, this smoothing distribution is the object of interest. Therefore, employing this inference on the state-space model described in equations (13) and (14), will allow the input-state estimation task to be carried out.

3 INPUT ESTIMATION OF A BOUC-WEN SYSTEM

In this work, the Bouc-Wen hysteretic system [14] is considered as a typical and interesting benchmark. The model is formed in the same manner as in [15], with the equation of motion given as,

$$m\ddot{y} + c\dot{y} + ky + z(y, \dot{y}) = u(t) \tag{15a}$$

$$\dot{z}(y, \dot{y}) = \alpha\dot{y} - \beta(\gamma|\dot{y}||z|^{\nu-1}z + \delta\dot{y}|z|^\nu) \tag{15b}$$

The parameters of the system are chosen here to be $m = 2$, $c = 100$, $k = 5 \times 10^4$, $\alpha = 5 \times 10^4$, $\beta = 1 \times 10^3$, $\gamma = 0.8$, $\delta = -1.1$, $\nu = 1$. The response of this system is simulated subject to two different loading scenarios. First, the response of the system is simulated when the load is a random sample drawn from a Gaussian process in time, the exact form of the proposed model. Secondly, the response of the system to a sine wave excitation at 30 Hz, an input which is not explicitly drawn from the GP prior over the forcing, but for which the GP may be a suitable prior. It will now be shown that in both of these cases, very good estimates of the internal states of the Bouc-Wen system and estimates of the unmeasured forcing signals can be recovered. In both cases, the measured quantity is the acceleration of the oscillator, which is corrupted by an additive white Gaussian noise with a variance of approximately 2% of the variance of the measured acceleration. For both experiments a Matérn 5/2 kernel is used to model the unknown latent force which adds three additional states to the nonlinear SSM being identified.

3.1 Loaded with GP

Initially, loading the oscillator with a sample of a Gaussian process with a Matérn 5/2 kernel, the measured acceleration response to this load is simulated and shown in Figure 1.

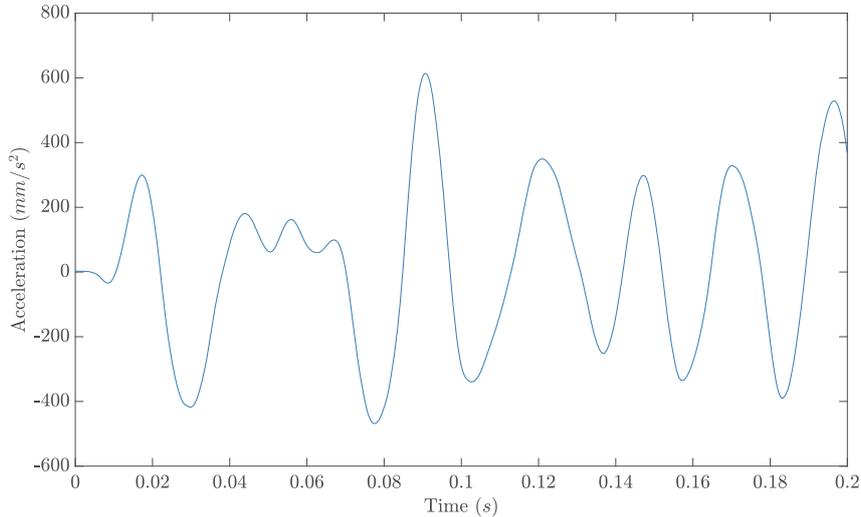


Figure 1: Measured acceleration from the Bouc-Wen oscillator when forced by a random sample from a Gaussian process with a Matérn 5/2 kernel.

This measured acceleration is used as the observed quantity in the model, as described in the previous section. The Bouc-Wen equations are used to describe a nonlinear state-space model of the dynamic system, which is coupled with the state-space representation of a Gaussian process in time with a zero mean and a Matérn 5/2 kernel (the same as used to generate the data). Given this known model form, it is expected that the model should perform well, provided the inference scheme used (PGAS) converges appropriately.

In Figure 2, the estimated states from the proposed inference scheme are shown. On the left hand side of the figure, the MCMC sampled paths are shown; the Markov chain was run for 5000 iterations with the first 1000 discarded as burn-in and the chain thinned by a factor of two. The particle filter was run as a bootstrap filter with 15 particles included; notably, even with this low number of particles the smoothing distribution can be seen to be well approximated. If the samples are used to construct a Gaussian approximation of the smoothing distribution, as shown in the left hand column of the figure, then it can be observed that all of the ground truth

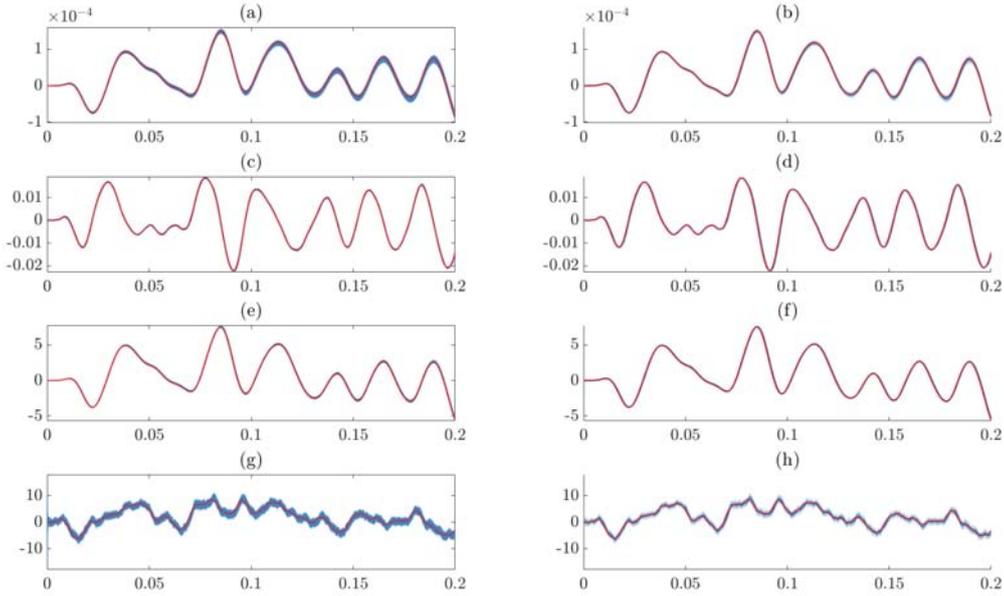


Figure 2: Estimated states inferred via the PGAS sampling of the smoothing distributions. In the left column, the samples obtained from the MCMC scheme and on the right Gaussian approximations to the distributions with one, two and three sigma intervals shaded. The four rows correspond to the first four states of the model: displacement, velocity, $z(y, \dot{y})$ and the latent force. In all plots the ground truth is indicated in red.

states lie within a three-sigma confidence interval.

To consider the quality of the fit in a purely deterministic sense, a normalised mean-squared error metric is used such that,

$$NMSE = \frac{100}{N\sigma^2} \sum_{i=1}^N \{(y_i - \hat{y}_i)\}$$

if y is the measured “true” signal of length N with variance σ^2 and the point estimates are given by \hat{y} . Intuitively, this leads to values bounded at the lower end by zero which corresponds to an exact fit and, as the quality of the fit degrades, the value increases. For some sense of this quality, a prediction where every point was equal to the sample mean of $\{y\}_{i=1}^N$ will give an NMSE of 100. Anecdotally, one can consider a value less than 5 to be a ‘very good’ fit and below 1 an ‘excellent’ fit.

For the estimations shown in Figure 2, if the mean of the samples is used as the expectation over the states to provide a point estimate, the NMSE values are calculated. For the displacement (frames (a) and (b)) the NMSE is 0.38; for the velocity (frames (c) and (d)), 0.33; for $z(y, \dot{y})$ (frames (e) and (f)) 0.25; and for the force estimate (frames (g) and (h)) 2.18. These metrics indicate that the method is performing well and confirms what is presented visually in Figure 2. Excellent recovery of the dynamic states of the oscillator is observed, including the additional hidden state related to the hysteresis in the system, and the estimate of the forcing is also very good. Some of the increase in the NMSE when considering the forcing can be attributed to the high-frequency components of this force which are smoothed out when the mean over the samples is used as the point estimate. Since the uncertainty in the forcing is also

quantified and shown in the figure, it can be seen that the behaviour of the forcing has been captured remarkably well in this complex system.

3.2 Loaded with Sine Wave

In the previous experiment, it was known that the form of the latent force model exactly matched the true forcing signal being applied to the system, i.e. the system was loaded with a sample of a Gaussian process. A more challenging and realistic test case is one where the loading signal is not a draw from a Gaussian process, but instead some known deterministic function. A sine wave is chosen to be such a function with the forcing defined as $u(t) = 120 \sin(30 \cdot 2\pi \cdot t)$.

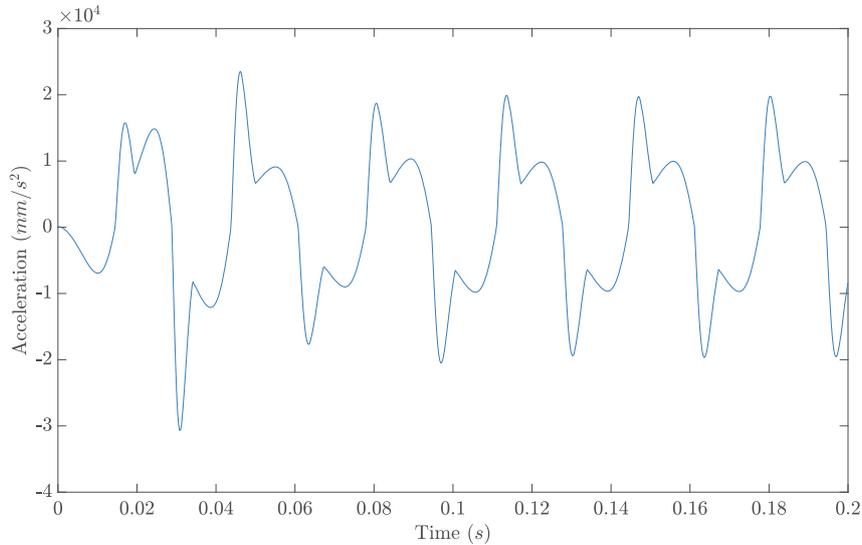


Figure 3: Measured acceleration from the Bouc-Wen oscillator when forced by a 30 Hz sine wave.

The response of the system to this sinusoidal forcing is shown in Figure 3. It is worth noting at this point that, if it were known that a sine wave had been used to load the system, then it is likely far less involved methods would work well to recover that signal. However, the power of the proposed approach is that it uses the GP as a functional prior over the loading function, which removes the need for prior knowledge with respect to the functional form of the forcing which has been applied. This flexibility is particularly important where the system may be subject to complicated and varying loads which are not easily expressed as a known function, e.g. wind or wave loading on offshore structures.

As previously, the nonlinear GP latent force model is used to draw samples from the smoothing distribution of a nonlinear SSM which is the Bouc-Wen model augmented by a GP with a Matérn 5/2 covariance function. 5000 samples of the states are sampled using PGAS, of which 1000 are discarded and 15 particles are used in the bootstrap particle filter. Doing so allows the results in Figure 4 to be constructed in the same way as Figure 2. In the left hand column, the samples obtained from MCMC are shown in blue, with the ground truth shown in red and on the right Gaussian assumed densities created by taking the expectation and variance of the samples at every point in time. The rows again correspond to the states: displacement, velocity, $z(y, \dot{y})$ and force. Reassuringly, the methodology continues to perform well, visually, even when the loading signal does not match the prior functional form of the GP very closely.

As with the previous case it is possible to consider the NMSE between the ground truth and

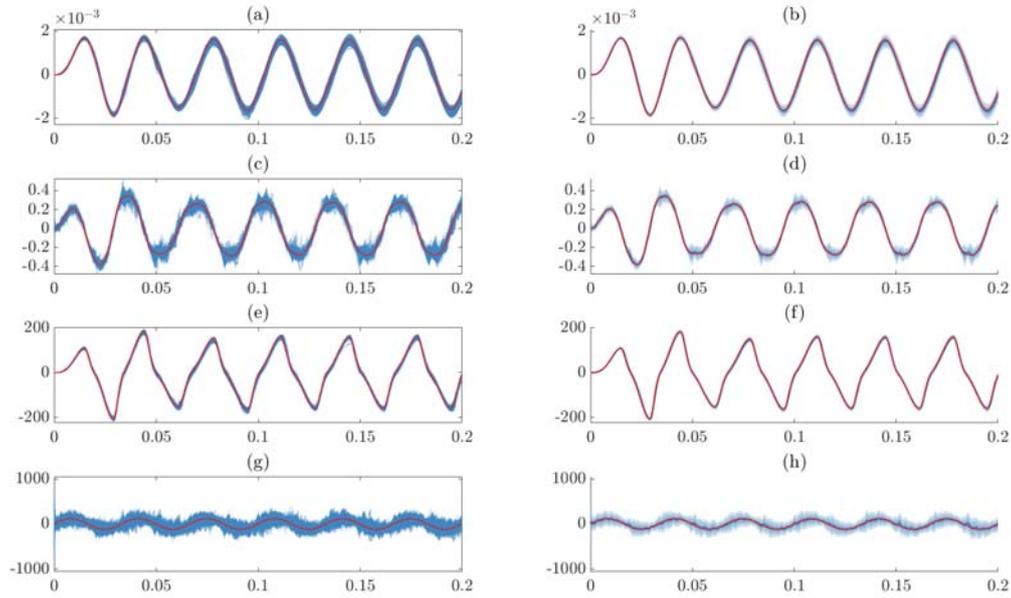


Figure 4: Estimated states inferred via the PGAS sampling of the smoothing distributions. In the left column, the samples obtained from the MCMC scheme and on the right Gaussian approximations to the distributions with one, two and three sigma intervals shaded. The four rows correspond to the first four states of the model: displacement, velocity, $z(y, \dot{y})$ and the latent force. In all plots the ground truth is indicated in red.

the expected values of the smoothing distributions. Considering the mean estimates gives values of 0.48, 0.36, 0.31 and 5.22 for the displacement, velocity, $z(y, \dot{y})$ and force states respectively. These values are larger than those observed in the previous experiment but only marginally so, which reinforces the visual quality of the fit seen in Figure 4. This slight increase in the error of the pointwise predictions is accompanied by an increase in the uncertainties estimated for all states. This uncertainty manifests as increased variance in the samples which have been generated from the Markov chain. These increases are most prominent in the forcing state, which sees a far larger degree of uncertainty, and in the velocity state where significant increases in uncertainty are seen close to the peaks of the response. It may be the case that the strong dependence of $z(y, \dot{y})$ on \dot{y} causes uncertainty to be pushed onto the velocity state when the effect of the nonlinearity on the observed acceleration is greatest.

Considering more closely the estimate of the forcing, the estimates in Figure 4 are also shown in Figure 5 for the forcing state only. It can be seen that relative to the case study in Section 3.1 the uncertainty in the forcing estimate is far higher. However, clearly the main trend of the sine wave has been recovered, and the ground-truth signal lies close to the expected values of the smoothing distribution. In the lower frame of Figure 5, the Gaussian approximation to the smoothing distribution more clearly shows how well the method is performing, despite the slight increase in NMSE for this case. Remembering that this distribution has been approximated by a set of Monte Carlo samples, it can be seen that the main cause of error is high-frequency content in the mean estimate. These errors may well be a consequence of the limited number of samples used to form the estimate, which will be discussed with reference to the computational load of the proposed method. It is the opinion of the authors that the results shown here represent a good recovery of the unknown load on the system.

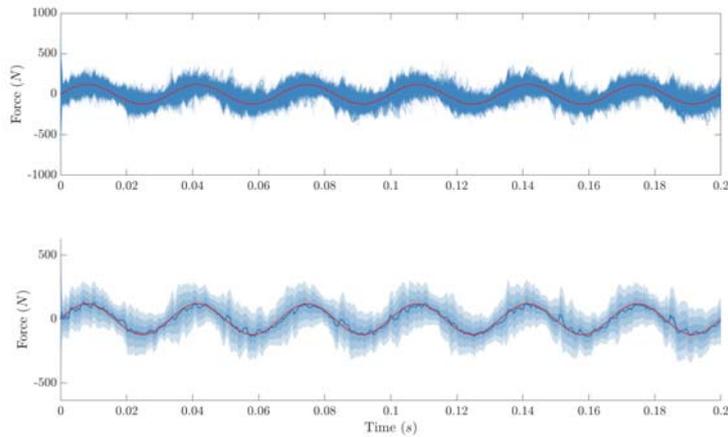


Figure 5: Estimated sine wave loading on the Bouc-Wen oscillator from the nonlinear latent force model. Ground truth shown in red. Above, samples of the possible loading signals. Below, mean estimate in blue and three sigma intervals shaded, of the approximated Gaussian distribution.

The results shown from these two case studies with the Bouc-Wen system provide reassuring evidence that the approach proposed for joint input-state estimation in nonlinear systems has the potential to work equally well when there are hysteretic nonlinearities present in the system. It would appear that the additional states introduced in the model do not cause significant observability issues, in that the estimates of the states and the unknown loads remain very good. However, it should be noted that there is still room for improvement. One shortcoming of the methodology presented in this work is the significant computational burden. This stems from the requirement to run multiple particle filters, which significantly increases the computation time relative to the linear case [8]. To overcome this practical limitation, it may be necessary to consider more efficient inference schemes in the future, or to resort to a different approximation of the nonlinear system, for example, a Gaussian filter which approximates the nonlinearity. This is expected to be of most benefit when the nonlinearity is weak in the system.

4 CONCLUSION

This paper has sought to understand the performance of a nonlinear input-state estimation methodology proposed in [1] when the nonlinear system contains a hysteretic nonlinearity. A Bouc-Wen system was chosen as a typical example of such a nonlinear dynamic model. It was shown that, adopting the nonlinear Gaussian process latent restoring force approach based on Particle Gibbs with Ancestor Sampling, it was possible to perform highly-accurate nonlinear input-state estimation on this challenging dynamical system. Errors in state estimates were seen to be consistently below 1%, including the additional internal state of the model, $z(y, \dot{y})$. The estimates of the forcing were also seen to result in low pointwise error: 2.18% in the case where a GP was used as the loading signal and 5.22% when a sine wave forcing was used. In conjunction with these low error values, reasonable estimation of the uncertainty in the states was also recovered which may be of more value when this type of identification is coupled with further analyses. Finally, it was discussed how the major drawback of the proposed method, i.e. its computational burden, is a promising area for further investigation.

5 ACKNOWLEDGEMENTS

The authors gratefully acknowledge funding from the Engineering and Physical Sciences Research Council (EPSRC), UK, under grants: EP/S001565/1 and EP/R003645/1. Thanks is also extended to Ramboll Energy for their support of J. D. Longbottom.

References

- [1] Timothy J. Rogers, Keith Worden, and Elizabeth J. Cross. Bayesian joint input-state estimation for nonlinear systems. *Vibration*, 3(3):281–303, 2020.
- [2] Simo Särkkä. *Bayesian Filtering and Smoothing*, volume 3. Cambridge University Press, 2013.
- [3] Anthony O’Hagan. Curve fitting and optimal design for prediction. *Journal of the Royal Statistical Society: Series B (Methodological)*, 40(1):1–24, 1978.
- [4] Christopher K.I. Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006.
- [5] Mauricio Alvarez, David Luengo, and Neil D. Lawrence. Latent force models. In *Artificial Intelligence and Statistics*, pages 9–16, 2009.
- [6] Jouni Hartikainen and Simo Särkkä. Kalman filtering and smoothing solutions to temporal gaussian process regression models. In *2010 IEEE International Workshop on Machine Learning for Signal Processing*, pages 379–384. IEEE, 2010.
- [7] Jouni Hartikainen and Simo Sarkka. Sequential inference for latent force models. *arXiv preprint arXiv:1202.3730*, 2012.
- [8] Timothy J. Rogers, Keith Worden, and Elizabeth J. Cross. On the application of Gaussian process latent force models for joint input-state-parameter estimation: With a view to Bayesian operational identification. *Mechanical Systems and Signal Processing*, 140: 106580, 2020.
- [9] Arnaud Doucet and Adam M. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering*, 12(656-704):3, 2009.
- [10] Christian Andersson Naesseth, Fredrik Lindsten, and Thomas B. Schön. Sequential Monte Carlo for graphical models. *Advances in Neural Information Processing Systems*, 27: 1862–1870, 2014.
- [11] Christophe Andrieu, Arnaud Doucet, and Roman Holenstein. Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(3):269–342, 2010.
- [12] Andreas Svensson, Thomas B. Schön, and Manon Kok. Nonlinear state space smoothing using the conditional particle filter. *IFAC-PapersOnLine*, 48(28):975–980, 2015.
- [13] Fredrik Lindsten, Michael I. Jordan, and Thomas B. Schön. Particle Gibbs with ancestor sampling. *The Journal of Machine Learning Research*, 15(1):2145–2184, 2014.

- [14] Mohammed Ismail, Fayçal Ikhouane, and José Rodellar. The hysteresis Bouc-Wen model, a survey. *Archives of Computational Methods in Engineering*, 16(2):161–188, 2009.
- [15] Maarten Schoukens and Jean Philippe Noël. Three benchmarks addressing open challenges in nonlinear system identification. *IFAC-PapersOnLine*, 50(1):446–451, 2017.

INFLUENCE OF DIFFERENT FULLY NON-STATIONARY ARTIFICIAL TIME HISTORIES GENERATION METHODS ON THE SEISMIC RESPONSE OF FREQUENCY-DEPENDENT STRUCTURES

Federica Genovese^{1,2*}, Giuseppe Muscolino² and Alessandro Palmeri¹

¹ School of Architecture, Building and Civil Engineering, Loughborough University,
Sir Frank Gibb, Loughborough LE11 3TU, England, United Kingdom
F.Genovese@Lboro.ac.uk, A.Palmeri@Lboro.ac.uk

² Department of Engineering, University of Messina,
C.da di Dio, Villaggio S. Agata, 98166, Messina, Italy
fedgenovese@unime.it, gmuscolino@unime.it

Abstract

Advances in computational power and numerical optimization routines enable the application of rigorous simulation and optimization techniques for the design and assessment of structural and geotechnical (S&G) systems. Nowadays, the state of practice in the seismic design of high-importance structures has progressively moved toward the use of time-history dynamic analysis that are highly sensitive to the way of modelling the dynamic action. To be reliable, this type of analysis requires efficient methodologies for the selection of the seismic inputs used as ground excitations.

In this paper, a new stochastic approach, based on the use of the wavelet transform is proposed to generate an arbitrary number of seismic records having the same characteristics of a target accelerogram. The choice of the amplitude and number of bands in which to partition the frequency domain is a key source of variability to consider for the generation of samples with the desired time-varying amplitude and frequency content.

To evaluate the influence of the use of an alternative fully non-stationary artificial accelerograms generation methods, a comparison between the proposed method and the one recently proposed by the first two authors is also presented in this paper. Specifically, in order to quantify the influence of accelerograms models on the seismic response, a structural system with viscoelastic damping is analysed, representative of a broad range of frequency-dependent S&G assets.

Keywords: Fully non-stationary models, Harmonic Wavelet Transforms, Artificial accelerograms, Time-history dynamic analyses, Maxwell Model.

1 INTRODUCTION

The pseudo-acceleration response spectrum, typically used by international seismic codes for modelling the earthquake-induced ground shaking, significantly facilitates the design of regular structure through proper modal combination rules. However, there are many cases (e.g. seismic site response analysis and seismic design of structures with energy dissipation devices) in which the use of the elastic design response spectrum is not considered appropriate and non-linear dynamic time-history analyses are preferred. In these situations, the use of a suite of real accelerograms is an attractive option for modelling the seismic excitation; thus, different Ground Motion Selection and Modification methods have been proposed in literature (see e.g. [1-4]).

As evidenced in [5-7], the result of the selection procedures is influenced by multiple sources of uncertainties related to the seismic hazard at the site of interest and to the mechanical characteristics of the soil layers. As a consequence, it could be very difficult, if not impossible, to select an adequate number of recorded accelerograms without applying large scale factors to each record of the set, which in turn would distort the original signals characteristics.

A suitable alternative for the definition of the input ground motion consists in the use of sets of artificial accelerograms, generated by procedures able to capture the large variability of the seismological parameters observed in recorded time-histories (see e.g [8,9]).

The wavelet transform represents a useful tool to perform a joint-time frequency representation of non-stationary signals; for this reason, different wavelet-based approaches have been used for simulating artificial non-stationary accelerograms (see e.g. [10,11]).

In this paper, a new stochastic artificial accelerograms generation method, based on the use of circular wavelets transform (*CWT*), is presented.

In comparison to other generation strategies available in the literature, the proposed procedure enables the generation of artificial time histories without the need of defying the evolutionary power spectral density (*EPSD*) function of the ground acceleration.

The correct choice of the amplitude and number of bands in which to partition the frequency domain is one of the main sources of variability to consider to achieve the generation of samples with the desired non-stationary characteristics.

In order to quantify the influence of different accelerograms models on the seismic response, a comparison between the proposed method and the piecewise *EPSD* function method, recently proposed in [12] is also reported in this paper.

The second stochastic generation method, for a given target accelerogram, requires the following steps: *i*) find a fully non-stationary model of earthquake ground motion such that the target accelerogram may be considered as one of its samples; *ii*) evaluate the mean elastic response spectrum of a set of generated fully non-stationary accelerogram samples; *iii*) satisfy the compatibility with the elastic target response spectrum by means of an iterative procedure.

In order to highlights the performance of these two alternative probabilistic models on the seismic response, a structural system with frequency-dependent damping, based on the viscoelastic Maxwell model, is analysed in this paper.

2 WAVELET-BASED FULLY NON-STATIONARY GENERATION METHOD

The wavelet analysis consists in the expansion of a given signal in terms of “wavelets”, which are generated by scaling and shifting a chosen function called “mother wavelet”. Among all different types of wavelets, the “harmonic” and “musical” ones proposed in [13] are particularly useful for dynamic analysis. These families of wavelets are complex-valued functions in the time domain, with a rectangular box-shaped Fourier transform in the frequency domain.

Another approach to decompose a real-valued signal into the superposition of complex-valued wavelets $\psi_{\{m,n\},k}(t)$ having complex-valued combination coefficients $a_{\{m,n\},k}$ consists in the use of the circular wavelets. The whole set of band circular wavelets is generated by:

$$\psi_{\{m,n\},k}(t) = \frac{1}{n-m} \sum_{j=m}^{n-1} \exp\left[i 2 \pi j \left(t - \tau_{\{m,n\},k} \right) \right] \quad (1)$$

where the notation $\{m,n\}$ is used to denote a wavelet occupying the band of circular frequencies from $2\pi m$ to $2\pi n$, with $n > m$; $\tau_{\{m,n\},k} = k / (n-m)$ is a deterministic time shift of the wavelets belonging to the $\{m,n\}$ -indexed frequency band; $k = 0, \dots, (n-m-1)$ is an integer number; and $i = \sqrt{-1}$ is the imaginary unit.

In the discrete wavelets transform, the complex-valued combination coefficients $a_{\{m,n\},k}$ of a target signal in the time domain are calculated by a discrete convolution of the signal, say $\ddot{U}_g(t)$, with the band wavelets $\psi_{\{m,n\},k}(t)$ [3]:

$$a_{\{m,n\},k} = (n-m) \sum_{\ell=0}^N \ddot{U}_g(t_\ell) \cdot \frac{\Delta t}{t_f} \cdot \overline{\psi_{\{m,n\},k}\left(\frac{t_\ell}{t_f}\right)} \quad (2)$$

where the over-bar denotes the complex conjugate, $t_\ell = \ell \cdot \Delta t$ is the ℓ th of the $N = t_f / \Delta t$ discrete time instants at which the signal is discretized, being Δt and t_f the sampling interval and the time duration of the selected signal, respectively.

In this paper, the circular wavelets have been used for the randomisation of the target signal $\ddot{U}_g(t)$. Specifically, the generation of the r th fully non-stationary artificial sample of the random process can be evaluated by the following formula:

$$f^{(r)}(t) = 2 \sum_{\{m,n\}} \sum_{k=0}^{n-m-1} \sum_{j=m}^{n-1} \frac{|a_{\{m,n\},k}|}{n-m} \times \cos \left[i 2 \pi j \left(\frac{t}{t_f} - \tau_{\{m,n\},k} \right) + \theta_{\{m,n\},k} + \phi_{\{m,n\},k}^{(r)} \right] \quad (3)$$

where $\phi_{\{m,n\},k}^{(r)}$ is the r th realization of a random variable uniformly distributed over the interval $[0, 2\pi]$ while $\theta_{\{m,n\},k} = \arg\{a_{\{m,n\},k}\}$ is the corresponding deterministic phase of the complex-valued coefficient of the target signal.

3 EVOLUTIONARY POWER SPECTRAL DENSITY FUNCTION METHOD FOR MODELLING SEISMIC ACTION

For the sake of completeness, the four-step method recently proposed in [12] for generating random samples of a fully non-stationary zero-mean Gaussian process consistent with a target accelerogram is summarized in this section.

First, the time axis is divided in n contiguous time intervals, in which a uniformly modulated process is introduced as the product of a deterministic modulating function, $a(t)$, times a stationary zero-mean Gaussian sub-process $X_k(t)$, whose power spectral density (PSD) function $G_{X_k}(\omega)$ is filtered by two Butterworth filters:

$$G_{X_k}(\omega) = \beta_k \left(\frac{\omega^2}{\omega^2 + \omega_{H,k}^2} \right) \left(\frac{\omega_{L,k}^4}{\omega^4 + \omega_{L,k}^4} \right) \rho_k \left(\frac{1}{\rho_k^2 + (\omega + \Omega_k)^2} + \frac{1}{\rho_k^2 + (\omega - \Omega_k)^2} \right); \quad k = 1, \dots, n \quad (4)$$

where β_k is evaluated in such a way that the sub-process $X_k(t)$ possesses unit variance. The predominant circular frequency Ω_k and the frequency bandwidth ρ_k in Eq. (4) depend on the occurrences of maxima P_k and of zero-level up-crossings $N_{0,k}^+$ of the target accelerogram, in the various k intervals:

$$\Omega_k \cong \frac{2\pi N_{0,k}^+}{\Delta T_k}; \quad \rho_k \cong \frac{\pi N_{0,k}^+}{2\Delta T_k} \left[\pi - 2 \frac{N_{0,k}^+}{P_k} \right] \quad (5)$$

Second, the modulating function $a(t)$ is evaluated by least-square fitting the *cumulative expected energy function* of the stochastic process to the cumulative energy function $E_{\ddot{U}_g}(t)$ of the target accelerogram subdivided in three-time intervals:

$$a(t) = \sum_{j=1}^2 \bar{a}_j(t) \mathbb{W}(t_{j-1}, t_j) + a(t_2) \exp \left[\frac{t-t_2}{t_f-t_2} \ln \left(\frac{|\ddot{U}_g(t_f)|}{a(t_2)} \right) \right] \mathbb{W}(t_2, t_3). \quad (6)$$

in which $\mathbb{W}(t_{j-1}, t_j) = \mathcal{U}(t-t_j) - \mathcal{U}(t-t_{j-1})$ is the window function and $\mathcal{U}(t)$ the Heaviside unit step function.

Third, the i th sample of the random process is generated via the superposition of harmonic functions with random phases:

$$F_0^{(i)}(t) = a(t) \sqrt{2\Delta\omega} \left[\sum_{k=1}^n \sum_{r=1}^{m_N} \mathbb{W}(t_{k-1}, t_k) \sin(r \Delta\omega t + \theta_r^{(i)}) \sqrt{G_{X_k}(r \Delta\omega)} \right] \quad (7)$$

being $\theta_r^{(i)}$ the random phase angles, uniformly distributed over the interval $[0, 2\pi]$ and m_N the number of parts in which the k th *PSD* function $G_{X_k}(\omega)$ is discretized with a $\Delta\omega$ frequency sampling interval.

Finally, the spectrum-compatibility is obtained by reducing the gap between the mean spectrum of the generated samples $\bar{S}^{(j-1)}(\omega, \zeta_0)$ and the target one $S^{(T)}(\omega, \zeta_0)$, through the introduction of a corrective iterative *PSD* function $\bar{G}_{X_k}^{(j)}(\omega)$:

$$\bar{G}_{X_k}^{(j)}(\omega) = \bar{G}_{X_k}^{(j-1)}(\omega) \frac{S^{(T)}(\omega, \zeta_0)^2}{\bar{S}^{(j-1)}(\omega, \zeta_0)^2} \quad (8)$$

being $\bar{G}_{X_k}^{(0)}(\omega) = 1$ [14] and ζ_0 the viscous damping.

According to the formulation described in [15], the generic spectrum-compatible sample can be generated as:

$$\bar{F}_0^{(i)}(t) = a(t) \sqrt{2\Delta\omega} \left[\sum_{k=1}^n \sum_{r=1}^{m_N} \mathbb{W}(t_{k-1}, t_k) \sin(r \Delta\omega t + \theta_r^{(i)}) \sqrt{\bar{G}_{X_k}^{(j)}(r \Delta\omega) G_{X_k}(r \Delta\omega)} \right]. \quad (9)$$

Different types of spectrum-compatibility can be achieved, modifying the corrective iterative *PSD* function term $\bar{G}_{X_k}^{(j)}(\omega)$ as shown in [16].

4 NUMERICAL APPLICATION

In order to quantify the influence of the way of modelling the expected seismic action, two set of one hundred *Fully Non-Stationary* samples consistent with a target seismic accelerogram have been generated using the proposed wavelet-based method (*CWT*) and the evolutionary piecewise power spectral density (*EPSD*) function procedure proposed in [12].

A comparison in terms of displacement response spectra, computed for SDoF oscillators with viscoelastic damping based on the Maxwell model is also presented in the paper.

4.1 Target Motion

The North-South component of the ground motion recorded at Vasquez Rocks Park during the 1994 Northridge earthquake has been used in the following as target accelerogram $\ddot{U}_g(t)$.

The selected ground motion, downloaded from the Peer database [17], having a moment magnitude $M_w = 6.7$ and a site-source distance $R_{JB} = 23.1$ km [18], has been recorded with a sampling time $\Delta t = 0.02$ s by a station having an average shear wave velocity in the upper 30 m equal to $V_{s,30} = 996$ m/s (EC8 [19], soil class “A”). The total Intensity of the target accelerogram, having an overall duration $t_f = 36.6$ s, is equal to $I_0 = 1.9$ m²/s³, while the total number of zero-level up-crossings and peaks are $N_0^+ = 196$ and $P_0 = 212$, respectively.

In Figure 1, the trend of the time-history of the analysed target accelerogram is reported.

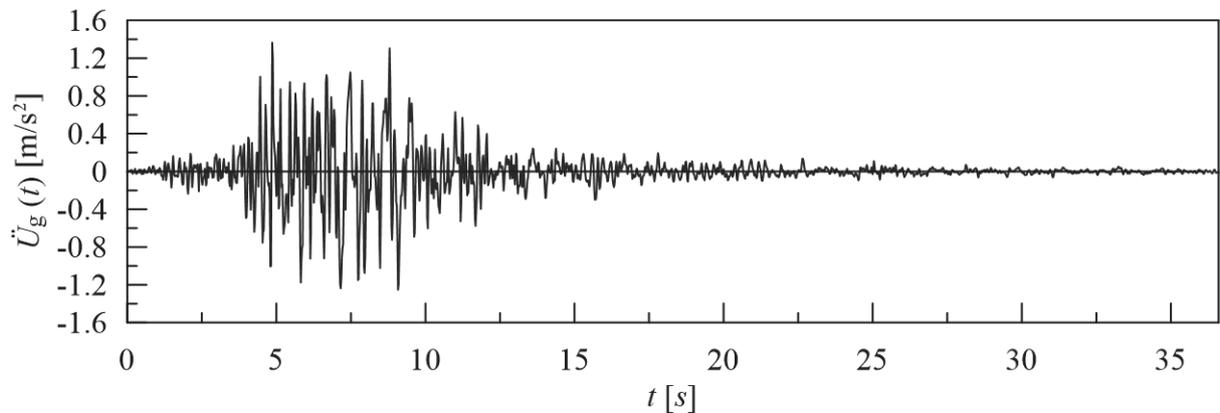


Figure 1: Time histories of the 1994 Northridge earthquake accelerogram.

4.2 Wavelet-based method

A fundamental step of the proposed Circular Wavelet Transform (*CWT*) method consists in the correct choice of the amplitudes of bands in which to divide the frequency domain of the selected signal $\ddot{U}_g(t)$.

In this paper, three different schemes have been investigated considering a different subdivision of the cumulative Fourier energy function evaluated as:

$$E(f) = \int_0^{fc} (|\mathcal{F}[\ddot{U}_g(t)]|)^2 df \quad (10)$$

fc being the cut-off frequency and $\mathcal{F}[\ddot{U}_g(t)]$ the Fourier transform of the target signal.

Accordingly, the frequency domain has been subdivided in frequency bands of:

- 1) equal-spaced bandwidths (*ESB*);
- 2) constant energy bandwidths (*CEB*);
- 3) non-uniform energy bandwidths (*NUEB*).

In the latter case, the choice of the bandwidths has been made according to the main changes in the slope of the $E(f)$ function. In all three investigated schemes a subdivision into 10 parts has been carried out.

The representation of the subdivision of the cumulative Fourier energy function for the three analysed schemes is reported in Figure 2.

In Figure 3, the mean values of the modules of the Fourier spectra, obtained for the three analysed configurations, are compared with the target one. It can be observed that the mean value of the module of the Fourier Spectrum of the generated samples is in a good agreement with the target one only in the case of subdivision of the frequency domain with energy criterion (schemes 2 and 3).

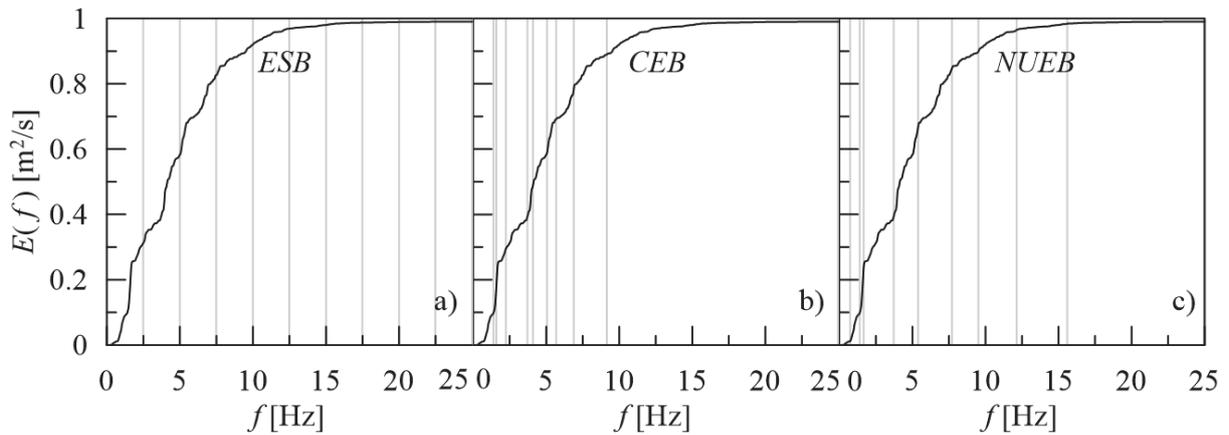


Figure 2: Representation of cumulative Fourier energy function (black line) together with the frequency bands subdivision (grey vertical lines) for the three investigated schemes: a) equal-spaced bandwidths (*ESB*); b) constant energy bandwidths (*CEB*); c) non-uniform energy bandwidths (*NUEB*).

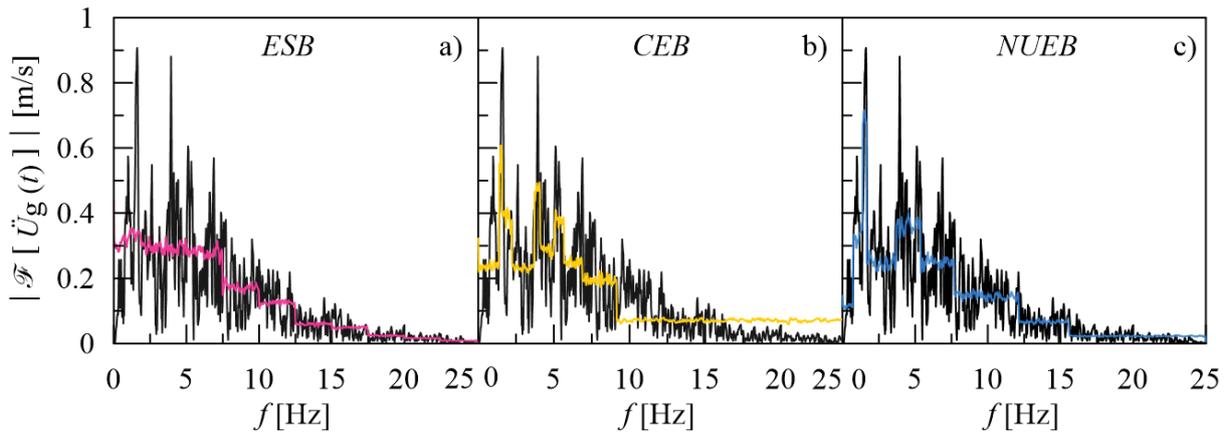


Figure 3: Comparison among mean Fourier spectrum module of the generated samples by the *CWT* method and the target one (black line) considering: a) equal-spaced bandwidths (*ESB*); b) constant energy bandwidths (*CEB*); c) non-uniform energy bandwidths (*NUEB*).

In Figure 4, three generic samples (coloured lines) are plotted against the target accelerogram (black line), showing that in all cases the variation in amplitude appears to be preserved in the time domain.

In Figure 5, the acceleration spectrum of the target accelerogram (red line) is compared to that obtained as the mean value of 100 samples (black solid line). The confidence interval evaluated as the mean value plus/minus the corresponding standard deviation (black dotted lines) and the envelope of the maximum and minimum values of all samples (shaded area) are also reported in Figure 5.

In Figures 6a and 6b, further comparisons are offered in terms of the cumulative energy functions $I_0(t)$ and the cumulative zero-level up crossing functions $N_0^+(t)$, respectively.

From the observation of the results obtained in the time and frequency domain, it emerges that a subdivision of the frequency domain in non-uniform energy bandwidths (third scheme) leads to outcomes statistically closer to those of the target event.

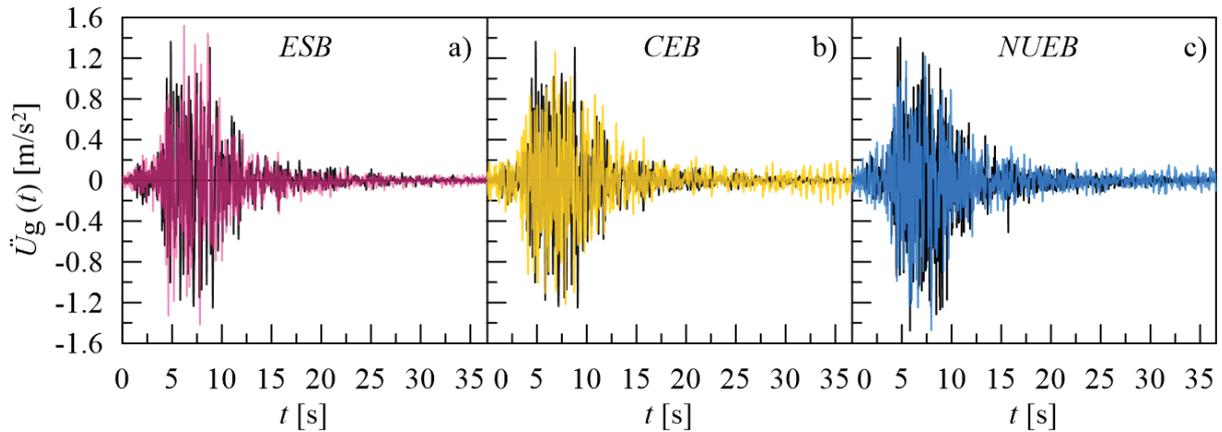


Figure 4: Comparison among the target accelerogram (black line) and the i th generated sample by the proposed CWT method, considering a subdivision of the frequency domain in: a) equal-spaced bandwidths (magenta line); b) constant energy bandwidths (yellow line); c) non-uniform energy bandwidths (blue line).

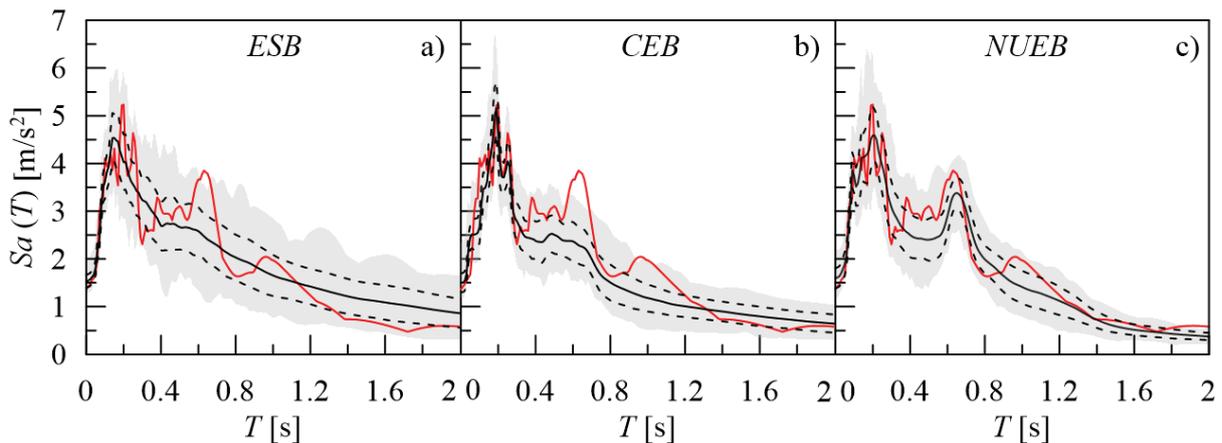


Figure 5: Comparison among the acceleration response spectrum of the target accelerogram (red solid line) with statistics of the artificial ones: mean value function (black line); mean value plus/minus standard deviation functions (black dashed lines); envelope of the maximum and minimum values of all samples (shaded area); considering a subdivision of the frequency domain in: a) equal-spaced bandwidths (*ESB*); b) constant energy bandwidths (*CEB*); c) non-uniform energy bandwidths (*NUEB*).

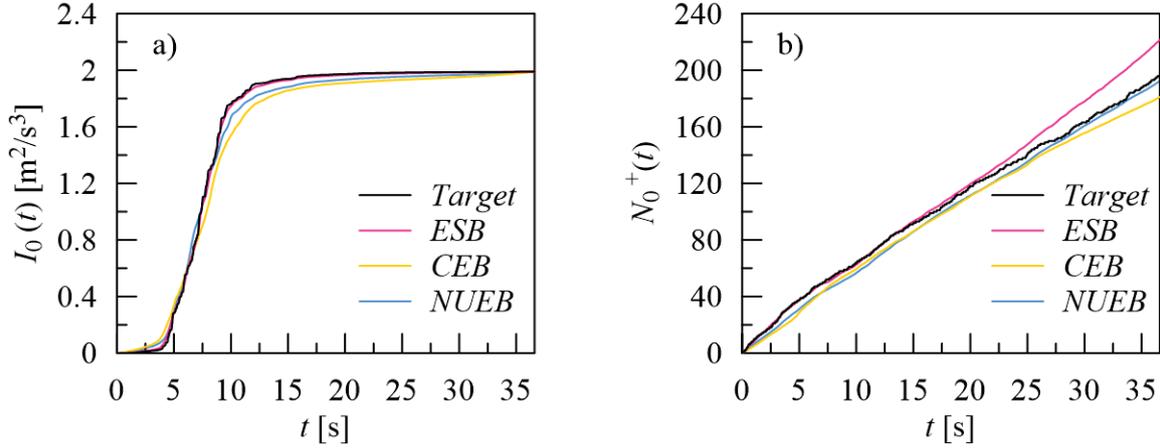


Figure 6: Comparison between the target (black line) and the averages a) cumulative energy functions, b) zero level up crossing functions, considering: equal-spaced bandwidths (ESB); constant energy bandwidths (CEB); c) non-uniform energy bandwidths (NUEB).

4.3 Evolutionary Power spectral density function (EPSD) method

Using the iterative procedure described in Section 3, through 4 iterations, a set of 100 artificial accelerograms has been generated using appropriate modulating and *PSD* functions which allowed to preserve the amplitude and the frequency content of the target ground motion. Further details about the parameters that characterize these functions can be found in [15].

4.4 Dynamic analyses of linear visco-elastic systems

In this section, displacement response spectra have been computed to illustrate the structural response of SDoF oscillators provided with linear viscoelastic damping ruled by:

$$m\ddot{u}(t) + r(t) = \ddot{U}_g(t) \quad (11)$$

where m is the mass of the system, $\ddot{u}(t)$ is the second-order derivative of the displacement relative to the ground $u(t)$ and $r(t)$ is the reaction force that in the Maxwell Model is given by:

$$r(t) = k_0 u(t) + k_M \lambda(t) \quad (12)$$

being k_0 the equilibrium modulus and $\lambda(t)$ the additional internal variable taken as the deformation of the Maxwell element, ruled by the state equation:

$$\dot{\lambda}(t) = \dot{u}(t) - \frac{\lambda}{\tau} \quad (13)$$

where:

$$\tau = \frac{c_M}{k_M} \quad (14)$$

k_M and c_M being the elastic stiffness and the viscous coefficient of the Maxwell element, respectively, while $\dot{u}(t)$ is the first-order derivative of the displacement $u(t)$.

In this paper the values of $k_M = 400$ N/m and $c_M = 40$ Ns/m have been assumed, according to case examined in [20].

In Figure 7, the displacement spectra of the target accelerogram (red lines) are compared to that obtained as the mean value of 100 samples (black continues lines), using: a) the Circular Wavelet transform method, with a subdivision of the frequency domains in non-uniform energy bandwidths (case 3); b) the Evolutionary Power spectral density function method.

The confidence intervals are evaluated as the mean values plus/minus the corresponding standard deviations (black dotted lines); the envelope of the maximum and minimum values of all samples (shaded area) are also reported in Figure 7.

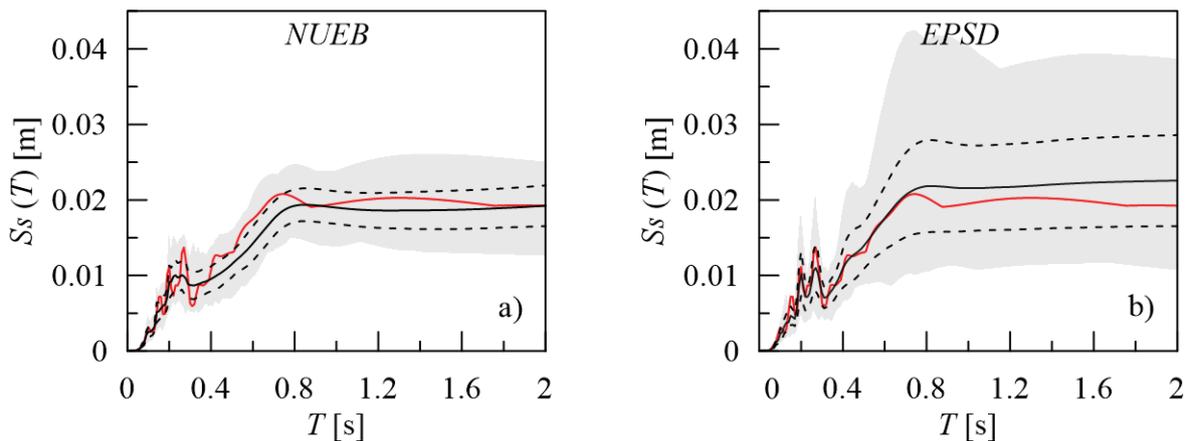


Figure 7: Comparison between target displacement spectrum (red line) of a system with linear viscoelastic damping based on the Maxwell model together with statistics of the artificial ones: mean value function (black solid line); mean value plus/minus standard deviation functions (black dashed lines); envelope of the maximum and minimum values of all samples (shaded area): a) *CWT* method with non-uniform energy bandwidths (*NUEB*), b) *EPSD* function method.

5 CONCLUSIONS

Seismic response of structural and geotechnical systems is often highly sensitive to the way of modelling the dynamic action. In order to verify the influence of different fully non-stationary artificial accelerograms generation procedures, two stochastic methods, have been compared.

The *CWT*-method, proposed in this paper, consists in a phase angle rotation of the circular wavelets in the complex-valued space and allows the generation of the required number of fully non-stationary samples without the need of defying the evolutionary power spectral density function of the ground acceleration.

The choice of the number and the amplitude of bands in which to divide the frequency domain is an important step to generate samples with the desired time-variation of amplitude and frequency content.

The numerical results show that the generated samples obtained by the *CWT* method are close to the target one in the case of the subdivision of the frequency domain in non-uniform energy bandwidths; furthermore, the average displacement spectrum of the artificial samples generated by the *CWT* method and computed for SDoF oscillators with viscoelastic damping has a trend close to the target one.

The application of iterative corrections in the *EPSD* method allows the target displacement spectrum to completely fall into the confidence interval evaluated as the mean value plus/minus standard deviation of the displacement spectrum of the generated samples. It appears that the *EPSD* function method allows to obtain samples having a displacement response spectrum closer to the target one; however, it tends to be more complex and requires several iterative steps.

REFERENCES

- [1] F. Genovese, D. Aliberti, G. Biondi, E. Cascone, A procedure for the selection of input ground motion for 1D seismic response analysis. *Earthquake Geotechnical Engineering for Protection and Development of Environment and construction*. F. Silvestri and N. Moraci eds. *7th International Conference on Earthquake Geotechnical Engineering (ICEGE 2019)*, Rome, Italy, June 17-20, 2019, pp. 2591-2598, 2019.
- [2] J.J. Bommer, B. Acevedo, The use of real earthquake accelerograms as input to dynamic analysis. *Journal of Earthquake Engineering*, **8**, Special issue, 1, pp. 43-91, 2004.
- [3] D. Cecini, A. Palmeri, Spectrum-compatible accelerograms with harmonic wavelets, *Computers and Structures*, **147**, pp.26-35, 2015.
- [4] M. Rota, E. Zuccolo, L. Taverna, M. Corigliano, C. G. Lai, A. Penna, Mesozonation of the Italian territory for the definition of real spectrum-compatible accelerograms, *Bulletin of Earthquake Engineering*, **10**, pp. 1357–1375, 2012.
- [5] F. Genovese, D. Aliberti, G. Biondi, E. Cascone, Influence of Soil Heterogeneity on the Selection of Input Motion for 1D Seismic Response Analysis. F. Calvetti, F. Cotecchia, A. Galli, C. Jommi eds, *Geotechnical Research for Land Protection and Development. CNRIG 2019, Lecture Notes in Civil Engineering, Springer, Cham*, **40**, pp. 694–704, 2020.
- [6] F. Genovese, D. Aliberti, G. Biondi, E. Cascone. Geotechnical aspects affecting the selection of input motion for seismic site response analysis. M. Papadarakakis, M. Fragiadakis eds, *7th Computational Methods in Structural Dynamics and Earthquake Engineering (COMPDYN 2019), 24-26 June 2019, Crete, Greece*, **1**, pp. 151-161, 2019.
- [7] F. Genovese, Influence of Soil Non-linear Behaviour on the Selection of Input Motion for Dynamic Geotechnical Analysis. M. Barla, A. Di Donna, D. Sterpi D. eds, *Challenges and Innovations in Geomechanics, 16th International Conference of the International Association for Computer Methods and Advances in Geomechanics (IACMAG 2021)*, Lecture Notes in Civil Engineering, Springer, Cham, **126**, pp. 588-596, 2021.
- [8] G. Stefanou, S. Tsiliopoulos, Estimation of evolutionary power spectra of seismic accelerograms. M. Papadarakakis, M. Fragiadakis (Eds), *7th International Conference on Computational Methods in Structural Dynamics and Earthquake Engineering*, **3**, pp. 5880–5888, 2019.
- [9] P. Cacciola, A stochastic approach for generating spectrum-compatible fully non-stationary earthquakes, *Computers and Structures*, **88**, pp. 889-901, 2010.
- [10] J. Iyama, H. Kuwamura, Application of wavelets to analysis and simulation of earthquake motions, *Earthquake Engineering and Structural Dynamics*, **28**, no. 3, pp. 255–272, 1999.
- [11] A. Giaralis and P. D. Spanos, Wavelet-based response spectrum compatible synthesis of accelerograms - Eurocode application (EC8), *Soil Dynamics and Earthquake Engineering*, **29**, no. 1, pp. 219–235, 2009.
- [12] G. Muscolino, F. Genovese, G. Biondi, E. Cascone, Generation of Fully Non-Stationary Random Processes Consistent with Target Seismic Accelerograms, *Soil Dynamics and Earthquake Engineering*, **141**(106467), pp. 1-14, 2021.

- [13] D. Newland, Harmonic and musical wavelet. Proceedings, *Royal Society A*, **44**, pp.605-620, 1994.
- [14] E.H. Vanmarcke, D.A. Gasparini, Simulated earthquake ground motions. *K - Seismic Response Analysis of Nuclear Power Plant Systems K1 - Ground Motion and Design Criteria SMiRT 4*, San Francisco, USA, 1977.
- [15] F. Genovese, G. Muscolino, G. Biondi and E. Cascone. Generation of artificial accelerograms consistent with earthquake-induced ground motions. M. Papadrakakis, M. Fragiadakis, C. Papadimitriou eds, *11th International Conference on Structural Dynamic, (EURODYN 2020)*, Virtual, Athens, Greece, November 23-26 2020; Code 165382, **2**, pp. 3027-3042, 2020.
- [16] F. Genovese, G. Muscolino, G. Biondi, E. Cascone. A novel method for the generation of fully non-stationary spectrum compatible artificial accelerograms. M. Papadrakakis, M. Fragiadakis eds, *8th Computational Methods in Structural Dynamics and Earthquake Engineering, (COMPDYN 2021)*, Virtual, Athens, Greece, June 27–30 2021.
- [17] T.D. Ancheta, R.B. Darragh, J.P. Stewart, E. Seyhan, W.J. Silva, B.S.J. Chiou, K.E. Wooddell, R.W. Graves, A.R. Kottke, D.M. Boore, T. Kishida, and J.L. Donahue, PEER NGA-West2 database. Pacific Earthquake Engineering Research Center/03; California; 2013.
- [18] W.B. Joyner, D.M. Boore, Peak horizontal acceleration and velocity from strong motion records including records from the 1979 Imperial Valley, California, earthquake. *Bulletin of the Seismological Society of America*; **71**, pp. 2011–2038, 1981.
- [19] EC8 (European Committee for Standardization, Eurocode 8). Design of structures for earthquakes resistance-Part 1: General rules, seismic actions and rules for buildings, (EN 1998-1).
- [20] G. Muscolino, A. Palmeri, Dynamic analysis of viscoelastically damped structures, B.H.V. Topping, G. Montenero, R. Montenegro eds, *8th International conference on computational structures technology*, Las Palmas de Gran Canaria-Spain, pp. 325-347, 2006.

LOW-COMPLEXITY ZONOTOPES CAN ENHANCE UNCERTAINTY QUANTIFICATION (UQ)

Olga Kosheleva¹ and Vladik Kreinovich²

¹Department of Teacher Education
University of Texas at El Paso
500 W. University, El Paso, TX 79968, USA
e-mail: olgak@utep.edu

²Department of Computer Science
University of Texas at El Paso
500 W. University, El Paso, TX 79968, USA
e-mail: vladik@utep.edu

Keywords: Uncertainty Quantification, Interval Uncertainty, Zonotopes

Abstract. *In many practical situations, the only information that we know about the measurement error is the upper bound Δ on its absolute value. In this case, once we know the measurement result \tilde{x} , the only information that we have about the actual value x of the corresponding quantity is that this value belongs to the interval $[\tilde{x} - \Delta, \tilde{x} + \Delta]$. How can we estimate the accuracy of the result of data processing under this interval uncertainty? In general, computing this accuracy is NP-hard, but in the usual case when measurement errors are relatively small, we can linearize the problem and thus, make computations feasible. This problem is well studied when data processing results in a single value y , but usually, we use the same measurement results to compute the values of several quantities y_1, \dots, y_n . What is the resulting set of tuples (y_1, \dots, y_n) ? In this paper, we show that this set is a particular case of what is called a zonotope, and that we can use known results about zonotopes to make the corresponding computational problems easier to solve.*

1 FORMULATION OF THE PROBLEM

1.1 Main objective of science and engineering

What do we want? We want to predict what will happen in the future – this is what science does, and what we want to select the actions that will leads to the best possible future – this is, crudely speaking, what engineering is for.

Both to predict the future state of the world and to select the best action, we must have information about the current state of the world, i.e., about the values of all the quantities that characterize this state. This information mostly comes from measurements. To predict the future value y of a quantity or to describe each control parameter y , we use the known relation $y = f(x_1, \dots, x_N)$ between this future value (or control parameter) and the current values of several related quantities x_1, \dots, x_N :

- we measure the values of the quantities x_1, \dots, x_N , and
- we apply the algorithm $f(x_1, \dots, x_N)$ to the results $\tilde{x}_1, \dots, \tilde{x}_N$ of measuring the quantities x_1, \dots, x_N , and return the value $\tilde{y} = f(\tilde{x}_1, \dots, \tilde{x}_N)$.

1.2 Need for uncertainty quantification

Measurements are never absolutely accurate; see, e.g., [9]. The result \tilde{x} of each measurement is, in general, different from the actual (unknown) value x of the corresponding quantity. In other words, the *measurement error* $\Delta x \stackrel{\text{def}}{=} \tilde{x} - x$ is, in general, different from 0.

Since, in general, the measurement result \tilde{x}_i is, in general, different from x_i , our estimate $\tilde{y} = f(\tilde{x}_1, \dots, \tilde{x}_N)$ based on the measurement results is, in general, different from the desired value $y = f(x_1, \dots, x_N)$.

How different can they be? What can we say about the estimation error $\Delta y \stackrel{\text{def}}{=} \tilde{y} - y$? This is very important to know in many practical situations. For example, suppose that we are prospecting for oil, and we estimated that in some location, there is $\tilde{y} = 150$ million tons. What shall we do? It depends on the accuracy of this estimate:

- if $y = 150 \pm 20$, this is very good news; we should dig a well and start producing oil;
- on the other hand, if $y = 150 \pm 200$, then maybe at this location, there is no oil at all; in this case, it is better to perform some additional measurements first, to decrease the risk of wasting money on the expensive well.

Estimating the approximation error Δy based on the known information about the measurement errors Δx_i is one of the main problems of uncertainty quantification.

1.3 Traditional probability-based approach to uncertainty quantification and its limitations

Traditional engineering approach to uncertainty quantification assumes that we know the probability distributions of each measurement error Δx_i [9]. And indeed, in many real-life situations we have this knowledge. However, there are many important practical situations when we do not know these probabilities. To explain why, let us recall where the information about the probabilities comes from.

In the ideal world, for each measuring instrument, we should compare, several times, the measurement result \tilde{x} with the actual value x of the corresponding quantity – and for each

such comparison, compute the measurement error $\Delta x = \tilde{x} - x$. After a sufficient number of measurements, we would get a large sample of values Δx . Based on this sample, we will then be able to find the corresponding probability distribution.

Of course, in reality, we never know the exact actual values of the physical quantities. However, in many cases, there exists another – much more accurate – measuring instrument, whose measurement error Δx_s is much smaller than the measuring error of the tested instrument; such much-more-accurate measuring instruments are known as *standard* ones. In this case, with high accuracy, the value \tilde{x}_s measured by the standard measuring instrument is approximately equal to the actual value, and the difference $\delta x \stackrel{\text{def}}{=} \tilde{x} - \tilde{x}_s$ between the results of the two measurements is approximately equal to the desired measurement error Δx . Thus, we can measure, several times, the same quantities by both measuring instruments, and use the resulting sample to find the probability distribution of the corresponding measurement error.

In many cases, such a calibration is indeed performed, and we get the corresponding probability distributions. However, in many other cases, such a calibration is not done – and thus, we do not know the corresponding probabilities. There are two main reasons why calibration is not done.

The first reason is that sometimes, we use state-of-the-art measuring instruments, for which no other instrument is more accurate. This happens a lot in advanced science: e.g., it would be nice if near the Hubble telescope, we would have a 5 times more accurate instrument – but the Hubble telescope is the best we have. This often happens in applications as well. For example, geophysical companies often use state-of-the-art measuring equipment: this equipment costs money, but it is still cheaper to use such expensive measuring instruments than to risk wasting even more money on, e.g., drilling oil well where there is no oil at all.

The second reason is more mundane: yes, potentially, in a manufacturing plant, we can, in principle, calibrate all the sensors, and get the corresponding probability distributions, but there is a problem. Many sensors are very cheap nowadays: kids play with robotic toys that measure distances to the walls etc. as they go, and these sensors can be bought for a few bucks. However, calibrating each sensor requires access to an expensive accurate measuring instrument – and it would cost several orders of magnitude more than the sensor itself. This is too expensive for a manufacturing plant — which already usually operates at a very low profit margin.

1.4 Enter interval uncertainty

If we do not know probabilities of different values of measurement error Δx , what do we know? For a device to be called a measuring instrument, we need to know at least some upper bound Δ on the absolute value of the measurement error: $|\Delta x| \leq \Delta$. If we do not even know such an upper bound, this means that after a measurement by this instrument, we cannot say anything about the actual value of the measured quantity: it can be as far away from the measurement result as we can imagine. In other words, what such a device would produce is a wild guess, not a measurement result. Thus, such a bound is always produced by the manufacturer of the measuring instrument.

And if we cannot find the probabilities of different values Δx , this upper bound is all we know. In this case, once we know the measurement result \tilde{x} , the only information that we have about the actual value x of the measured quantity is that this value is somewhere in the interval $[\tilde{x} - \Delta, \tilde{x} + \Delta]$. Such uncertainty is naturally called *interval uncertainty*.

1.5 Need for interval computations

Let us go back to the situation when, instead of the actual (ideal) value $y = f(x_1, \dots, x_N)$, we have an estimate $\tilde{y} = f(\tilde{x}_1, \dots, \tilde{x}_N)$ based on the measurement results $\tilde{x}_1, \dots, \tilde{x}_N$. If for each of N measurements, we only know the upper bound Δ_i on the absolute value of the corresponding measurement error Δx_i , then all we know about the actual value y is that it is equal to $f(x_1, \dots, x_N)$ for some $x_i \in [\tilde{x}_i - \Delta_i, \tilde{x}_i + \Delta_i]$.

Thus, all we can say about the value y is that it belongs to the set

$$Y \stackrel{\text{def}}{=} \{f(x_1, \dots, x_N) : x_i \in [\tilde{x}_i - \Delta_i, \tilde{x}_i + \Delta_i] \text{ for all } i\}. \quad (1)$$

For continuous functions $f(x_1, \dots, x_N)$ this set is also an interval. The problem of computing the endpoint of this interval is known as the problem of *interval computations*; see, e.g., [7, 8].

In general, the interval computation problem is NP-hard; see, e.g., [6]. This means that unless $P = NP$ (which most computer scientists do not believe to be true), it is not possible to have a feasible algorithm that solves all particular cases of this problems. However, in many practical situations, there exist efficient algorithms that either compute the desired range Y – or at least compute a good approximation to Y .

1.6 Possibility of linearization

One of the cases when a feasible algorithm for uncertainty quantification is possible is when the measurement errors Δx_i are reasonably small – and usually, they are reasonable small. In this case, we can use one of the main ideas of computations in physics (see, e.g., [2, 12]): expand the corresponding expression in Taylor series in terms of the corresponding small quantities, and keep only linear terms in this expansion. In our case, by definition of the measurement error $\Delta x_i = \tilde{x}_i - x_i$, we have $x_i = \tilde{x}_i - \Delta x_i$, thus:

$$\Delta y = f(\tilde{x}_1, \dots, \tilde{x}_N) - f(x_1, \dots, x_N) = f(\tilde{x}_1, \dots, \tilde{x}_N) - f(\tilde{x}_1 - \Delta x_1, \dots, \tilde{x}_N - \Delta x_N). \quad (2)$$

Expanding the expression in the right-hand side of (2) in Taylor series in terms of Δx_i , we get

$$f(\tilde{x}_1 - \Delta x_1, \dots, \tilde{x}_N - \Delta x_N) = f(\tilde{x}_1, \dots, \tilde{x}_N) - \sum_{i=1}^N c_i \cdot \Delta x_i, \quad (3)$$

where we denoted $c_i \stackrel{\text{def}}{=} \frac{\partial f}{\partial x_i}$. Thus, the formula (2) takes the following form:

$$\Delta y = \sum_{i=1}^N c_i \cdot \Delta x_i. \quad (4)$$

In this linearized case, we can feasibly compute the bounds on Δy . Indeed, since each measurement error Δx_i takes values from the interval $[-\Delta_i, \Delta_i]$, and different measurement errors do not depend on each other, the largest possible value of the sum (1) is attained when each term $c_i \cdot \Delta x_i$ attains the largest possible value. The corresponding linear function $c_i \cdot \Delta x_i$ is increasing when $c_i > 0$ and decreasing when $c_i < 0$. Thus:

- when $c_i > 0$, the largest possible value of the quantity $c_i \cdot \Delta x_i$ is attained when Δx_i is the largest possible, i.e., when $\Delta x_i = \Delta_i$; the resulting largest value of the quantity $c_i \cdot \Delta x_i$ is equal to $c_i \cdot \Delta_i$;

- when $c_i < 0$, the largest possible value of the quantity $c_i \cdot \Delta x_i$ is attained when Δx_i is the smallest possible, i.e., when $\Delta x_i = -\Delta_i$; the resulting largest value of the quantity $c_i \cdot \Delta x_i$ is equal to $-c_i \cdot \Delta_i$.

In both cases, the largest possible value of the quantity $c_i \cdot \Delta x_i$ is equal to $|c_i| \cdot \Delta_i$. Thus, the largest possible value Δ of the sum (4) is equal to

$$\Delta = \sum_{i=1}^N |c_i| \cdot \Delta_i. \quad (5)$$

By using this formula, we can explicitly compute Δ in N steps – i.e., in feasible time.

Comment. While, strictly speaking, this algorithm is feasible, still, in situations when we have a large number N of inputs, it requires a large amount of computation time. It should be mentioned that there exist more efficient algorithms for computing Δ ; see, e.g., [5].

1.7 Need to estimate the joint uncertainty of several data processing results – the main problem that we analyze in this paper

All the above discussions are about estimating a *single* quantity y . In reality, we usually estimate *several* different characteristics y_1, \dots, y_n based on the same data $\tilde{x}_1, \dots, \tilde{x}_N$:

$$\begin{aligned} \tilde{y}_1 &= f_1(\tilde{x}_1, \dots, \tilde{x}_N); \\ &\dots \\ \tilde{y}_n &= f_n(\tilde{x}_1, \dots, \tilde{x}_N). \end{aligned} \quad (6)$$

For example, when we predict weather, we do not just predict temperature at one locations, we predict weather, wind speed and direction, and humidity at several locations.

What is the accuracy of the resulting estimations? In other words, what can we say about the corresponding approximation errors

$$\Delta y_j \stackrel{\text{def}}{=} \tilde{y}_j - f_j(x_1, \dots, x_N). \quad (7)$$

As we have mentioned earlier, in many practical situations, we only know the upper bounds on the measurement errors – so that we have interval uncertainty, for which the only information that we have about each measurement error Δx_i is the upper bound Δ_i on its absolute value: $|\Delta x_i| \leq \Delta_i$. Also, in many practical situations, measurement errors are relatively small – so that we can ignore quadratic (and higher order) terms in the Taylor expansions. Then, we get the linearized formulas

$$\begin{aligned} \Delta y_1 &= c_{1,1} \cdot \Delta x_1 + \dots + c_{1,N} \cdot \Delta x_N; \\ &\dots \\ \Delta y_n &= c_{n,1} \cdot \Delta x_1 + \dots + c_{n,N} \cdot \Delta x_N, \end{aligned} \quad (8)$$

where we denoted $c_{j,i} \stackrel{\text{def}}{=} \frac{\partial f_j}{\partial x_i}$.

For each value y_j , we can use the above techniques and find the interval of possible values of the approximation error Δy_j . However, this is not enough: we also need to also know what combinations of the values (y_1, \dots, y_n) – i.e., equivalently, of the approximation errors $(\Delta y_1, \dots, \Delta y_n)$ – are possible. For example, when we predict weather, in some cases, the

future temperature in two nearby locations can range from 15 to 25 degrees. However, unless these two locations are separated by a mountain – as we have in our city of El Paso – the temperatures at these two locations cannot differ too much: we can have (15, 16) and even, probably, (15, 17), but we cannot have (15, 25). How can we take this into account? How can we describe the corresponding set of tuples (y_1, \dots, y_n) ?

This is the problem that we analyze in this paper.

2 ANALYSIS OF THE PROBLEM: ENTER ZONOTOPES

2.1 General approach to solving problems: reduce and/or reformulate

A usual approach to solving a new problem is to try to find similar problems that have been already solved – or at least for which there are some partial solutions. If we cannot immediately come up with such a similar somewhat-solved problem, a natural idea is to try to reformulate our problems in equivalent terms so that it will be easier to find a similar problem.

2.2 Enter zonotopes

For our problem, this reformulation becomes possible if we reformulate the formulas (8) in vector terms, as

$$\Delta y = c_1 \cdot \Delta x_1 + \dots + c_N \cdot \Delta x_N, \quad (9)$$

where $\Delta x_i \in [-\Delta_i, \Delta_i]$, and we denoted

$$\Delta y \stackrel{\text{def}}{=} (\Delta y_1, \dots, \Delta y_n) \quad (10)$$

and

$$c_i \stackrel{\text{def}}{=} (c_{1,i}, \dots, c_{n,i}). \quad (11)$$

For each j , the set S_i of all the vectors $\Delta x_i \cdot c_i$ for $\Delta x_i \in [-\Delta_i, \Delta_i]$ forms a straight line segment connecting the points $\Delta_i \cdot c_i$ and $-\Delta_i \cdot c_i$. The desired set S of all possible values of the sum (9) is thus equal to the set of all possible sums of vectors from the corresponding sets S_i :

$$S = \{s_1 + \dots + s_N : s_1 \in S_1, \dots, s_N \in S_N\}. \quad (12)$$

In geometry, the construction (12) is known as a *Minkowski sum* of the sets S_1, \dots, S_N ; this sum is denoted by

$$S = S_1 + \dots + S_N. \quad (13)$$

The Minkowski sum of several straight line segments is known as a *zonotope*. Thus, our conclusion is that the desired set of possible values of the tuple $\Delta y = (\Delta y_1, \dots, \Delta y_n)$ is a zonotope.

2.3 Main conclusion of this section

So, to solve our main problem – of estimating the joint uncertainty of several data processing results – we need to be able to deal with zonotopes.

2.4 An interesting observation: every zonotope can be thus represented

We have shown that every set of possible values of the tuple $(\Delta y_1, \dots, \Delta y_n)$ – and thus, of the tuple (y_1, \dots, y_n) – is a zonotope. Let us show that, vice versa, every zonotope can be thus represented. Indeed, in the above representation, we use straight-line segments centered at

0. Every straight-line segment T_i can be represented as the sum $T_i = m_i + S_i$ of its midpoint m_i and a segment $S_i \stackrel{\text{def}}{=} T_i - m_i$ centered at 0, i.e., a segment connecting one of its endpoints $c_i = (c_{i,1}, \dots, c_{i,N})$ with the opposite endpoint $-c_i = (-c_{i,1}, \dots, -c_{i,N})$. Thus, each zonotope

$$T = T_1 + \dots + T_N \quad (14)$$

can be represented as

$$T = (m_1 + \dots + m_N) + (S_1 + \dots + S_N). \quad (15)$$

The set $S_1 + \dots + S_n$ can be interpreted as the set of possible approximation errors for a data processing algorithm

$$f_i(x_1, \dots, x_N) = c_{i,1} \cdot x_1 + \dots + c_{i,N} \cdot x_n, \quad (16)$$

when we take $\Delta_1 = \dots = \Delta_N = 1$. Thus, every zonotope can indeed be represented as the set of possible tuples (y_1, \dots, y_n) for some data processing algorithm.

2.5 Historical comment

The idea of using zonotopes was described, e.g., in [10], where it is shown that for a *specific* data processing algorithm – namely, for the least square estimation under interval uncertainty – the resulting set of possible tuples is a zonotope. In this paper, we show that this is true for *all* data processing algorithms – and we also show that, vice versa, every zonotope can be thus represented.

3 HOW TO DEAL WITH ZONOTOPES: WHAT IS KNOWN, WHAT WE PROPOSE, AND WHAT ARE THE REMAINING OPEN PROBLEMS

3.1 What is known

In computational geometry, there are several efficient algorithms for dealing with zonotopes; see, e.g., [3, 4]. Some of these algorithms have been efficiently used in [10].

3.2 What is the difficulty with the known algorithms

The main problem with these algorithms is that the exact description of the uncertainty-related zonotope in an n -dimensional space requires as many n -dimensional parameters c_i as there are measured quantities x_1, \dots, x_N . In many practical problems, e.g., in seismology, N is in thousands, so this description becomes difficult to process.

3.3 What we propose: general idea

The possibility to make computations easier comes from the fact that the number n of desired properties y_1, \dots, y_n is much smaller than N . So, to speed up computations, we propose to use the known results [1, 11] about the possibility of approximating zonotopes with “low-complexity” sets, i.e., sets determined by a much smaller number of n -dimensional parameters.

By approximating a set S , we mean, as usual, producing a set A for which, for some small number δ :

- each element $s \in S$ is δ -close to some element $a \in A$, and
- each element $a \in A$ is δ -close to some element $s \in S$.

In mathematics, this closeness is usually described by saying that the Hausdorff distance $d_H(A, S)$ between the sets S and A is smaller than or equal to δ , where the Hausdorff distance $d_H(A, S)$ is defined as the small distance for which the above two conditions are true.

We can also say that the set A approximates the set S with *relative accuracy* ε if $d_H(A, S) \leq \delta \cdot \text{diam}(S)$, where the diameter $\text{diam}(S)$ is defined as the largest distance between two points from the set S – this is a natural generalization of the width of an interval and the diameter of a disk or of a sphere to general sets.

3.4 First simplifying result

The first simplifying result from [1, 11] is that each n -dimensional zonotope S can be approximated, with any given relative accuracy $\varepsilon > 0$, by a “low-complexity” zonotope, which is the sum of $N' = c(\varepsilon) \cdot n \cdot (\log(n))^3$ segments for some constant c depending on ε .

Since $n \ll N$, the new number of n -dimensional vector parameters is much smaller than N – thus, the problem becomes easier to handle.

3.5 Second simplifying result

The second simplifying result from [1, 11] is that each symmetric convex polyhedron – in particular, each zonotope – can be approximated, with any given relative accuracy $\varepsilon > 0$, by a convex polyhedron with “low” number of vertices $v_1, \dots, v_{N'}$ – namely, by a polyhedron for which the number N' is also bounded, from above, by the value $c(\varepsilon) \cdot n \cdot (\log(n))^d \ll N$, for some small constant d .

In this case, the approximating set A is the convex combination of these vertices v_j . In other words, all elements $a = (a_1, \dots, a_n)$ from the approximating set A have the form

$$a = c_1 \cdot v_1 + \dots + c_{N'} \cdot v_{N'} \quad (17)$$

where $c_i \geq 0$ and

$$c_1 + \dots + c_{N'} = 1. \quad (18)$$

3.6 What are the remaining open problems

The results from [1, 11] (that we propose to use) are effective – they drastically reduce the complexity of the corresponding problem – but at present, they are not supported by efficient computational algorithms for producing the corresponding approximations. These results are still useful – we spend time on computing the approximation only once, and then, we can enjoy the benefits of this reduction for every single way we want to process this set.

However, it would be nice to have efficient algorithms for producing the corresponding approximations. Designing such efficient algorithms is the main open problem that we want to emphasize. Hopefully, the fact that – as we have shown – such algorithms will be very helpful:

- not just in somewhat obscure computational geometry problems,
- but also in generic problems of uncertainty quantification

will hopefully encourage researchers to design such algorithms.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation grants 1623190 (A Model of Change for Preparing a New Generation for Professional Practice in Computer Science), and HRD-1834620 and HRD-2034030 (CAHSI Includes).

It was also supported by the program of the development of the Scientific-Educational Mathematical Center of Volga Federal District No. 075-02-2020-1478.

REFERENCES

- [1] J. Bourgain, J. Lindenstrauss, and V. Milman, Approximation of zonoids by zonotopes, *Acta Mathematica*, **162**(1–2), 73–141, 1989.
- [2] R. Feynman, R. Leighton, and M. Sands, *The Feynman Lectures on Physics*, Addison Wesley, Boston, Massachusetts, 2005.
- [3] J.E. Goodman and J. O’Rourke (eds.), *Handbook of Discrete and Computational Geometry*, CRC Press, Boca Raton, Florida, 1997.
- [4] P.M. Gruber and J.M. Willis (eds.), *Handbook of Convex Geometry*, Elsevier, Amsterdam, 1993.
- [5] V. Kreinovich and S. Ferson, A new Cauchy-based black-box technique for uncertainty in risk analysis, *Reliability Engineering and Systems Safety*, **85**(1–3), 267–279, 2004.
- [6] V. Kreinovich, A. Lakeyev, J. Rohn, and P. Kahl, *Computational Complexity and Feasibility of Data Processing and Interval Computations*, Kluwer, Dordrecht, 1998.
- [7] G. Mayer, *Interval Analysis and Automatic Result Verification*, de Gruyter, Berlin, 2017.
- [8] R.E. Moore, R.B. Kearfott, and M.J. Cloud, *Introduction to Interval Analysis*, SIAM, Philadelphia, 2009.
- [9] S.G. Rabinovich, *Measurement Errors and Uncertainties: Theory and Practice*, Springer, New York, 2005.
- [10] S. Schön and H. Kurtterer, Using zonotopes for overestimation-free interval least squares – some geodetic applications, *Reliable Computing*, **11**, 137–155, 2004.
- [11] T. Tao, Exploring the toolkit of Jean Bourgain, *Bulletin on the American Mathematical Society*, **58**, 155–171, 2021.
- [12] K.S. Thorne and R.D. Blandford, *Modern Classical Physics: Optics, Fluids, Plasmas, Elasticity, Relativity, and Statistical Physics*, Princeton University Press, Princeton, New Jersey, 2017.

BOUNDS OF RELIABILITY FUNCTION FOR STRUCTURAL SYSTEMS SUBJECTED TO A SET OF RECORDED ACCELEROGRAMS

Federica Genovese¹, Giuseppe Muscolino², and Alba Sofi³

¹ Department of Engineering, University of Messina Villaggio S. Agata, 98166 Messina, Italy
e-mail: federica.genovese@unime.it

² Department of Engineering, University of Messina, Inter-University Centre of Theoretical and Experimental Dynamics, Villaggio S. Agata, 98166 Messina, Italy
e-mail: gmuscolino@unime.it

³ Department of Architecture and Territory (dArTe), University “Mediterranea” of Reggio Calabria, Inter-University Centre of Theoretical and Experimental Dynamics, via dell’Università 25, 89124 Reggio Calabria, Italy
e-mail: alba.sofi@unirc.it

Abstract

The present study focuses on reliability analysis of linear discretized structures subjected to ground motion acceleration modeled as a zero-mean Gaussian stationary random process fully characterized by an imprecise power spectral density (PSD) function i.e. with interval parameters. The bounds of such interval parameters are determined by analyzing a large set of accelerograms recorded on rigid soil deposits. To discard outliers from the set of accelerograms, the Chauvenet’s Criterion is applied. Then, to assess structural safety, the imprecise PSD function of ground motion acceleration is incorporated into the formulation of the classical first-passage problem. Due to imprecision of the excitation, the reliability function of the selected extreme value response process turns out to have an interval nature. It is shown that the bounds of the interval reliability function can be readily evaluated by exploring suitable combinations of the endpoints of the interval spectral moments of the selected response process identified relying on structural dynamic properties and taking advantage of the dependency of the proposed imprecise PSD function on three interval parameters only.

Keywords: Ground Motion Accelerograms, Gaussian Stationary Processes, Imprecise Power Spectral Density, Interval Analysis, Interval Solution of First-Passage Problem.

1 INTRODUCTION

In seismically active regions, seismic forces represent the most critical actions on structural systems. These forces are closely related to ground motion accelerations due to earthquake, which vary during the seismic event. It follows that seismic accelerations are the main functions to be analyzed in earthquake engineering.

Although the seismic accelerograms should be considered as samples of zero-mean non-stationary Gaussian random processes, within the *strong-motion time duration*, earthquake ground motions can be reasonably modeled as samples of zero-mean stationary Gaussian stochastic processes. The *power spectral density (PSD)* function, which fully characterizes the stationary model, must account for the site properties as well as for the dominant frequency of the ground motion [1]. Indeed, it is widely recognized that the parameters of the *PSD* function are strongly influenced by the geotechnical characteristics of the soil deposits on which the seismic accelerograms are recorded [2]. Stationary models have also been developed to obtain sets of acceleration time histories consistent with response spectra (see e.g., [3, 4]). Recently, a new model of the *PSD* function depending on the frequency of *peaks* and of *zero-level up-crossings* as well as on the *total intensity* of recorded accelerograms has been proposed by Muscolino et al. [5]. Unfortunately, the main parameters of the *PSD* function are very often only vaguely known.

It is now widely recognized that, when available information on the various sources of uncertainty is vague, incomplete or fragmentary, the use of non-probabilistic approaches (see e.g., [6, 7]) is more appropriate to model non-deterministic properties. Accordingly, in this paper, the main parameters characterizing the seismic acceleration spectrum are modeled as interval variables whose bounds are estimated by analyzing a set of accelerograms, recorded by stations located on rock subsoil. In particular, according to the international seismic code EC8 [8], accelerograms recorded on soil class A, in areas of high and moderate seismicity (surface-wave magnitude $M_s > 5.5$, Type 1 spectrum), are here analysed.

Since the total number of both *zero-level up-crossings* and *peaks*, and the *total energy* in the *strong-motion time region* are different from one accelerogram to another, in order to define the *PSD* function parameters of the model developed by Muscolino et al. [5], and to quantify the uncertainty affecting ground motion representation, the spectral content of accelerograms in such time region is studied. In particular, the histograms associated with the main properties characterizing the recorded accelerograms are analyzed once the *Chauvenet's Criterion* [9] is iteratively applied [10] to discard outliers. As a final outcome of the statistical analysis of the selected accelerograms, the ranges of variability of the main parameters of the *PSD* function are determined. Thus, it is shown that the *PSD* function, representative of accelerograms recorded on soils with specific geotechnical characteristics, can be appropriately modeled as a function of interval parameters [11, 12] whose ranges reflect the main properties of the excitation. A notable feature of the proposed model of the *imprecise PSD* function is that it depends on three interval parameters only.

Once earthquake excitation within the *strong-motion time region* is modeled as a zero-mean Gaussian stationary random process, fully characterized by an *imprecise PSD* function, in this study the attention is focused on reliability analysis of seismically excited linear discretized structures. In the framework of random vibration theory, under the assumption that a structure fails as soon as the generic structural response process of interest (e.g., displacement, strain or stress) firstly exceeds a prescribed safe domain within a specified time interval, the probability of failure is usually identified with the *first-passage probability*. The *reliability function* is the complement to unity of the *first-passage probability*. It represents the probability that the maximum value of the generic structural response process of interest is equal to or

less than the critical level. Here, the Vanmarcke model [13] is adopted. According to this model, the *reliability function* depends on the first three *spectral moments* of the selected stationary random response process of interest. As a result of the imprecision of the *PSD* function of seismic excitation, the *spectral moments* as well as the *reliability function* turn out to be interval functions too. The *lower bound (LB)* and *upper bound (UB)* of the *interval reliability function*, which define a p-box [14] representative of structural performance, and of the *interval failure probability* can be readily evaluated by exploring selected combinations of the three interval parameters characterizing the imprecise *PSD* function. Such combinations are identified as those providing the bounds of the spectral moments of the selected response process and are shown to depend on the dynamic characteristics of the structure.

Seismically excited linear oscillators are analyzed to investigate the influence of imprecision of earthquake acceleration spectrum on structural safety as well as to demonstrate the accuracy of the proposed procedure for the evaluation of the *LB* and *UB* of the *interval reliability function*.

2 ESTIMATION OF PARAMETERS OF THE PSD FUNCTION BY ANALYSING RECORDED ACCELEROGRAMS

2.1 Statistical analysis of accelerograms recorded on rigid soil deposits

In order to characterize uncertainties affecting the *power spectral density (PSD)* function of seismic action, a set of 44 accelerograms, recorded on rigid soil deposits, is downloaded from the *Engineering Strong Motion* database (*ESM*) [15]. All the accelerograms considered in this work are recorded by stations located on rock subsoil (soil class A type 1 [8]), in areas of high and moderate seismicity (surface-wave magnitude $M_s > 5.5$, Type 1 spectrum). Attention is focused on the following properties characterizing the accelerograms: *i*) the *strong-motion time duration*, T_D , defined as the portion of time between the 5% and 95% of the cumulative energy; *ii*) the frequency of *zero-level up-crossings*, ν_N ; *iii*) the frequency of the *peaks*, ν_p ; *iv*) the normalized *total intensity*, $\sigma_{\bar{v}_g}^2$; *v*) the predominant circular frequency, Ω_0 ; *vi*) the frequency bandwidth, ρ_0 . By analyzing statistically the set of recorded accelerograms, the aforementioned properties are characterized through the following quantities reported in Table 1: the *lower bound (LB)*, the *upper bound (UB)*, the *midpoint (mid)*, the *normalized deviation amplitude (dev/mid)*, defined as $(UB-LB)/(UB+LB)$ [11], the mean-value (*MV*), the standard deviation (*SD*), the skewness (*skew*) and the kurtosis (*kurt*) [16].

In the framework of the statistical analysis of experimental data, the first step is to identify and discard outliers. For this purpose, in the present study the *Chauvenet's Criterion* is applied iteratively until the number of complying accelerograms remains stable [10]. Discarding outliers, a set of 20 accelerograms for rigid soil deposits into high and moderate seismicity regions is obtained. The main parameters characterizing such set of accelerograms are reported in Table 2. Notice that the iterative application of the *Chauvenet's Criterion* leads to smaller scatters of the parameters, as can be inferred from the lower values of the normalized deviation amplitudes. However, further statistical analyses are needed to reduce the range of variability of the variance $\sigma_{\bar{v}_g}^2$ which exhibits huge fluctuations around the midpoint value.

As will be shown through numerical results, such fluctuations yield a very wide range of the *reliability function* of seismically excited structures whose performance, therefore, might cover high safety levels as well high failure probabilities.

	<i>LB</i>	<i>UB</i>	<i>mid</i>	<i>dev/mid</i>	<i>MV</i>	<i>SD</i>	<i>skew</i>	<i>kurt</i>
T_D [s]	3.690	19.470	11.580	0.681	10.894	3.850	0.144	2.426
ν_N [Hz]	1.704	13.980	7.842	0.783	6.268	2.651	1.329	4.287
ν_P [Hz]	4.158	20.858	12.508	0.668	10.370	3.664	1.077	3.879
$\sigma_{\ddot{U}_g}^2$ [m ² /s ⁴]	0.010	2.062	1.036	0.990	0.181	0.420	3.741	16.392
Ω_0 [rad/s]	10.708	87.838	49.273	0.783	39.380	16.654	1.329	4.287
ρ_0 [rad/s]	6.216	36.583	21.399	0.709	18.644	6.788	1.147	3.897

Table 1. Main characteristics of selected accelerograms.

	<i>LB</i>	<i>UB</i>	<i>mid</i>	<i>dev/mid</i>	<i>MV</i>	<i>SD</i>	<i>skew</i>	<i>kurt</i>
T_D [s]	5.025	19.470	12.248	0.598	12.523	3.883	-0.004	2.252
ν_N [Hz]	3.746	7.802	5.774	0.351	5.591	1.166	0.192	2.031
ν_P [Hz]	6.006	11.406	8.706	0.310	8.955	1.629	-0.132	1.672
$\sigma_{\ddot{U}_g}^2$ [m ² /s ⁴]	0.010	0.092	0.051	0.804	0.036	0.028	0.958	2.457
Ω_0 [rad/s]	23.534	49.024	36.279	0.351	35.127	7.327	0.192	2.031
ρ_0 [rad/s]	11.145	20.899	16.022	0.304	16.442	2.897	-0.240	1.582

 Table 2. Main characteristics of selected accelerograms evaluated through the iterative application of the *Chauvenet's Criterion*.

2.2 Interval PSD function

In this paper, the seismic acceleration, $\ddot{U}_g(t)$, is characterized by the unimodal one-sided *power spectral density (PSD)* function recently proposed by Muscolino et al. [5]:

$$G_{\ddot{U}_g}(\omega) = \sigma_{\ddot{U}_g}^2 \beta_0 \left(\frac{\omega^2}{\omega^2 + \omega_H^2} \right) \left(\frac{\omega_L^4}{\omega^4 + \omega_L^4} \right) G_0^{(CP)}(\omega) \quad (1)$$

where ω_L and ω_H are the control frequencies of the second-order low pass and first-order high pass Butterworth filters, respectively, $G_0^{(CP)}(\omega)$ is a unimodal one-sided *PSD* function, which can be viewed as the linear combination of the displacement and velocity responses of a second-order oscillator subjected to two statistically independent Gaussian white noise processes [17], given by:

$$G_0^{(CP)}(\omega) = \frac{\rho_0}{\pi} \left[\frac{1}{\rho_0^2 + (\omega + \Omega_0)^2} + \frac{1}{\rho_0^2 + (\omega - \Omega_0)^2} \right]. \quad (2)$$

In the previous equation, ρ_0 and Ω_0 are the circular frequency bandwidth and the predominant circular frequency of the filtered stationary processes, respectively. Furthermore, in Eq.(1) the coefficient β_0 is evaluated in such a way that the process $\ddot{U}_g(t)$ possesses unitary variance [5]. To define the main parameters of this model, the total number of both *zero-level up-crossings* and *peaks*, and the *total energy* have to be evaluated [5]. However, as shown in the previous section, these quantities are different from one accelerogram to another. It follows that, in order to define the *PSD* function parameters of the model proposed by Muscolino et al. [5], and to quantify the uncertainty affecting ground motion representation, the spectral content of accelerograms in the *strong-motion time regions* has to be analyzed.

In this paper, the above described uncertainties affecting the *PSD* function of ground motion acceleration are modelled via *interval analysis* [11, 12]. In particular, the so-called *Improved Interval Analysis* [18] is adopted. Accordingly, the generic interval variable x_i^I is defined as follows:

$$x_i^I = [\underline{x}_i, \bar{x}_i] \equiv x_{\text{mid},i}(1 + \alpha_i^I) = x_{\text{mid},i}(1 + \Delta\alpha_i \hat{e}_i^I) \quad (3)$$

where the symbols \underline{x}_i and \bar{x}_i denote the *LB* and *UB* of the interval, respectively; the apex *I* characterizes the interval variables; $\hat{e}_i^I = [-1, 1]$ is the so-called *i*-th *extra unitary interval (EUI)*, associated with the *i*-th interval variable [18]. In Eq.(3), $x_{\text{mid},i}$ and $\Delta\alpha_i$ are the *mid-point value* (or mean) and the normalized *deviation amplitude* (or radius) of x_i^I , given, respectively, by:

$$x_{\text{mid},i} = \frac{\underline{x}_i + \bar{x}_i}{2}; \quad \Delta\alpha_i = \frac{\Delta x_i}{x_{\text{mid},i}} = \frac{\bar{\alpha}_i - \underline{\alpha}_i}{2} > 0 \quad (4a,b)$$

where $\Delta x_i = (\bar{x}_i - \underline{x}_i)/2$ is the *deviation amplitude* of x_i^I ; $\Delta\alpha_i$ represents the *deviation amplitude* (or radius) of the dimensionless interval fluctuation $\alpha_i^I = \Delta\alpha_i \hat{e}_i^I$ around $x_{\text{mid},i}$ such that $\Delta\alpha_i < 1$.

Notice that, in the framework of interval symbolism, a generic interval-valued function f and a generic interval-valued matrix/vector function \mathbf{A} of the interval variables α_i^I , ($i = 1, 2, \dots$) or β_j^I ($j = 1, 2, \dots$) and of classical, not interval, quantities b_k ($k = 1, 2, \dots$) and c_ℓ ($\ell = 1, 2, \dots$) will be denoted in equivalent form, respectively, as:

$$f^I(b_1, b_2, \dots) \equiv f(b_1, b_2, \dots, \alpha_1^I, \alpha_2^I, \dots); \quad \mathbf{A}^I(c_1, c_2, \dots) \equiv \mathbf{A}(c_1, c_2, \dots, \beta_1^I, \beta_2^I, \dots). \quad (5a,b)$$

The *PSD* function introduced in Eq.(1) depends on three parameters: the predominant circular frequency, Ω_0 ; the circular frequency bandwidth, ρ_0 ; and the variance, $\sigma_{\ddot{U}_g}^2$. These parameters are herein treated as interval variables i.e.:

$$\begin{aligned}\Omega_0^I &= [\underline{\Omega}_0, \bar{\Omega}_0] = \Omega_{0,\text{mid}}(1 + \alpha_{\Omega_0}^I) = \Omega_{0,\text{mid}}(1 + \Delta\alpha_{\Omega_0} \hat{e}_{\Omega_0}^I); \\ \rho_0^I &= [\underline{\rho}_0, \bar{\rho}_0] = \rho_{0,\text{mid}}(1 + \alpha_{\rho_0}^I) = \rho_{0,\text{mid}}(1 + \Delta\alpha_{\rho_0} \hat{e}_{\rho_0}^I); \\ (\sigma_{\ddot{U}_g}^2)^I &= [\underline{\sigma}_{\ddot{U}_g}^2, \bar{\sigma}_{\ddot{U}_g}^2] = \sigma_{\ddot{U}_g,\text{mid}}^2(1 + \alpha_{\sigma_{\ddot{U}_g}^2}^I) = \sigma_{\ddot{U}_g,\text{mid}}^2(1 + \Delta\alpha_{\sigma_{\ddot{U}_g}^2} \hat{e}_{\sigma_{\ddot{U}_g}^2}^I).\end{aligned}\quad (6a-c)$$

The midpoints, $\Omega_{0,\text{mid}}$, $\rho_{0,\text{mid}}$, $\sigma_{\ddot{U}_g,\text{mid}}^2$, and the normalized deviation amplitudes, $\Delta\alpha_{\Omega_0}$, $\Delta\alpha_{\rho_0}$, $\Delta\alpha_{\sigma_{\ddot{U}_g}^2}$, are assumed equal to those listed in the fourth and fifth columns of Table 2.

Under this assumption, the *PSD* function representative of accelerograms recorded on soils with specific geotechnical characteristics is consistently modelled as an interval function. This implies that the seismic spectra with deterministic parameters proposed in literature provide only indicative models of recorded accelerograms in seismic areas which may differ from the actual ones. Note that, since the *PSD* function in Eq. (1) depends linearly on $(\sigma_{\ddot{U}_g}^2)^I$, this variable could be set a posteriori as a function of site seismic hazard or as a function of the expected peak ground acceleration. Based on this observation, the following expression of the interval extension of the *PSD* function given in Eq.(1) is assumed:

$$G_{\ddot{U}_g}(\omega; \Omega_0^I, \rho_0^I, (\sigma_{\ddot{U}_g}^2)^I) \equiv (\sigma_{\ddot{U}_g}^2)^I \tilde{G}_{\ddot{U}_g}^I(\omega) \quad (7)$$

with

$$\tilde{G}_{\ddot{U}_g}^I(\omega) \equiv \tilde{G}_{\ddot{U}_g}(\omega; \Omega_0^I, \rho_0^I) = \beta_0^I \left(\frac{\omega^2}{\omega^2 + (\omega_H^I)^2} \right) \left(\frac{(\omega_L^I)^4}{\omega^4 + (\omega_L^I)^4} \right) G_0^{(CP)}(\omega; \Omega_0^I, \rho_0^I) \quad (8)$$

where [5]

$$\beta_0^I \equiv \beta_0(\Omega_0^I, \rho_0^I); \quad \omega_H^I \equiv \omega_H(\Omega_0^I) = 0.1 \Omega_0^I; \quad \omega_L^I \equiv \omega_L(\Omega_0^I, \rho_0^I) = \Omega_0^I + 0.8 \rho_0^I. \quad (9a-c)$$

For illustration purposes, Fig. 1 shows the realizations of the *imprecise PSD* function $\tilde{G}_{\ddot{U}_g}^I(\omega)$ pertaining to the extreme values of the interval parameters Ω_0^I and ρ_0^I along with the nominal spectrum. Notice that imprecision causes a significant variation of the *PSD* function.

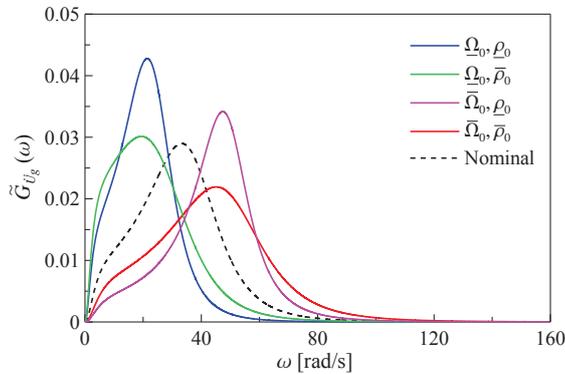


Figure 1: Realizations of the *imprecise PSD* function of ground motion acceleration $\tilde{G}_{\ddot{U}_g}^I(\omega)$.

3 EQUATIONS GOVERNING THE PROBLEM

3.1 Interval equations of motion

Consider now a structural system subjected to the normalized interval stationary process $\ddot{U}_g^I(t) \equiv \ddot{U}_g(t; \Omega_0^I, \rho_0^I) = \ddot{U}_g(t; \Omega_0^I, \rho_0^I, \sigma_{\ddot{U}_g}^I) / \sigma_{\ddot{U}_g}^I$, whose equations of motion can be written as:

$$\mathbf{M} \ddot{\mathbf{U}}^I(t) + \mathbf{C} \dot{\mathbf{U}}^I(t) + \mathbf{K} \mathbf{U}^I(t) = -\mathbf{M} \boldsymbol{\tau} \ddot{U}_g^I(t) \quad (10)$$

where \mathbf{M} , \mathbf{C} and \mathbf{K} are the $n \times n$ mass, damping and stiffness matrices, respectively; $\boldsymbol{\tau}$ is the n -array listing the influence coefficients; $\mathbf{U}^I(t) \equiv \mathbf{U}(t; \Omega_0^I, \rho_0^I)$ is the interval stationary Gaussian vector process of normalized displacements; and a dot over a variable denotes differentiation with respect to time t . Under the assumption of classically damped system, the equations of motion can be decoupled by applying modal analysis. To this aim, the following modal coordinate transformation is introduced:

$$\mathbf{U}^I(t) = \boldsymbol{\Phi} \tilde{\mathbf{Q}}^I(t) = \sum_{j=1}^s \boldsymbol{\phi}_j \tilde{Q}_j^I(t) \Rightarrow \tilde{U}_i^I(t) = \sum_{j=1}^s \phi_{ij} \tilde{Q}_j^I(t) \quad (11)$$

where $\tilde{\mathbf{Q}}^I(t)$ is the vector of modal coordinates; $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1 \quad \boldsymbol{\phi}_2 \quad \dots \quad \boldsymbol{\phi}_s]$ is the modal matrix, of order $n \times s$, collecting the $s \leq n$ eigenvectors $\boldsymbol{\phi}_j$, normalized with respect to the mass matrix, \mathbf{M} , solutions of the following eigenproblem:

$$\mathbf{K}^{-1} \mathbf{M} \boldsymbol{\Phi} = \boldsymbol{\Phi} \boldsymbol{\Omega}^{-2}; \quad \boldsymbol{\Phi}^T \mathbf{M} \boldsymbol{\Phi} = \mathbf{I}_s. \quad (12)$$

In the previous equation, $\boldsymbol{\Omega}$ is a diagonal matrix listing the undamped natural circular frequencies ω_j ; \mathbf{I}_s is the identity matrix of order s ; and the superscript T denotes the transpose operator. Once the modal matrix $\boldsymbol{\Phi}$ is evaluated, by applying the coordinate transformation (11) to Eq.(10), the following set of decoupled interval second-order ordinary differential equations is obtained:

$$\ddot{\tilde{\mathbf{Q}}}^I(t) + \boldsymbol{\Xi} \dot{\tilde{\mathbf{Q}}}^I(t) + \boldsymbol{\Omega}^2 \tilde{\mathbf{Q}}^I(t) = \mathbf{p} \ddot{U}_g^I(t) \quad (13)$$

where $\boldsymbol{\Xi} = \boldsymbol{\Phi}^T \mathbf{C} \boldsymbol{\Phi}$ is the generalized damping matrix, and $\mathbf{p} = -\boldsymbol{\Phi}^T \mathbf{M} \boldsymbol{\tau}$ is the vector of participation factors. For classically damped structures, the modal damping matrix $\boldsymbol{\Xi}$ is a diagonal matrix listing the quantities $2\xi_j \omega_j$, being ξ_j the modal damping ratio.

3.2 Characterization of interval stochastic response processes

The interval zero-mean stationary Gaussian stochastic response process $\mathbf{U}^I(t)$, ruled by the equations of motion in Eq.(10), is completely characterized in the frequency domain by the interval one-sided PSD function matrix, $\tilde{\mathbf{G}}_{\mathbf{UU}}^I(\omega) \equiv \tilde{\mathbf{G}}_{\mathbf{UU}}(\omega; \Omega_0^I, \rho_0^I)$, given by:

$$\tilde{\mathbf{G}}_{\mathbf{UU}}^I(\omega) = \boldsymbol{\Phi} \tilde{\mathbf{G}}_{\mathbf{QQ}}^I(\omega) \boldsymbol{\Phi}^T = \boldsymbol{\Phi} \mathbf{H}_M^*(\omega) \mathbf{p} \mathbf{p}^T \mathbf{H}_M^T(\omega) \boldsymbol{\Phi}^T \tilde{G}_{U_g}^I(\omega) \quad (14)$$

where the asterisk means complex conjugate; $\tilde{\mathbf{G}}_{\mathbf{QQ}}^I(\omega)$ is the interval PSD function matrix of the modal coordinate vector $\tilde{\mathbf{Q}}^I(t)$; $\mathbf{H}_M(\omega)$ is the modal frequency response function (FRF) matrix, defined as:

$$\mathbf{H}_M(\omega) = \left[-\omega^2 \mathbf{I}_s + j\omega \mathbf{\Xi} + \mathbf{\Omega}^2 \right]^{-1} \quad (15)$$

with $j = \sqrt{-1}$ denoting the imaginary unit.

The interval one-sided *PSD* function matrix, which fully characterizes the stochastic response vector process of geometric interval displacements $\mathbf{U}^I(t)$, can be derived from Eq.(14) as $\mathbf{G}_{\mathbf{U}\mathbf{U}}^I(\omega) \equiv \mathbf{G}_{\mathbf{U}\mathbf{U}}(\omega) \left(\omega; \Omega_0^I, \rho_0^I, (\sigma_{\tilde{U}_g}^2)^I \right) = (\sigma_{\tilde{U}_g}^2)^I \tilde{\mathbf{G}}_{\mathbf{U}\mathbf{U}}^I(\omega)$.

The generic response quantity of interest, $Y_h^I(t)$ (e.g., displacement, strain or stress at a critical point) can be determined from the knowledge of the interval displacement vector $\tilde{\mathbf{U}}^I(t)$ as follows:

$$Y_h^I(t) \equiv Y_h \left(t; \Omega_0^I, \rho_0^I, (\sigma_{\tilde{U}_g}^2)^I \right) = \mathbf{q}_h^T \mathbf{U}^I(t) = \sigma_{\tilde{U}_g}^I \mathbf{q}_h^T \tilde{\mathbf{U}}^I(t) \quad (16)$$

where \mathbf{q}_h is a vector collecting the combination coefficients relating the response process $Y_h^I(t)$ to $\mathbf{U}^I(t)$.

The complete probabilistic characterization of the interval stationary Gaussian random response process in Eq.(16) requires the knowledge of the interval one-sided *PSD* function $G_{Y_h Y_h}^I(\omega)$ of $Y_h^I(t)$ defined as follows:

$$G_{Y_h Y_h}^I(\omega) = (\sigma_{\tilde{U}_g}^2)^I \mathbf{q}_h^T \tilde{\mathbf{G}}_{\mathbf{U}\mathbf{U}}^I(\omega) \mathbf{q}_h \quad (17)$$

where $\tilde{\mathbf{G}}_{\mathbf{U}\mathbf{U}}^I$ is given by Eq. (14).

4 BOUNDS OF INTERVAL RELIABILITY FUNCTION

Under the assumption that a structure fails as soon as the response at a critical location exceeds a prescribed safe domain for the first time, the probability of failure is usually identified with the *first-passage probability*, i.e. the probability that the *extreme value* random process for the generic structural response process of interest (e.g., displacement, strain or stress) firstly exceeds the safety bounds within a specified time interval $[0, T]$.

For the generic response quantity $Y_h^I(t)$ (see Eq.(16)) of a structure under imprecise seismic excitation, the *extreme value* random process, over the time interval $[0, T]$, has an interval nature and is mathematically defined as:

$$Y_{\max, h}^I(T) \equiv Y_{\max, h} \left(T; \Omega_0^I, \rho_0^I, (\sigma_{\tilde{U}_g}^2)^I \right) = \max_{0 \leq t \leq T} \left| Y_h \left(t; \Omega_0^I, \rho_0^I, (\sigma_{\tilde{U}_g}^2)^I \right) \right| \quad (18)$$

where the symbol $|\bullet|$ denotes absolute value.

The *interval cumulative distribution function (ICDF)*, $L_{Y_{\max, h}^I}^I(b, T)$, of the *extreme value* random process, often called in literature *interval reliability function*, represents the probability that $Y_{\max, h}^I(T)$ is equal to or less than the barrier level b within the time interval $[0, T]$. In random vibration theory, the evaluation of the *CDF* function is a quite challenging task. Even in the simplest case of the stationary response of a *SDOF* linear oscillator under zero-mean Gaussian white noise, the exact solution of this problem is not available. Hence, several approximate techniques have been proposed in literature, which differ in generality, complexity

and accuracy. In the framework of approximate methods, the *first-passage* failure criterion by Vanmarcke [13] is usually adopted. According to this criterion, the *ICDF* function of the *extreme value* random process $Y_{\max,h}^I(T)$ can be written as:

$$L_{Y_{\max,h}^I}^I(b, T) \equiv L_{Y_{\max,h}^I}(b, T; \Omega_0^I, \rho_0^I, (\sigma_{\tilde{U}_g}^2)^I) \cong \exp \left[-T \frac{1}{\pi} \sqrt{\frac{\tilde{\lambda}_{2,Y_h}^I}{\tilde{\lambda}_{0,Y_h}^I}} \left[\frac{1 - \exp \left(-b \left(\tilde{\delta}_{Y_h}^I \right)^{1.2} \sqrt{\frac{\pi}{2(\sigma_{\tilde{U}_g}^2)^I \tilde{\lambda}_{0,Y_h}^I}} \right)}{\exp \left(\frac{b^2}{2(\sigma_{\tilde{U}_g}^2)^I \tilde{\lambda}_{0,Y_h}^I} \right) - 1} \right] \right] \quad (19)$$

where $\tilde{\delta}_{Y_h}^I$ is the so-called *interval bandwidth parameter* of the stationary process $\tilde{Y}_h^I(t) = Y_h^I(t) / \sigma_{\tilde{U}_g}^I$ defined as:

$$\tilde{\delta}_{Y_h}^I \equiv \tilde{\delta}_{Y_h}^I(\Omega_0^I, \rho_0^I) = \sqrt{1 - \frac{\operatorname{Re}\{\tilde{\lambda}_{1,Y_h}^I\}^2}{\tilde{\lambda}_{0,Y_h}^I \tilde{\lambda}_{2,Y_h}^I}}. \quad (20)$$

In the previous equations, $\tilde{\lambda}_{0,Y_h}^I \equiv \tilde{\lambda}_{0,Y_h}^I(\Omega_0^I, \rho_0^I)$, $\tilde{\lambda}_{1,Y_h}^I \equiv \tilde{\lambda}_{1,Y_h}^I(\Omega_0^I, \rho_0^I)$ and $\tilde{\lambda}_{2,Y_h}^I \equiv \tilde{\lambda}_{2,Y_h}^I(\Omega_0^I, \rho_0^I)$ are the interval spectral moments [19] of zero-, first- and second-order, respectively, of the normalized random process $\tilde{Y}_h^I(t) = Y_h^I(t) / \sigma_{\tilde{U}_g}^I$. Indeed, introducing the interval function $\tilde{G}_{Y_h,Y_h}^I(\omega) \equiv G_{Y_h,Y_h}^I(\omega; \Omega_0^I, \rho_0^I, (\sigma_{\tilde{U}_g}^2)^I) / (\sigma_{\tilde{U}_g}^2)^I$ (see Eq. (17)), the following relationship holds:

$$\tilde{\lambda}_{\ell,Y_h}^I \equiv \tilde{\lambda}_{\ell,Y_h}^I(\Omega_0^I, \rho_0^I) \equiv \int_0^\infty \omega^\ell \tilde{G}_{Y_h,Y_h}^I(\omega; \Omega_0^I, \rho_0^I) d\omega = \int_0^\infty \omega^\ell \tilde{G}_{Y_h,Y_h}^I(\omega) d\omega, \quad \ell = 0, 1, 2. \quad (21)$$

The *LB* and *UB* of the *ICDF* are formally defined as:

$$\begin{aligned} \underline{L}_{Y_{\max,h}^I}(b, T) &\equiv \min_{\Omega_0 \in \Omega_0^I, \rho_0 \in \rho_0^I, \sigma_{\tilde{U}_g}^2 \in (\sigma_{\tilde{U}_g}^2)^I} \left\{ L_{Y_{\max,h}^I}(b, T; \Omega_0, \rho_0, \sigma_{\tilde{U}_g}^2) \right\}; \\ \bar{L}_{Y_{\max,h}^I}(b, T) &\equiv \max_{\Omega_0 \in \Omega_0^I, \rho_0 \in \rho_0^I, \sigma_{\tilde{U}_g}^2 \in (\sigma_{\tilde{U}_g}^2)^I} \left\{ L_{Y_{\max,h}^I}(b, T; \Omega_0, \rho_0, \sigma_{\tilde{U}_g}^2) \right\}. \end{aligned} \quad (22a,b)$$

The previous bounds can be evaluated by performing global optimization for each value of the barrier level b under the constraint that the uncertain parameters range within the pertinent intervals. In the present study, a more efficient approach is proposed which takes advantage of the dependency of the imprecise *PSD* function on three interval parameters only. The key idea is to estimate the bounds of the *ICDF* starting from the knowledge of the *LB* and *UB* of the interval spectral moments of the normalized response process $\tilde{Y}_h^I(t) = Y_h^I(t) / \sigma_{\tilde{U}_g}^I$, which are formally defined as follows:

$$\begin{aligned}\underline{\tilde{\lambda}}_{\ell, Y_h} &\equiv \min_{\Omega_0 \in \Omega_0^I, \rho_0 \in \rho_0^I} \left\{ \tilde{\lambda}_{\ell, Y_h}(\Omega_0^I, \rho_0^I) \right\} \\ \bar{\tilde{\lambda}}_{\ell, Y_h} &\equiv \max_{\Omega_0 \in \Omega_0^I, \rho_0 \in \rho_0^I} \left\{ \tilde{\lambda}_{\ell, Y_h}(\Omega_0^I, \rho_0^I) \right\},\end{aligned}\quad \ell = 0, 1, 2. \quad (23a,b)$$

Notice that only two optimization parameters appear in Eqs. (23a,b) i.e. $\rho_0 \in \rho_0^I$ and $\Omega_0 \in \Omega_0^I$. The interval spectral moments $\tilde{\lambda}_{\ell, Y_h}^I$ ($\ell = 0, 1, 2$), in general, are not monotonic functions of the uncertain parameters affecting the *PSD* function of seismic excitation (see e.g., [20]). However, in the framework of the proposed model of the *imprecise PSD* function (see Eq.(7)), the combinations of the values of the parameters $\rho_0 \in \rho_0^I$ and $\Omega_0 \in \Omega_0^I$ which yield the bounds of the interval spectral moments $\tilde{\lambda}_{\ell, Y_h}^I$ ($\ell = 0, 1, 2$) can be readily predicted relying on the dynamic behaviour of the seismically excited structure. In particular, attention should be focused on the possibility that resonance with some vibration modes of the structure occurs as the predominant frequency of the excitation $\Omega_0 \in \Omega_0^I$ ranges over the pertinent interval.

5 NUMERICAL APPLICATION

The presented procedure is applied to a single-degree-of-freedom (*SDOF*) system under seismic excitation characterized by the proposed *imprecise PSD* function (see Eq.(7)) with interval parameters defined in Table 2 for moderate and high seismicity areas. The *SDOF* system is characterized by stiffness $k = 23500$ kN/m and modal damping ratio $\xi_0 = 0.05$. In order to show the influence of resonance on the proposed procedure, two different values of the mass are assumed: $m^{(1)} = 52200$ kg (Case 1); $m^{(2)} = m^{(1)} / 2 = 26100$ kg (Case 2).

In Case 1, the natural frequency of the *SDOF* system is $\omega_0^{(1)} = \sqrt{k / m^{(1)}} = 21.217$ rad/s and no resonance can occur for values of the fluctuations of the predominant circular frequency $\Omega_0 \in \Omega_0^I$ of the seismic excitation within the assigned range. In Case 2, the natural frequency of the *SDOF* system is $\omega_0^{(2)} = \sqrt{k / m^{(2)}} = 30.006$ rad/s and resonance may occur when the dimensionless fluctuation of the predominant circular frequency Ω_0^I takes the value $\alpha_{\Omega_0}^{(R)} = -0.173$, such that $|\alpha_{\Omega_0}^{(R)}| < \Delta\alpha_{\Omega_0}$ (see Eq.(6a)). Attention is focused on the interval displacement zero-mean stationary Gaussian random process $U^I(t)$.

Figures 2a and 2b display some realizations of the *imprecise PSD* function $\tilde{G}_{UU}^I(\omega)$ of the displacement process $\tilde{U}^I(t) = U^I(t) / \sigma_{\tilde{v}_g}^I$ for Case 1 and Case 2, respectively. For both cases, samples pertaining to all possible combinations of the endpoints of the uncertain parameters Ω_0^I and ρ_0^I (see Table 2) as well as to the nominal values are plotted. In addition, for Case 2 the realizations associated with $\Omega_0 = \omega_0^{(2)}$ (resonance condition) and ρ_0^I equal to its *UB* or *LB* are plotted. By inspection of Fig. 2a, it is observed that, in Case 1, among the considered samples of the *PSD* of the response, those yielding the *UB* and *LB* of the zero-order spectral moment $\tilde{\lambda}_{0,U}^I$ are associated with $\underline{\Omega}_0, \underline{\rho}_0$ and $\bar{\Omega}_0, \bar{\rho}_0$, respectively. In Case 2, Fig. 2b shows

that, the sample corresponding to the resonance condition $\Omega_0 = \omega_0^{(2)}$ and $\underline{\rho}_0$ is expected to yield the *UB* of $\tilde{\lambda}_{0,U}^I$, while the *LB* is associated again with $\bar{\Omega}_0, \underline{\rho}_0$.

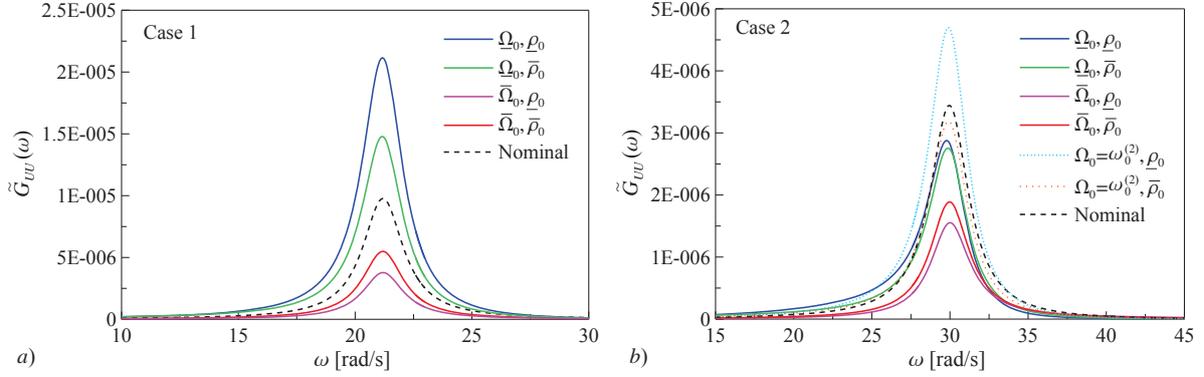


Figure 2: Realizations of the *imprecise PSD* function of the displacement process $\tilde{U}^I(t) = U^I(t) / \sigma_{U_g}^I$: a) Case 1 and b) Case 2.

Figures 3 and 4 display the 3D plots of the spectral moments $\tilde{\lambda}_{\ell,U}^I$ ($\ell = 0, 1, 2$) versus the dimensionless fluctuations $\alpha_{\Omega_0} \in \alpha_{\Omega_0}^I = [-\Delta\alpha_{\Omega_0}, \Delta\alpha_{\Omega_0}]$ and $\alpha_{\rho_0} \in \alpha_{\rho_0}^I = [-\Delta\alpha_{\rho_0}, \Delta\alpha_{\rho_0}]$ of the uncertain parameters $\Omega_0 \in \Omega_0^I$ and $\rho_0 \in \rho_0^I$ for Case 1 and Case 2, respectively. As inferred from Fig. 2, the *LB* of $\tilde{\lambda}_{\ell,U}^I$ ($\ell = 0, 1, 2$) is obtained when the uncertain parameters are set equal to the extreme values $\bar{\Omega}_0$ and $\underline{\rho}_0$ for both Case 1 and Case 2. As far as the *UB* is sought, it is observed that in Case 1 it is achieved when the predominant circular frequency, Ω_0^I , and the circular frequency bandwidth, ρ_0^I , are both equal to their *LB*, in agreement with the prediction suggested by Fig. 2a. Conversely, Fig. 4 shows that in Case 2 the spectral moments are not monotonic functions of $\Omega_0 \in \Omega_0^I$ and it can be reasonably assumed that they achieve the *UB* when $\Omega_0 = \omega_0^{(2)}$ and $\rho_0 = \underline{\rho}_0$, as suggested by Fig. 2b.

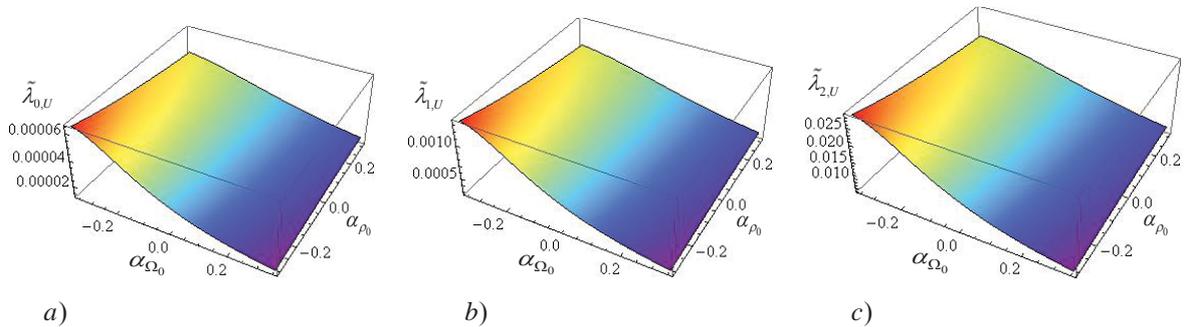


Figure 3: Spectral moments of the displacement process $\tilde{U}^I(t) = U^I(t) / \sigma_{U_g}^I$ versus the dimensionless fluctuations of the uncertain predominant frequency Ω_0 and circular frequency bandwidth ρ_0 : a) zero-order; b) first-order; c) second-order (Case 1).

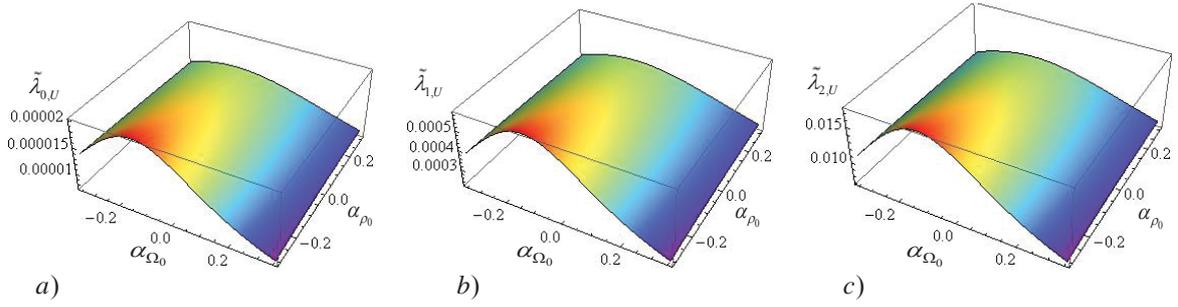


Figure 4: Spectral moments of the displacement process $\tilde{U}^I(t) = U^I(t) / \sigma_{\tilde{U}_g}^I$ versus the dimensionless fluctuations of the uncertain predominant frequency Ω_0 and circular frequency bandwidth ρ_0 : a) zero-order; b) first-order; c) second-order (Case 2).

Figures 5 and 6 show the bounds of the *ICDF*, $L_{U_{\max}}^I(b, T)$, and of the *interval failure probability function*, $P_{f,U_{\max,1}}^I(b, T) = 1 - L_{U_{\max}}^I(b, T)$, along with the nominal solutions pertaining to Case 1 and Case 2, respectively. The observation time is set equal to $T = 30$ s. The proposed bounds are contrasted with the “Exact” ones provided by the scanning method. An excellent agreement is observed. The proposed *LB* and *UB* of the *ICDF* are obtained from Eq. (19) setting the variance of ground motion acceleration $(\sigma_{\tilde{U}_g}^2)^I$ and the spectral moments $\tilde{\lambda}_{\ell,U}^I$ ($\ell = 0, 1, 2$) simultaneously equal to their *UB* and *LB*, respectively. Notice that the width of the intervals of $L_{U_{\max}}^I(b, T)$ and $P_{f,U_{\max,1}}^I(b, T)$ is very large mainly due to the high degree of uncertainty affecting the variance of ground motion acceleration $(\sigma_{\tilde{U}_g}^2)^I$ (see Table 2). In order to obtain results useful for design purposes, tighter bounds of the variance need to be derived by analyzing the selected set of recorded accelerograms. Furthermore, Figures 5 and 6 show that neglecting imprecision of the *PSD* function of seismic excitation may lead to a significant overestimation of the safety level of the structure. Indeed, for a given threshold b , the worst-case scenario identified by the *LB* of the *ICDF* and *UB* of the *interval failure probability* is highly underestimated compared to the nominal values obtained assuming deterministic parameters of the *PSD* function of the excitation.

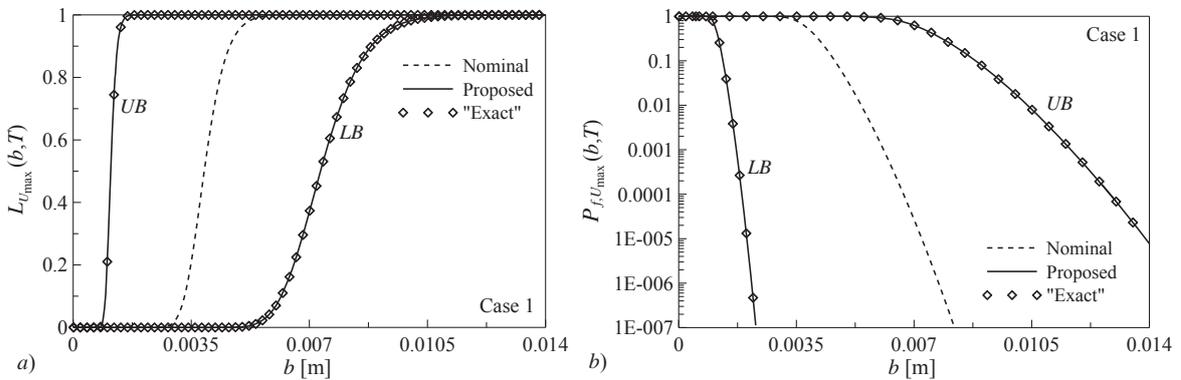


Figure 5: a) *ICDF* and b) *interval failure probability* (in semi-logarithmic scale) of the *extreme value* displacement process $U_{\max}^I(T)$: bounds obtained by applying the proposed approach and the scanning method; nominal solution (Case 1).

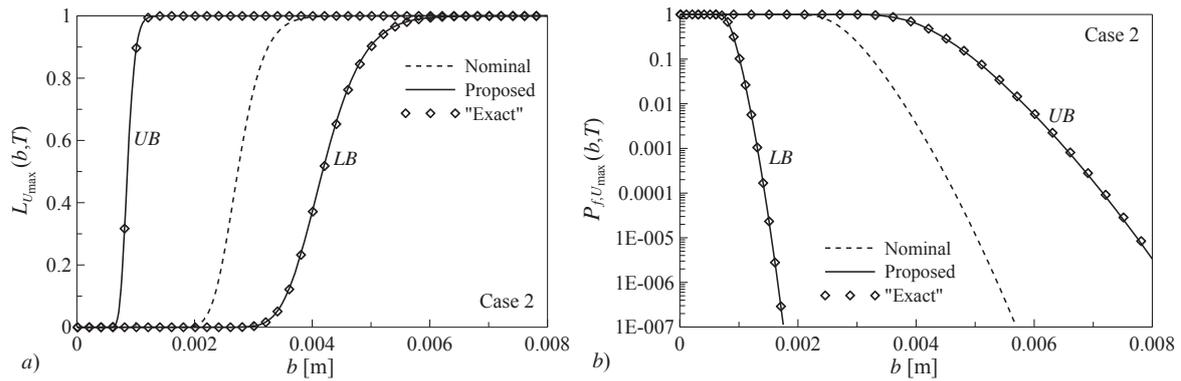


Figure 6: a) ICDF and b) interval failure probability (in semi-logarithmic scale) of the extreme value displacement process $U_{\max}^I(T)$: bounds obtained by applying the proposed approach and the scanning method; nominal solution (Case 2).

6 CONCLUSIONS

- Reliability analysis of linear discretized structures under seismic excitation has been addressed in the framework of the *first-passage* theory taking into account uncertainties affecting the definition of the input described by a zero-mean Gaussian stationary random process.
- By analysing a set of accelerograms recorded on rigid soil deposits it has been recognized that the *power spectral density (PSD)* function of ground motion acceleration has an imprecise nature. Indeed, the three main parameters characterizing the *PSD* function i.e. the predominant circular frequency, the circular frequency bandwidth and the variance of the process, assume very different values from one accelerogram to another. Moreover, the analysis has shown that the *PSD* function representative of accelerograms recorded in soils with specific geotechnical characteristics is more appropriately described by an interval function.
- This entails that the well-established seismic spectra, with deterministic parameters proposed in literature, provide only indicative models of recorded accelerograms in seismic areas which may differ from the actual ones.
- Numerical results have demonstrated the remarkable influence of imprecision of earthquake excitation on structural performance. In particular, it has been shown that neglecting uncertainties affecting the main parameters of earthquake spectrum may lead to significant overestimation of the safety level.

REFERENCES

- [1] R. Clough, J. Penzien, *Dynamics of structures*, 2nd Edition. McGraw-Hill, 1993.
- [2] F. Genovese, D. Aliberti, G. Biondi, E. Cascone, Geotechnical aspects affecting the selection of input motion for seismic site response analysis, 151-161, M. Papadrakakis and M. Fragiadakis eds. 7th International Conference COMPDYN 2019, Crete, Greece, 24-26 June, 2019.
- [3] P. Cacciola, P. Colajanni, G. Muscolino, Combination of modal responses consistent with seismic input representation, *Journal of Structural Engineering (ASCE)*, **130**, 47–55, 2004.

- [4] G. Barone, F. Lo Iacono, G. Navarra, A. Palmeri, Closed-form stochastic response of linear building structures to spectrum consistent seismic excitations, *Soil Dynamics and Earthquake Engineering*, **142**, e102749, 2019.
- [5] G. Muscolino, F. Genovese, G. Biondi, E. Cascone, Generation of fully non-stationary random processes consistent with target seismic accelerograms, *Soil Dynamics and Earthquake Engineering*, **141**, 106467, 2021.
- [6] I. Elishakoff, Possible limitations of probabilistic methods in engineering, *Applied Mechanics Reviews*, **53**, 19–36, 2000.
- [7] I. Elishakoff, M. Ohsaki, *Optimization and Anti-Optimization of Structures under Uncertainty*. Imperial College Press, London, 2010.
- [8] EC8 (European Committee for Standardization, Eurocode 8). Design of structures for earthquakes resistance-Part 1: General rules, seismic actions and rules for buildings, (EN 1998-1).
- [9] G. Barbato, E.M. Barini, G. Genta, R. Levi, Features and performance of some outlier detection methods, *Journal of Applied Statistics*, **38**, 2133–2149, 2011.
- [10] M.P. Maples, D.E. Reichart, N.C. Konz, T.A. Berger, A.S. Trotter, J.R. Martin, D.A. Dutton, M.L. Paggen, R.E. Joyner, C.P. Salem, Robust Chauvenet Outlier Rejection. *The Astrophysical Journal Supplement Series*, **238**, 2 (49pp), 2018.
- [11] R.E. Moore, *Interval Analysis*, Prentice-Hall, Englewood Cliffs, 1966.
- [12] R.E. Moore, R.B. Kearfott, M.J. Cloud, *Introduction to Interval Analysis*, SIAM, Philadelphia, 2009.
- [13] E.H. Vanmarcke, On the distribution of the first-passage time for normal stationary random processes, *Journal of Applied Mechanics (ASME)*, **42**, 215–220, 1975.
- [14] S. Ferson, V. Kreinovich, L. Ginzburg, D.S. Myers, K. Sentz, *Constructing probability boxes and Dempster-Shafer structures*, Sandia National Laboratories SAND2002–4015, 2003.
- [15] L. Luzi, G. Lanzano, C. Felicetta, M.C. D’Amico, E. Russo, S. Sgobba, F. Pacor, ORFEUS Working Group 5, Engineering Strong Motion Database (ESM) (Version 2.0), Istituto Nazionale di Geofisica e Vulcanologia (INGV) 2020.
- [16] L.D. Lutes, S. Sarkani, *Random vibrations - analysis of structural and mechanical vibrations*, Elsevier Inc, Boston, 2004.
- [17] J.P. Conte, B.F. Peng, Fully nonstationary analytical earthquake ground-motion model, *Journal of Engineering Mechanics (ASCE)*, **123**, 15–24, 1997.
- [18] G. Muscolino, A. Sofi, Stochastic analysis of structures with uncertain-but-bounded parameters via improved interval analysis, *Probabilistic Engineering Mechanics*, **28**, 152–163, 2012.
- [19] E.H. Vanmarcke, Properties of spectral moments with applications to random vibration, *Journal of Engineering Mechanics (ASCE)*, **98**, 425–446, 1972.
- [20] M.G.R. Faes, M. A. Valdebenito, D. Moens, M. Beer, Bounding the first excursion probability of linear structures subjected to imprecise stochastic loading. *Computers and Structures*, **239**, 106320, 2020.

STRUCTURAL RELIABILITY ESTIMATION OF STEEL MAST EXHIBITING RANDOM MECHANICAL AND ENVIRONMENTAL PARAMETERS

R. Bredow¹, M.Kamiński²

¹ PhD Candidate, Department of Structural Mechanics, Łódź University of Technology, POLAND
al. Politechniki 6, Łódź 90-924
e-mail: rafal.bredow@dokt.p.lodz.pl

² Professor, Department of Structural Mechanics, Łódź University of Technology, POLAND
al. Politechniki 6, Łódź 90-924
e-mail: marcin.kaminski@p.lodz.pl

Keywords: generalized stochastic perturbation technique, Stochastic Finite Element Method, reliability analysis, structural mechanics.

Abstract. *The aim of this work is to study an influence of environmental and mechanical uncertainties on reliability assessment of some steel mast subjected to the given dynamic wind spectrum. Some exemplary steel guyed mast structure has been tested including geometrical non-linearities inherent in its dynamic response history spectra for the ultimate and serviceability limit states. Numerical solution of dynamic excitation problem has been obtained using the Finite Element Method system ROBOT. Further research, which has been performed in computer algebra system MAPLE 2019, included sensitivity study regarding an order of polynomial approximation of structural response functions and also the resulting structural probabilistic characteristics presented as the functions of the input uncertainties. The Weighted Least Squares Method with triangular weight functions has been applied to recover some structural response polynomials. Probabilistic analysis has been performed with the use of the Iterative Stochastic Perturbation Technique, where accuracy of this method has been compared with the Monte-Carlo simulation and also with the semi-analytical approach. A coincidence in this comparison for the first two probabilistic moments of structural response has been discussed in this paper accordingly.*

1 INTRODUCTION

This work contains a study of the influence of environmental and mechanical uncertainties on reliability index of some guyed steel mast subjected to the given dynamic wind spectrum. This excitation has been defined using relatively short period spectrum following some experimental measurements. A structure of this mast has a height equal to 198.0 m and its shaft has been designed with the use of S235J2 steel in form of three-walled lattice with side width equal to 1.30 m. Leg members have been modelled as round pipes with diameter of 168.3 mm with the cross-section wall thickness adjusted to the ultimate limit state conditions. The mast face lacing elements have been designed as round pipes of the diameter 63.5 mm and their wall thickness has been designed quite similarly. Mast guys have been attached to the shaft at the altitudes equal 60.0 m, 120.0 m and 180.0 m from the ground level and they are introduced with the inclination angle equal to 45 degrees. A spiral single strand steel rope 1x37 with the diameter of 32.0 mm has been applied with the mean strength equal to 1960.0 MPa. The mean elasticity modulus of the cables has been assumed as 150.0 GPa, whereas elasticity modulus of the mast shaft elements has been adopted as 210.0 GPa. An initial tension for the mast guys has been provided by prestressing equivalent to 11.0 cm, 22.0 cm and 31.0 cm, correspondingly for the consecutive attachment levels with ascending order starting from the bottom.

Numerical model prepared in the Finite Element Method system ROBOT consists of 903 finite elements. Mast shaft has been modelled by 2 node bar elements described by linear shape functions and 6 degrees of freedom at each end – mast structure consists of 894 of such elements. Mast guys has been modelled as cable finite elements implemented in this system according to the small-sag theory. This furtherly contributes to assumptions such as the equilibrium of the cable is being found considering constant tensile force of the cable along its length.

2 UNCERTAINTY ANALYSIS

Several environmental and mechanical uncertainties have been taken into account for computer analysis of the dynamic response spectra in the most fragile elements of the mast. They have been defined as Gaussian variables with the given expectations and some interval of coefficient of variation. Both types of uncertainties and their implementation methods into numerical analysis have been discussed here as the functions of the approximating polynomial order and the input uncertainty level. The final graphs attached below include only these parameters, which appeared to be decisive for the mast reliability analysis.

2.1 Environmental uncertainties

The first environmental uncertainty is the external temperature applied to the mast structure. Two ranges of temperature load applied uniformly to the entire structure have been taken into consideration, namely from -10°C to +40°C, and also from -50°C until ±0°C. These two case studies of external temperature have been additionally discretized into 11 sub-cases with an increase equal to 5°C. The second environmental uncertainty has been described by the uncertain wind velocity distribution. This wind load has been modelled according to the Eurocode 1 guidelines for towers, chimneys and masts including an effect of the local wind

gusts [1]. A dynamic analysis of the wind influence on this structure has been performed for a time interval of 10 minutes according to the spectrum included in Fig. 1.

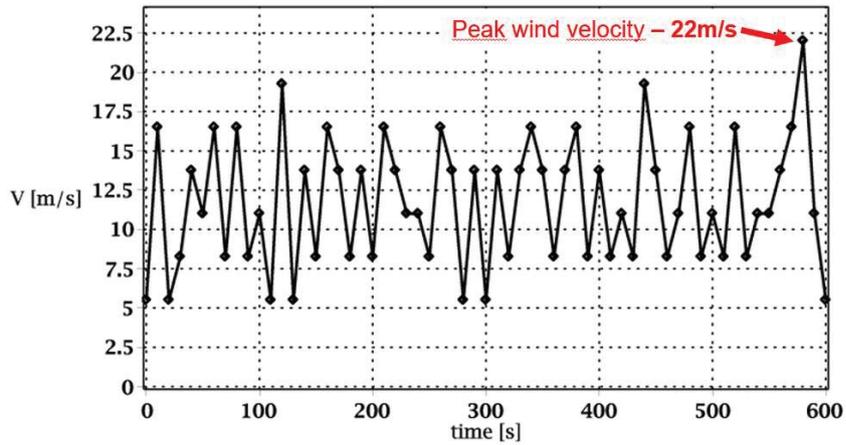
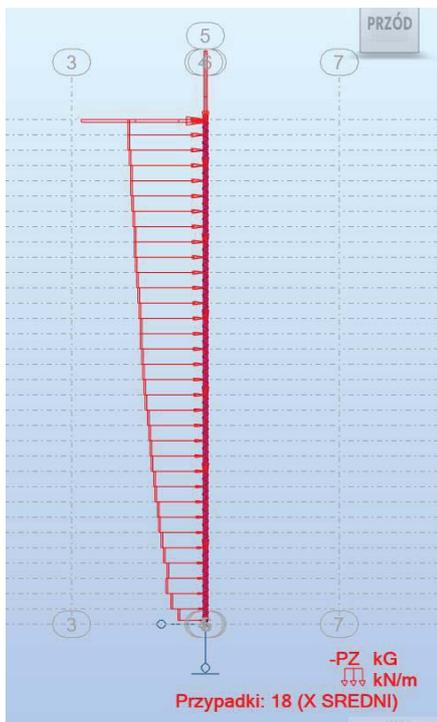


Figure 1: Wind velocity fluctuations in 10-minute interval

The uncertainty description of dynamic wind action upon this structure has been performed similarly to the temperature load case. It means that 11 types of dynamic wind load histories have been analyzed, each of them described by a different multiplication factor applied to the spectrum presented in Fig.1. Values of peak wind velocity taken into account are specifically presented and discussed further in section 3.2. Wind action expressed as pressure applied to the mast consists of some mean load and supplementary patch load simulating additional wind gusts along certain parts of the mast according to [1]. An example of the wind load distribution applied to the mast shaft, which consists of both load types, has been presented in Fig. 2.

a)



b)

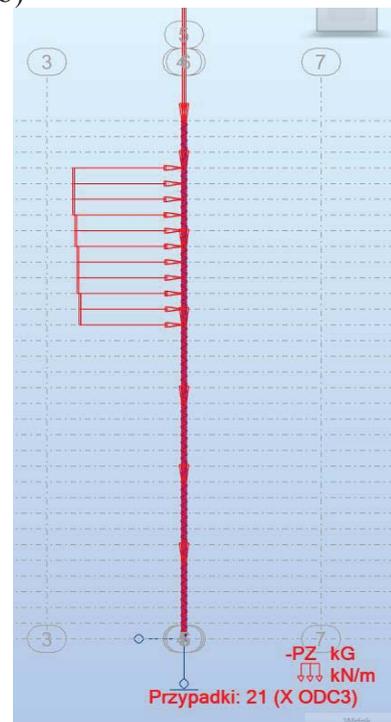


Figure 2: Wind load distribution along the mast height: a) mean load, b) patch load

2.2 Mechanical uncertainties

Mechanical uncertainties considered in this analysis are - the elasticity modulus of mast guys and, independently, elasticity modulus of mast shaft. In both cases 11 representative values of elasticity modulus have been taken into account. These representative test values of elasticity moduli have been calculated based on expected value, which refers directly to Eurocodes namely 210.0 GPa for mast shaft elements and 150.0 GPa for mast guys respectively. Multiplication factors for these representative values have been introduced as 0.90, 0.92, 0.94, 0.96, 0.98, 1.00, 1.02, 1.04, 1.06, 1.08, 1.10, consecutively.

2.3 State variables

State variables for the steel guyed mast structure have been chosen according to the Eurocode guidelines [2], [3], which indicates that verification of the Ultimate Limit States (ULS) and Serviceability Limit States (SLS) in probabilistic context is sufficient for its reliability. Analyzed state variables of the greatest interest represent both ULS and SLS states and they refer to the stress-state of main leg element, stress-state of face lacing element, global horizontal displacement of the top of mast shaft and also extreme twisting of the shaft. First two obviously have been taken upon with respect to ULS and the remaining two – for the SLS.

3 NUMERICAL SOLUTION

3.1 Dynamic response spectra

Numerical solution has been obtained via simulations performed in the system ROBOT using non-linear dynamic analysis procedure based on the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm. Integration of equations of motion induced by the wind spectrum has been performed with the use of Hilber-Hughes-Taylor (HHT) solver [4]. An accuracy of the HHT method for geometrically non-linear guyed mast structure has been studied by a contrast with the Newmark solver [5], which has been presented in Figure 3.

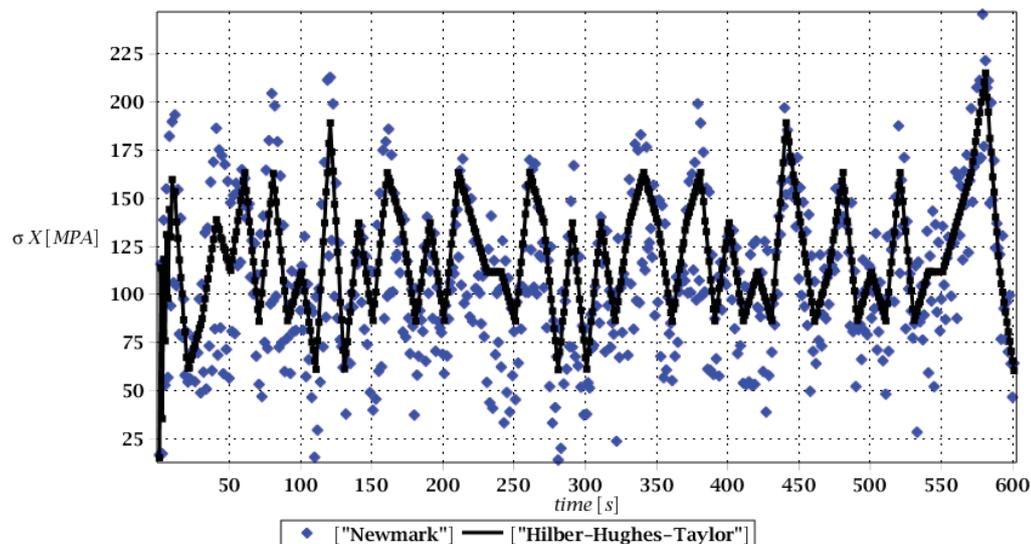


Figure 3: Response history for the stress in main leg computed using the HHT and the Newmark methods

The HHT solver is based upon specific time discretization, where the structural displacements and velocities in the given time step are described using displacements, velocities and accelerations in the previous time step as

$$\begin{cases} x_{i+1} = x_i + \Delta t \cdot \dot{x}_i + (1/2 - \beta) \cdot (\Delta t)^2 \cdot \ddot{x}_i \\ \dot{x}_{i+1} = \dot{x}_i + (1 - \gamma) \cdot \Delta t \cdot \ddot{x}_i \end{cases} \quad (1)$$

Time step of the method has been set here as 0.10 s and the output response results have been saved every tenth time steps (at every 1.0 s). The most dynamic response has been recalculated for several realizations of each random parameter about its expected value, so that 176 series of the Stochastic Finite Element Method computations have been performed resulting in 105,600 discrete results of the principal state variables.

3.2 Structural Response Function approximation

The Weighted Least Squares Method (WLSM) has been selected in order to perform the Structural Response Function (SRF) approximation in the form of polynomials of order varying from the 5th until the 10th one [6]. This has been completed for each state variable in the time step corresponding to the extreme values in the limit states. Let us note that the uniform, triangular and Dirac weight functions distributions have been initially considered for this approximation procedure. Triangular weight functions exhibited the best accuracy for the provided series of dynamic response data with respect to least square error minimization and also overfitting phenomenon minimization criterium, so that they have been applied in further SFEM analysis. The prescribed weights for each uncertain parameter have been given in such a way that the greatest weight corresponds to the expected value. Referring to Table 1, one can notice that weights for WLSM approximation of the temperature load case should be (1,2,3,4,5,6,7,8,9,8,7), while for the peak wind are proposed as (1,2,3,4,5,6,5,4,3,2,1). The discrete values of each uncertain parameters have been presented in Table 1.

Input uncertain parameter					
No.	Temperature load 1 °C	Temperature load 2 °C	Peak wind velocity m s ⁻¹	Elasticity modulus – guys GPa	Elasticity modulus – shaft GPa
1	-10.00	-50.00	19.80	135.0	189.0
2	-5.00	-45.00	20.24	138.0	193.2
3	0.00	<u>-40.00</u>	20.68	141.0	197.4
4	5.00	-35.00	21.12	144.0	201.6
5	10.00	-30.00	21.56	147.0	205.8
6	15.00	-25.00	<u>22.00</u>	<u>150.0</u>	<u>210.0</u>
7	20.00	-20.00	22.44	153.0	214.2
8	25.00	-15.00	22.88	156.0	218.4
9	<u>30.00</u>	-10.00	23.32	159.0	222.6
10	35.00	-5.00	23.76	162.0	226.8
11	40.00	0.00	24.20	165.0	231.0

Table 1: Discretization of the two uncertain parameters (expected values underlined)

Each data of the response history has been approximated by 5th to 10th order polynomials. Finally, 4 uncertain parameters, 4 state variables and 600 time steps have been analyzed and the WLSM approximation by 6 different polynomials has been performed. So that general database consists of 57,600 polynomials obtained solely by finding the solution with the HHT solver. For a brevity of this work presentation, the SRF representing extreme value of some state variables have been taken into account only. It has been assumed that this moment in dynamic analysis would be considered as the most dangerous from structural engineering perspective and its SRF should be serve in further mast reliability index estimation.

4 PROBABILISTIC ANALYSIS

Probabilistic analysis has been performed with the Stochastic Perturbation Technique (SPT), and independently using Monte-Carlo simulations (MCS) and semi-analytical method (SAM) also [7], [8]. The results obtained by SPT are marked with asterisk, these coming from the MCS - with a cross, while these obtained by SAM - with a diagonal box. The main study has been performed to verify how the order of approximation would affect the resulting basic probabilistic characteristics and, finally, reliability index determination. A coincidence in-between three different probabilistic methods has been also investigated. Each SRF has been independently approximated as a function of some uncertain parameter. To pursue the abovementioned investigation it has been assumed that the SRF are some functions of a random variable with unknown standard deviation. Then, the SRFs have been presented as a result related directly to the coefficient of variation of this particular variable.

Expected values of normal stresses in the mast leg are presented in Fig. 4 below as the functions of the input coefficient of variation of Young modulus of the guys and also of the response polynomial order. As it is demonstrated, an increase of the input uncertainty leads each time to a decrease of this expectation. Further, one may notice that the resulting extreme expectation seems to be quite sensitive to the chosen approximation order, especially for larger values of the coefficient α . Higher orders of this approximation make these variations more remarkable. The resulting coefficients of variation of these stresses in addition to a wind pressure coefficient of variation have been presented further in Fig. 5. An interrelation in-between these coefficients is almost linear and the impact of the polynomial approximation order is definitely smaller than in Fig. 4. On the other hand, the resulting coefficients of variation of these stresses with respect to Young modulus of the mast guys shown in Fig. 6 presented more nonlinear interrelation to coefficient α . Increase of parameter α in such case results in non-linear increase of output coefficient of variation.

The expected values of the normal stresses in face lacing elements are contained in Fig 7 as the functions of the input coefficient of variation of Young modulus of the guys and also of the response polynomial order. Fifth and sixth order polynomials in this case presented opposite monotonicity to higher order polynomials which can be observed for values of parameter α greater than 0.10. Once again one may notice that obtained expectations are sensitive to polynomial order for larger values of abovementioned parameter α . The resulting coefficient of variation of this stresses has been presented in Fig. 8. In this case an interrelation in-between input and output variations is exponentially related. Some additional inaccuracy in between SPT and SAM has been observed for tenth order polynomial approximation. It is seen that this randomness is the largest one, which confirms a fundamental role of the Ultimate Limit State for this structure safety.

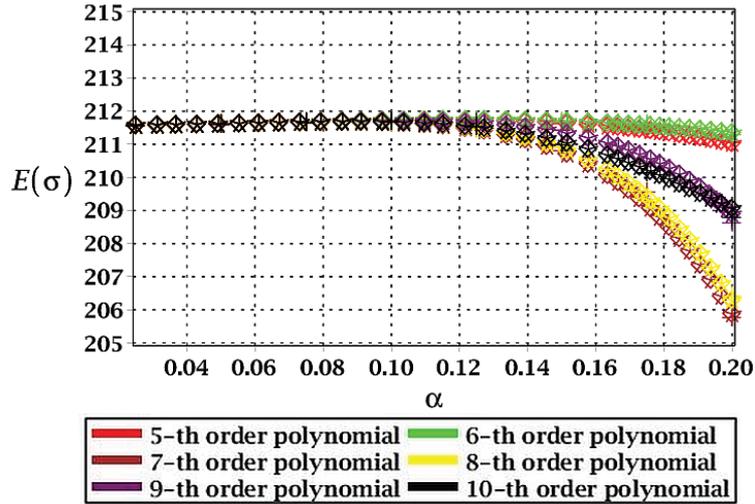


Figure 4: Expected value of response function describing stress in main leg in reference to guys elasticity modulus coefficient of variation

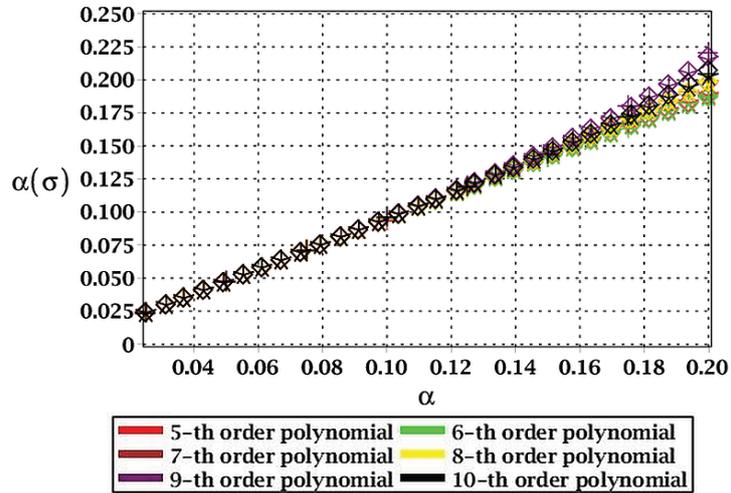


Figure 5: COV of response function describing stress in main leg in reference to wind load coefficient of variation

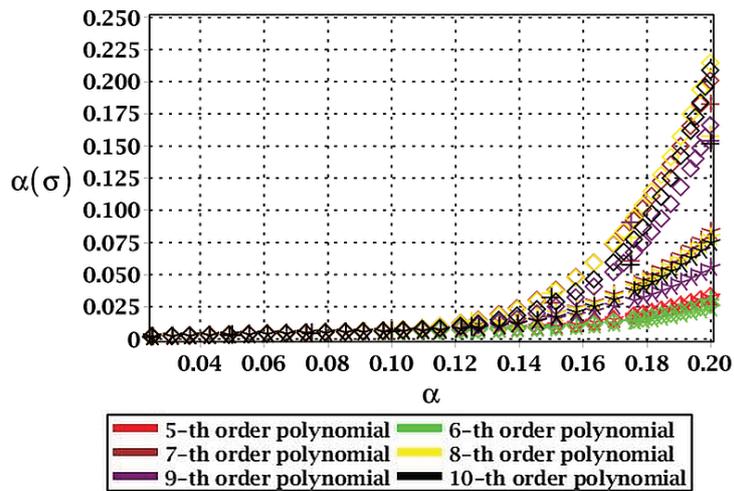


Figure 6: COV of response function describing stress in main leg in reference to guys elasticity modulus coefficient of variation

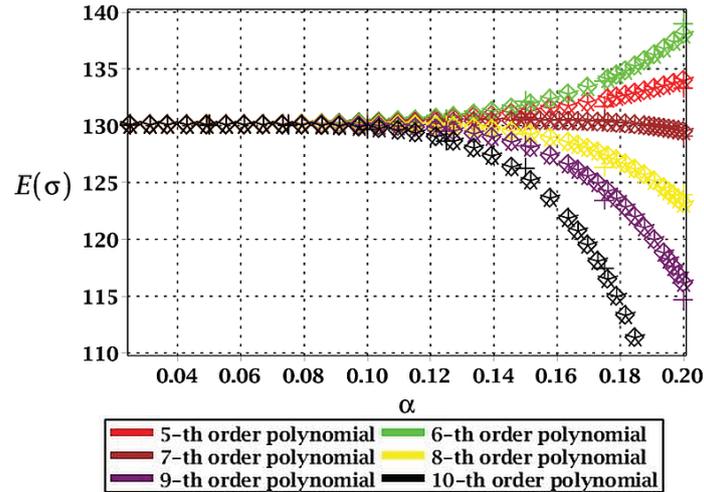


Figure 7: Expected value of response function describing stress in face lacing in reference to guys elasticity modulus coefficient of variation

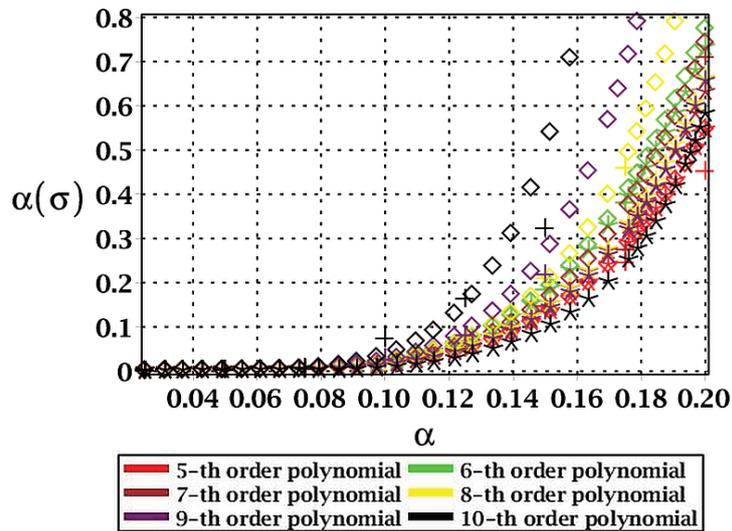


Figure 8: COV of response function describing stress in face lacing in reference to guys elasticity modulus coefficient of variation

Expected values of global horizontal displacement are shown in Fig. 9 below as the functions of the input coefficient of variation of Young modulus of the guys and also of the response polynomial order. As it was demonstrated, an increase of the input uncertainty leads each time to a increase of this expectation except for tenth order polynomial approximation. Further, one may notice that the resulting extreme expectation seems to be quite sensitive to the chosen approximation order, especially for larger values of the coefficient α . The resulting coefficients of variation of these stresses related to guys Young modulus coefficient of variation have been presented further in Fig. 10. An interrelation in-between these coefficients is exponentially monotonous. The best approximations of these coefficients of variations in-between all three probabilistic techniques has been observed for fifth, sixth and seventh order of polynomial approximation.

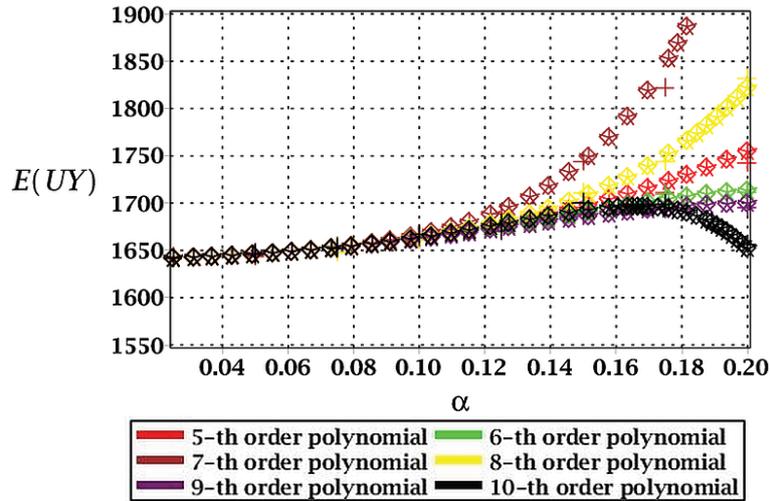


Figure 9: Expected value of response function in reference to guys elasticity modulus coefficient of variation

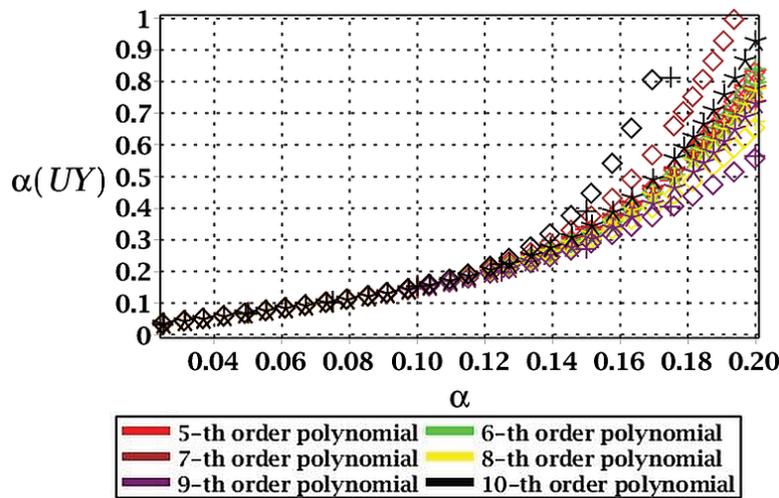


Figure 10: COV of response function in reference to guys elasticity modulus coefficient of variation

Expectations of a twist of the mast shaft are collected in Fig. 11 below as the functions of the input coefficient of variation of Young modulus of mast shaft and also of the response polynomial order. As it is demonstrated, an increase of the input uncertainty leads to a decrease of this expectation except for fifth order polynomial approximation where minor increase is observed. Further, one may notice that the resulting extreme expectation seems to be quite sensitive to the chosen approximation order, especially for larger values of the coefficient α . The resulting coefficients of variation of these stresses related to mast shaft Young modulus coefficient of variation have been presented further in Fig. 12. An interrelation in-between these coefficients is nonlinearly monotonous. The best approximations of these coefficients of variations in-between all three probabilistic techniques has been observed for fifth, sixth, seventh and eighth order of polynomial approximation.

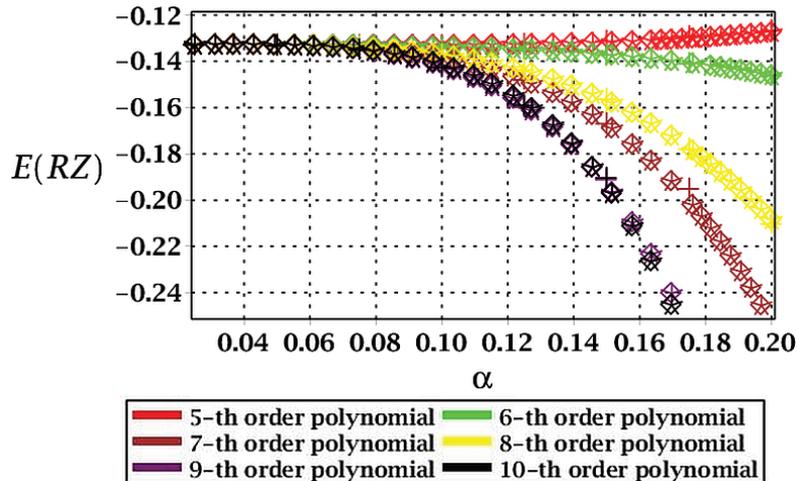


Figure 11: Expected value of response function in reference to shaft elasticity modulus coefficient of variation

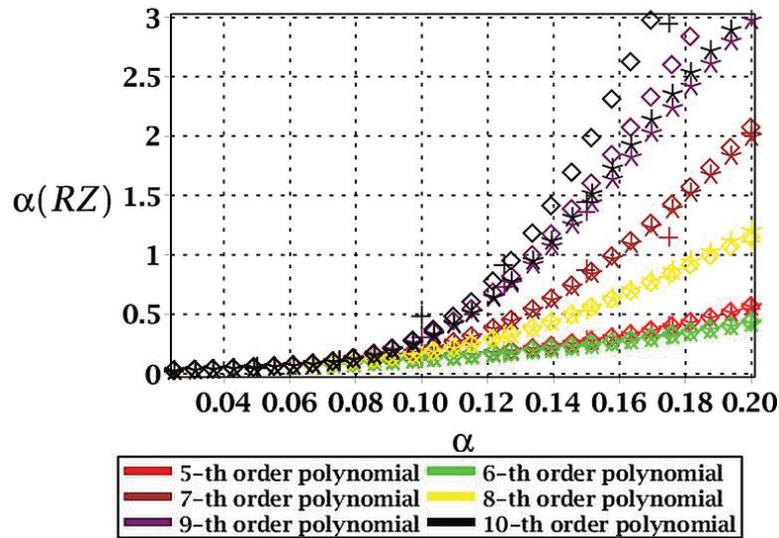


Figure 12: COV of response function in reference to shaft elasticity modulus coefficient of variation

5 RELIABILITY ASSESSMENT

Eurocode 0 statements regarding reliability of structures it should be noted that structures such as guyed steel masts undoubtedly belong to the highest reliability class RC3 thus their reliability assessment should be available at hand. The Authors propose in this paper to express reliability of the mast structure by the Cornell theory exposed in Eurocode 0. According to [2] the minimum reliability index value regarding ULS for RC3 class is equal to 4.3. For SLS the Eurocode 0 does not provide target reliability values with respect to the reversible SLS states and their arbitrary assumption must comply with the expectations of the investor as well as allow safe exploitation of the equipment attached to the structure.

The reliability index has been calculated as a function of coefficient of variation of the given input parameter and results of such investigation has been presented for most significant random parameters with respect to considered state variable under assessment. Significance of chosen random parameters once again has been dictated by the sensitivity of the mast structure

which has been visualized in probabilistic response spectra presented in chapter 4. From Figs. 13-16 can be deduced that all proposed polynomial approximations produces similar outcome in form of reliability responses related to input parameter α . From all uncertainties taken into account it can be noticed that the mast structure exhibits the greatest sensitivity towards uncertain Young modulus of guys and towards wind pressure. This observation can be expressed directly through determination and presentation of the reliability index as for Figs.14-15 reliability drops below some arbitrary level of acceptance at relatively small values of input α parameter. When acceptance level is set as reliability index equal to 2.0 for example, then the mean wind pressure and Young modulus should be described using random distributions with a coefficient of variation smaller than 0.08.

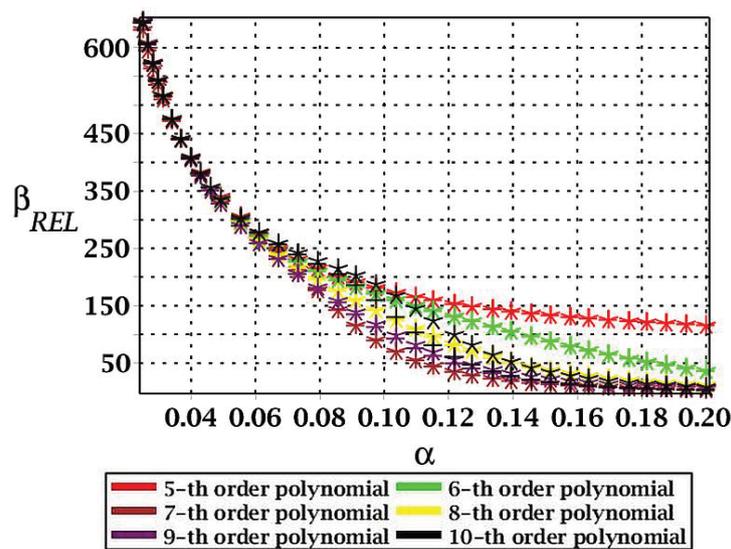


Figure 13: Reliability index in function of input COV of shaft elasticity modulus with respect to ULS state of stress in face lacing elements

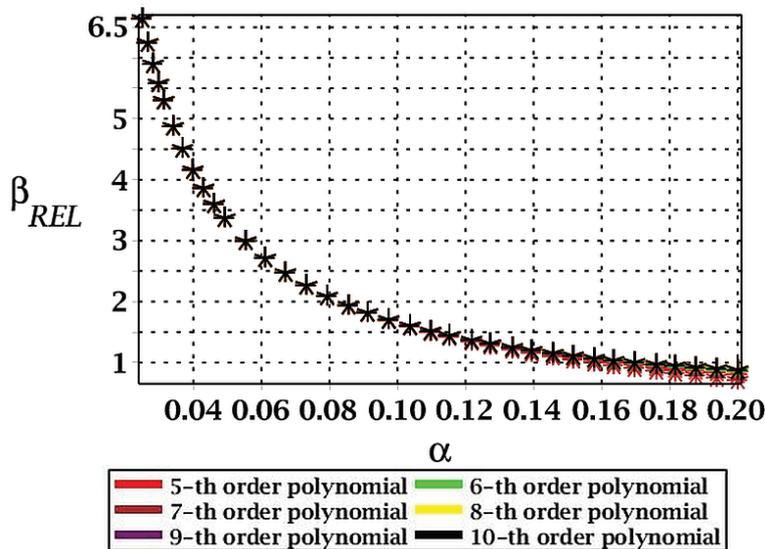


Figure 14: Reliability index in function of input COV of wind velocity with respect to SLS state of global horizontal displacement

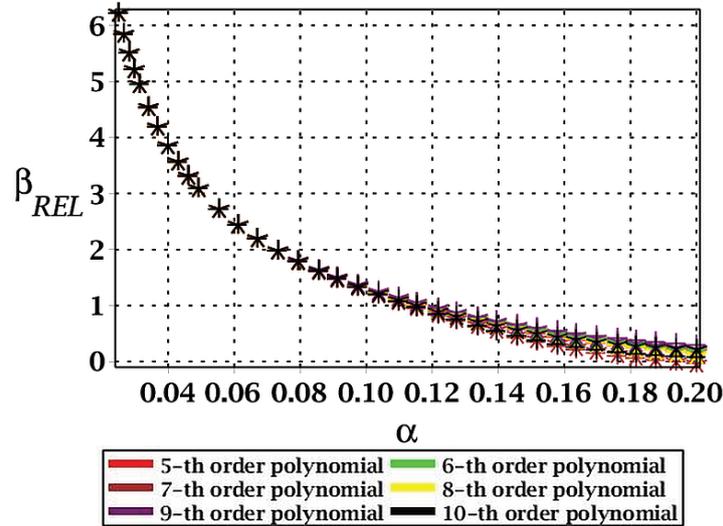


Figure 15: Reliability index in function of input COV of guys elasticity modulus with respect to SLS state of global horizontal displacement

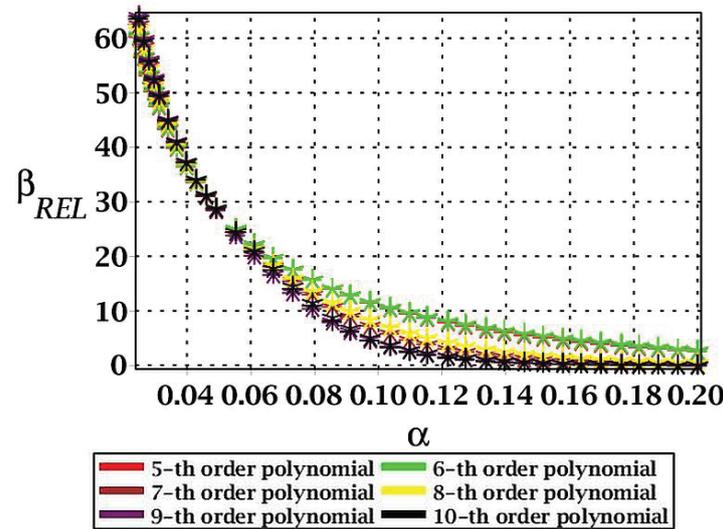


Figure 16: Reliability index in function of input COV of shaft elasticity modulus with respect to SLS state of twist of mast shaft

6 CONCLUDING REMARKS

The Iterative Stochastic Perturbation Technique shows quite satisfactory coincidence in comparison with both Monte-Carlo simulation estimators and also with the semi-analytical method moments and it requires also remarkably less computational time and effort. This is very important considering the fact that nonlinear geometrical effects are considered during dynamic excitation of the structure, which undoubtedly is large scale civil engineering structure. More interestingly, higher order polynomials have returned neither more stable nor predictable random responses. Different orders of polynomials used in WLSM show exquisite accuracy when the input coefficient of variation is smaller than about 0.10 – 0.12. When the input COV is above this value, the differences in-between various orders of polynomials

become more significant and even some of them show opposite monotonicity than the others. Some differences in between the SPT and referential techniques are observed above this uncertainty level. Nevertheless, material uncertainties or these associated with the dead loads are well known and undergo smaller deviations than this limit in civil engineering. In such case it has been proven that an order of the polynomial used in reliability index assessment can be arbitrarily assumed and may result in quite satisfactory outcome. On the other hand, when uncertainties taken into account exhibit larger deviations such as wind velocity history or some other environmental phenomenon, the technique presented in this paper might be used to firstly verify and choose proper other of polynomial or other function in such manner that it would perform satisfactory coincidence in comparison with referential techniques. Such a function then might be used in non-stationary reliability assessment of the structure exhibiting some nonlinearities and subjected at the same time to the dynamic excitation.

One can notice that the given guyed mast structure exhibits different sensitivities with respect to the chosen uncertainty sources. What is more, the structural sensitivity can be associated with the certain state variable under consideration. In example let us imagine considering the twist of mast shaft – one can assume that the general structural stiffness of the mast towards twist could be directly associated with the stiffness of mast shaft itself. Obtained results confirms that the uncertainty in mast shaft elasticity modulus contributes in the greatest manner to its dynamic response to the wind excitation. The output COV of structure response function correlated with shaft elasticity modulus increases most significantly with increasing parameter α of elasticity modulus of mast shaft. Similar observations can be made with respect to the other state variables. From this analysis it can be furtherly concluded that for both ULS states (stress analysis in both main legs and the face lacing elements) the wind velocity is the most significant random parameter and for the SLS state expressed by horizontal displacement the mast structure exhibits the greatest sensitivity to elasticity of mast guys. What is more the temperature load is the least significant uncertain parameter considered in this paper with respect to all four state variables taken into account.

Sensitivity analysis and reliability index estimation are strictly related with each other as it may be observed from in this paper. Reliability index reaches the lowest values for the uncertain parameter to which the structure is the most sensitive – regarding limit state under consideration. In summary for both ULS associated with stress the structure presents the greatest sensitivity towards uncertainty expressed in wind velocity. Then for SLS associated with horizontal displacement the structure exhibits significant sensitivity with respect to both wind velocity and elasticity of mast guys and finally for SLS associated with structural rotation, the structure exhibits the greatest sensitivity towards elasticity modulus of mast shaft.

From another perspective it can be also noticed that various orders of approximating polynomials usually lead to similar results of probabilistic computer analysis and also, in consequence, to similar reliability index values.

REFERENCES

- [1] *Eurocode 1: Actions on structures – Part 1-4: General actions – Wind actions* – European Committee for Standardization, Brussels, 2010
- [2] *Eurocode 0: Basis of structural design* – European Committee for Standardization, Brussels, 2002
- [3] *Eurocode 3: Design of steel structures – Part 3-1: Towers, masts and chimneys – Towers*

- and masts* – European Committee for Standardization, Brussels, 2006
- [4] Hilber, H.M., Hughes, T.J.R. & Taylor, R.L., *Improved Numerical Dissipation for Time Integration Algorithms in Structural Dynamics*, *Earthquake Engineering & Structural Dynamics*, 1997, 5, 282-292
- [5] Newmark, N.M., *A method of computation for structural dynamics*, *Journal of the Engineering Mechanics Division*, 1959, 85 (EM3): 67-94
- [6] Kamiński, M., Bredow, R., *Uncertainty Analysis for Overhead Powerlines by the Generalized Stochastic Perturbation Technique*, *Journal of Aerospace Engineering*, DOI: 10.1061/(ASCE)AS.1943-5525.0001270, 2021, (in press).
- [7] Kamiński M., *The Stochastic Perturbation Method for Computational Mechanics* – Chichester, Wiley.
- [8] Bredow. R., Kamiński, M., *Computer analysis of dynamic reliability of some concrete beam structure exhibiting random damping*, *Int. J. of Applied Mechanics and Engineering*, 2021, vol. 26, No.1, pp.45-64

INVERSE PROBLEMS FOR STOCHASTIC NEUTRONICS

Corentin Houpert¹, Josselin Garnier², Philippe Humbert³

¹CEA, DAM, DIF and IP Paris, École polytechnique
F-91297 Arpajon
e-mail: corentin.houpert@cea.fr

² IP paris, École polytechnique
Route de Saclay, 91120 Palaiseau
e-mail: josselin.garnier@polytechnique.edu

³CEA, DAM, DIF
F-91297 Arpajon
e-mail: philippe.humbert@cea.fr

Keywords: Uncertainty Quantification, Bayesian inverse problems, Adaptive Metropolis, Covariance matrix adaptation, Stochastic neutronics, Neutron point model approximation

Abstract. *Fissile matter detection and characterisation are crucial issues; especially in nuclear safety, safeguards, matter comptability, reactivity measurements. In this context, we want to identify a source of fissile matter knowing external measures such as instants of detection of neutrons during an interval of measure. Thus we observe the neutrons detection times emitted by the fissile matter and going through the detector, then we compute the moments of the empirical distribution of the number of neutrons detected during a time gate T . In order to identify the source we have to get the following parameters: the multiplication factor k of the system, the intensity of the source S , the fission efficiency ε_F .*

Given the parameters of the source there are some models that allow us to predict the moments of counted number of neutrons during a time gate T . We consider a point model stating monokinetic neutrons are moving in an infinite, isotropic and homogeneous medium. The method makes it possible to compute the first moments of the count number distribution.

Then, given the moments of counted number of neutrons during a time gate T we want to get the parameters of the fissile source. In order to achieve this goal, we will use the following method

- *Bayesian approach in order to get the distribution of parameters. The a posteriori distribution is non-trivial, samples can be achieved with Markov Chain Monte-Carlo methods with covariance matrix adaptation (MCMC with CMA).*

Nomenclature

Nuclear constants

α	Decreasing coefficient of the neutronic system
$\bar{\nu}$	Mean number of neutrons emitted by a fission event
$\bar{\nu}_S$	Mean number of neutrons emitted by a source event
λ_C	Capture rate by time unit
λ_F	Fission rate by time unit
D_{2S}	Diven factor of the source of order 2
D_2	Diven factor of the fission of order 2
D_{3S}	Diven factor of the source of order 3
D_3	Diven factor of the fission of order 3
f_ν	Probability the fission emits ν neutrons
$f_{\nu,S}$	Probability that source emits ν neutrons during a source event
p	Probability that a neutron causes a fission

Nuclear parameters

\mathbf{p}	Vector of the parameters of the system
\mathbf{p}^*	Vector of the parameters of the system to estimate
ε_C	Capture efficiency
k	Multiplication factor
S	Intensity of the source (neutron/units of time)

Observations and model outputs

\mathbf{M}	Vector of the first three simple statistical moments, the model
$\hat{\mathbf{M}}$	Vector of the first three simple empirical moments, the measures
T	Time gate (units of time)

1 Introduction

We are interested in fissile matter detection and characterisation. We want to determine the fissile source with external measures. Times of neutron detections during an interval of measure provides the observations.

We study here an inverse problems under limited data. The inverse problem is ill-posed, getting the entries of the model is challenging. To tackle this issue we use bayesian methods, the a posteriori distribution provides the relative probability of the entries knowing the measures, our observation [12]. In order to sample this distribution we will use a MCMC method: the Metropolis-Hastings algorithm.

Since the distribution is degenerate when the measures are extensive, we will use an Adaptive-Metropolis algorithm with Covariance Matrix Adaptation.

The paper is organized as follows. First, we introduce the neutron point model and expose the expressions of the simple moments of the neutrons count distribution [7].

Secondly, we will recall Bayes rules, present the requirements for the sampling and expose the given covariance for the measures. And we will also present the sampling of the a posteriori distribution, the discretisation with 3 parameters.

Finally, we will expose the results of the sampling with a benchmark. We will analyse the features of the sampled distribution with an explicit sampling and MCMC one, and settle how the work can be improved.

2 Stochastic neutronics problem, forward problem

The simplest model in neutronics is the point model approximation.

Definition 2.1 *Point model* [11]

The medium is infinite, homogeneous and isotropic. The neutrons are supposed point particles moving at the same speed. Moreover, we consider the neutron's life ends with a capture (with or without a detection) or a fission. These events are poissonian type. Neutrons are produced by fission and by the Poisson or compound Poisson type sources. A fission chain is modeled as a branching process.

The model is governed by the following parameters

2.1 Source

We model the source as a compound Poisson process with a strength

Definition 2.2

$$S := \text{Intensity of the compound Poisson process} \quad (1)$$

The probability distribution of the number of neutrons emitted by a source event is given by

$$f_{\nu,S} \quad (2)$$

where ν goes from 0 to the maximum number of neutrons emitted by the source $\nu_{max,S}$. The mean number of neutrons emitted by one source event is

$$\bar{\nu}_S := \sum_{\nu=0}^{\nu_{max,S}} \nu f_{\nu,S}. \quad (3)$$

From this, we can derive the following nuclear constants.

Definition 2.3 *The Diven factors of order 2 and 3 of the source probability distribution are*

$$D_{2S} := \frac{\sum_{\nu} \nu(\nu-1) f_{\nu,S}}{\bar{\nu}_S^2}, \quad D_{3S} := \frac{\sum_{\nu} \nu(\nu-1)(\nu-2) f_{\nu,S}}{\bar{\nu}_S^3} \quad (4)$$

2.2 Fission

Definition 2.4 Let p be the probability that a neutron causes a fission (so $1 - p$ is the probability that a neutron be captured).

The probability distribution of the number of neutrons produced by a fission is

$$f_\nu \quad (5)$$

where ν goes from 0 to the maximum number of neutrons emitted by the fission ν_{max} , and

$$\bar{\nu} := \sum_{\nu=0}^{\nu_{max}} \nu f_\nu \quad (6)$$

the mean number of neutrons emitted by one source event. When a fission occurs $\bar{\nu}$ neutrons are emitted on average.

Then

$$k := \bar{\nu}p \quad (7)$$

is the mean number of children of a neutron. We will call it the multiplication factor [2].

In our case $0 < k < 1$, so the system is stationary which is the most important configurations for nuclear safety applications [9].

As previously, we obtain the formulas of the Diven factors of the fission of order 2 and 3.

$$D_2 := \frac{\sum_\nu \nu(\nu - 1)f_\nu}{\bar{\nu}^2}, \quad D_3 := \frac{\sum_\nu \nu(\nu - 1)(\nu - 2)f_\nu}{\bar{\nu}^3} \quad (8)$$

Definition 2.5 The fission rate is

$$\lambda_F \quad (9)$$

2.3 Capture

The neutron count is the action of detecting the neutron presence.

Definition 2.6 The capture rate is

$$\lambda_C \quad (10)$$

Definition 2.7 We define the capture efficiency by

$$\varepsilon_C := \text{Probability that a captured neutron is detected} \quad (11)$$

This efficiency is linked to the fission one ε_F by the equality

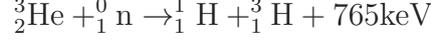
Definition 2.8

$$\varepsilon_F := \text{Detector efficiency} = \frac{\lambda_C \varepsilon_C}{\lambda_F} \quad (12)$$

this is the ratio of the mean number of detections over the mean number of induced fissions.

2.4 Measurements

We get our observations from a detector with Helium 3 [15]. Neutronics choose this element because its cross-section is large, so the capture probability is high. Neutrons are absorbed in the detector with the reaction



then the proton emerging from the reaction causes an electric current. During a time interval of duration T_{meas} , each instant of detection is stored as a list in file (see fig. 1). Then we obtain $n = \lfloor \frac{T_{meas}}{T} \rfloor$ realizations of $N_{[0,T]}$ the counted neutrons during a time gate T , and we compute the empirical moments of this distribution

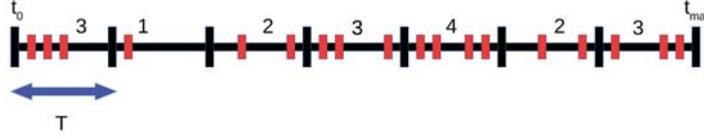


Figure 1: A measurement during t_0 and $t_{max} = t_0 + T_{meas}$, with a time gate T

2.5 Forward model

Definition 2.9 Let $N_{[0,T]}$ be the random variable representing the neutron counts during T . The three first associated moments of this distribution are

- $\mathbb{E}[N_{[0,T]}] :=$ the first moment of the neutron counted during T distribution
- $\mathbb{E}[N_{[0,T]}^2] :=$ the second simple moment of the neutron counted during T distribution
- $\mathbb{E}[N_{[0,T]}^3] :=$ the third simple moment of the neutron counted during T distribution

These are the outputs of our model.

In the case of the point model the simple moments can be expressed as a function of the so-called Feynman moments $Y_2(T)$, $Y_3(T)$ [9].

Proposition 2.10 The first three simple moments of $N_{[0,T]}$ are of the form

$$\begin{aligned} \mathbb{E}[N_{[0,T]}] &= \bar{\nu}_S S \frac{\varepsilon_F k}{(1-k)\bar{\nu}} T \\ \mathbb{E}[N_{[0,T]}^2] &= \mathbb{E}[N_{[0,T]}](1 + \mathbb{E}[N_{[0,T]}] + Y_2) \\ \mathbb{E}[N_{[0,T]}^3] &= \mathbb{E}[N_{[0,T]}](1 + 3Y_2 + Y_3) + 3\mathbb{E}[N_{[0,T]}]^2(1 + Y_2) + \mathbb{E}[N_{[0,T]}]^3 \end{aligned} \quad (13)$$

where the Feynman moments are given by

$$\begin{aligned} Y_2(T) &= \frac{\varepsilon_F D_2 k}{(k-1)^2} \left(1 - \rho \frac{\bar{\nu}_S D_2 S}{\bar{\nu} D_2}\right) \left(1 - \frac{1 - e^{-\alpha_Y T}}{\alpha_Y T}\right) \\ Y_3(T) &= 3 \left(\frac{\varepsilon_F D_2 k}{-(1-k)^2}\right)^2 \left(1 - \rho \frac{\bar{\nu}_S D_2 S}{\bar{\nu} D_2}\right) \left(1 + e^{-\alpha_Y T} - 2 \frac{1 - e^{-\alpha_Y T}}{\alpha_Y T}\right) \\ &\quad - \frac{\varepsilon_F D_3 k^3}{(k-1)^3} \left(1 - \frac{k-1}{k} \frac{\bar{\nu}_S^2 D_3 S}{\bar{\nu}^2 D_3}\right) \left(1 - \frac{3 - 4e^{-\alpha_Y T} + 2e^{-2\alpha_Y T}}{\alpha_Y T}\right) \end{aligned} \quad (14)$$

where $\alpha_Y = \lambda_C + \lambda_F(1 - \bar{\nu})$

A proof of this result can be found in [7].

Our forward model is

$$\begin{aligned} \mathbf{M} : \mathbb{R}^3 &\rightarrow \mathbb{R}^3 \\ \mathbf{p} &\mapsto \mathbf{M}(\mathbf{p}) \end{aligned} \quad (15)$$

where $\mathbf{p} = (\varepsilon_C, k, S)$ and $\mathbf{M}_j(\mathbf{p}) = \mathbb{E}[N_{[0,T]}^j]$.

3 Bayesian inverse problem

3.1 Bayes principle

We have the observations $\hat{\mathbf{M}}$ which are the estimated moments of $N_{[0,T]}$. Bayes theorem [13] states

$$\underbrace{\mathbb{P}(\mathbf{p}|\hat{\mathbf{M}})}_{\text{a posteriori distribution}} \propto \underbrace{\mathbb{P}(\hat{\mathbf{M}}|\mathbf{p})}_{\text{likelihood}} \underbrace{\mathbb{P}(\mathbf{p})}_{\text{a priori distribution}} \quad (16)$$

where the likelihood and the a priori distribution are as follows

1. Thanks to the Central Limit Theorem, given the parameter \mathbf{p} the measures are Gaussian with mean $\mathbf{M}(\mathbf{p})$ and covariance $\frac{1}{n} \mathbf{Cov}(\mathbf{p})$ where \mathbf{M} refers to the expression of the exact simple moments of the distribution of $N_{[0,T]}$, $\mathbf{Cov}(\mathbf{p})$ the covariance matrix of the three first simple moments, n the number of realizations.

This gives explicitly

$$\mathbb{P}(\hat{\mathbf{M}}|\mathbf{p}) \propto \frac{1}{\sqrt{\det(\frac{1}{n} \mathbf{Cov}(\mathbf{p}))}} e^{-\frac{1}{2} {}^t(\hat{\mathbf{M}} - \mathbf{M}(\mathbf{p})) \mathbf{Cov}(\mathbf{p})^{-1} (\hat{\mathbf{M}} - \mathbf{M}(\mathbf{p}))n} \quad (17)$$

which is the expression of the likelihood up to a multiplicative constant.

The computation of $\mathbf{Cov}(\mathbf{p})$ needs the expression of the simple moments up to the order 6, and this is too complex to be computed analytically. So we will use the empirical covariance matrix $\widehat{\mathbf{Cov}}$.

$$\tilde{\mathbb{P}}(\hat{\mathbf{M}}|\mathbf{p}) \propto \frac{1}{\sqrt{\det(\frac{1}{n} \widehat{\mathbf{Cov}})}} e^{-\frac{1}{2} {}^t(\hat{\mathbf{M}} - \mathbf{M}(\mathbf{p})) \widehat{\mathbf{Cov}}^{-1} (\hat{\mathbf{M}} - \mathbf{M}(\mathbf{p}))n} \quad (18)$$

2. The a priori distribution is assumed to be uniform on $[\varepsilon_{C,min}, \varepsilon_{C,max}] \times [k_{min}, k_{max}] \times [S_{min}, S_{max}]$.

Our goal is to sample the a posteriori distribution 16. We will use two different methods: a discrete sampling with a regular mesh and Adaptive Metropolis with Covariance Matrix Adaptation.

3.2 Explicit sampling of the a posteriori distribution

A simple way to obtain the explicit sampling of the a posteriori distribution is to use a regular mesh of the domain $[\varepsilon_{C,min}, \varepsilon_{C,max}] \times [k_{min}, k_{max}] \times [S_{min}, S_{max}]$ and compute the a posteriori distribution on each point of the mesh. The computations were done with N_e points in each directions. The overall number of evaluations of the forward model 15 is therefore N_e^3 . We also compute the moments in order to have some quantitative information: mean, variance, expectation to be compared with MCMC results.

Remark 3.1 By 16 the computation of the likelihood is true up to a multiplicative constant.

3.3 MCMC sampling of the a posteriori distribution

The principle of the method is to build a Markov chain that has the target distribution as its stationary distribution. Hence one can obtain a sample of the target distribution by sampling and recording states from the chain. Various algorithms exist for constructing such Markov chains, including the Metropolis-Hastings (MH) algorithm. The states of the MH chain are produced iteratively. At each iteration, the algorithm picks a random proposal according to some instrumental distribution that may depend on the current sample value. The proposal is the candidate for the next sample value and it is either accepted (in which case the proposal value is used in the next iteration) or rejected (in which case the proposal value is discarded, and the current value is used in the next iteration) with some probability. The probability of acceptance is determined by comparing the values of the target density at the current and proposal values so as to ensure that the MH chain has the target distribution as its stationary distribution.

We implement here a specific MCMC method [10]: the Adaptive Metropolis algorithm with Covariance Matrix Adaptation in order to sample a target distribution π .

The adaptation uses the Covariance matrix of the all the points proposed by the instrumental law and accepted by the rejection procedure in order to accept more.

We implemented the following algorithm using [1] and [6]. Here

$$\mathbf{p} = (p_1, p_2, p_3) \quad (19)$$

and we define $p_{1,min} = \varepsilon_{C,min}$, $p_{1,max} = \varepsilon_{C,max}$, $p_{2,min} = k_{min}$, $p_{2,max} = k_{max}$, $p_{3,min} = S_{min}$, $p_{3,max} = S_{max}$ The target distribution is denoted π .

We first initiate

- The empirical acceptance x_{rate} .
- The initial scale factor of the instrumental law $frac$
- The target acceptance rate x_{obj}
- The frequency of update of the scale factor $N_{MC,1}$
- The burnin phase duration N_{bp}
- The initial parameter \mathbf{p}_0 is chosen with the uniform distribution over $\bigotimes_{k=1}^3 [p_{k,min}, p_{k,max}]$

Iteration $i \rightarrow i + 1$

1. During $i \leq N_{bp}$, we use as instrumental law

$$\mathbf{q}_{i+1} \sim \mathcal{N}(\mathbf{p}_i, frac^2 \mathbf{C}_{bi}) \quad (20)$$

where \mathbf{q}_{i+1} the proposal, \mathbf{p}_i the last accepted point, $\mathbf{C}_{bi} = diag(((p_{k,max} - p_{k,min})^2)_{k=1}^3)$.

2. After the burnin we use the instrumental law as in algorithm 4 of [1]

$$\mathbf{q}_{i+1} \sim \mathcal{N}(\mathbf{p}_i, frac^2 \mathbf{C}_i) \quad (21)$$

where \mathbf{C}_i is defined by 24.

3. Finally we compute the acceptance rate α using the likelihood ratio of the proposal and the previous accepted point

$$\alpha(\mathbf{q}_{i+1}, \mathbf{p}_i) = \min\left(1, \frac{\pi(\mathbf{q}_{i+1})}{\pi(\mathbf{p}_i)}\right) \quad (22)$$

4. The acceptance-rejection criterion

$$u \sim \mathcal{U}([0, 1]) \quad (23)$$

If $u \leq \alpha(\mathbf{q}_{i+1}, \mathbf{p}_i)$ then \mathbf{q}_{i+1} is accepted: $\mathbf{p}_{i+1} = \mathbf{q}_{i+1}$ otherwise $\mathbf{p}_{i+1} = \mathbf{p}_i$ is applied

5. We update the scale factor when $i \equiv 0 \pmod{N_{MC,1}}$. We update the scale factor $frac$

$$frac = frac \exp(x_{rate} - x_{obj})$$

$$x_{rate} = \frac{\text{Number of acceptance}}{\text{Number of iterations}}$$

This is an algorithm with global scaling and with vanishing adaptation so the ergodicity of the algorithm is achieved. The vanishing factor $\gamma_i = \frac{1}{i}$, it must be chosen as $\sum_i \gamma_i = +\infty$ [1].

So that

$$\mathbf{C}_i = \frac{1}{k+1} \left(\sum_{i=0}^k \mathbf{p}_i \mathbf{p}_i^T + (k+1) \bar{\mathbf{p}}_k \bar{\mathbf{p}}_k^T \right) \quad (24)$$

where $\bar{\mathbf{p}}_i = \frac{1}{i+1} \sum_{k=0}^i \mathbf{p}_k$ as in [6]. The covariance matrix is updated as in [8]

$$\begin{aligned} \mathbf{C}_{i+1} &= (1 - \gamma_{i+1}) \mathbf{C}_i + \gamma_{i+1} (\mathbf{q}_i - \bar{\mathbf{p}}_i) (\mathbf{q}_i - \bar{\mathbf{p}}_i)^T \\ \bar{\mathbf{p}}_{i+1} &= (1 - \gamma_{i+1}) \bar{\mathbf{p}}_i + \gamma_{i+1} \mathbf{q}_i \end{aligned} \quad (25)$$

The empirical covariance matrix and the mean proposal are updated as follows

The target $x_{obj} = 0.234$ is chosen thanks to [4]. Since the a posteriori distribution can be really degenerate the use of \mathbf{C} and the global factor adaptation allows to sample well the distribution, even if highly degenerated.

4 Test-cases, numerical application

The real parameter will be

$$\mathbf{p}^* = \begin{pmatrix} \varepsilon_C \\ k \\ S \end{pmatrix} = \begin{pmatrix} 0.25 \cdot 10^{-3} \\ 0.5 \text{ or } 0.75 \text{ or } 0.95 \\ 70 \text{ ms}^{-1} \end{pmatrix} \quad (26)$$

In the following figures the cross represents \mathbf{p}^* , the observations $\hat{\mathbf{M}}$ are the values of the simple moments of \mathbf{p}^* .

We use the following bounds for the a priori distribution

$$\begin{aligned} \varepsilon_{C,min} &= 0.1 \cdot 10^{-2} \\ \varepsilon_{C,max} &= 0.4 \cdot 10^{-2} \\ k_{min} &= 0 \\ k_{max} &= 1 \\ S_{min} &= 20 \\ S_{max} &= 200 \end{aligned} \quad (27)$$

The quantity of interest is $N_{[0,T]}$ when $T = 10 \text{ ms}$ and for a time of measurement of $T_{meas} = 36, 360, 3600 \text{ s}$. We have also considered $\alpha_Y = 2 \text{ ms}^{-1}$ and

$$\begin{aligned} \bar{\nu} &= 2.4130 & \bar{\nu}_S &= 1.000 \\ D_2 &= 0.7992 & D_{2S} &= 0 \\ D_3 &= 0.4819 & D_{3S} &= 0 \end{aligned} \quad (28)$$

The initialization parameters of the AM algorithm are

$$\begin{aligned} x_{rate} &= 1 \\ x_{obj} &= 0.234 \\ frac &= 0.1 \\ N_{bp} &= 10^7 \\ N_{MC,1} &= \max\left(\frac{N_{MC}}{10000}, 1\right) \end{aligned} \quad (29)$$

Regarding the explicit sampling there are N_e^3 points in the grid with $N_e = 400$.

The 2D-a posteriori distribution of the parameter \mathbf{p} are estimated by histograms with 100×100 from the AM sample.

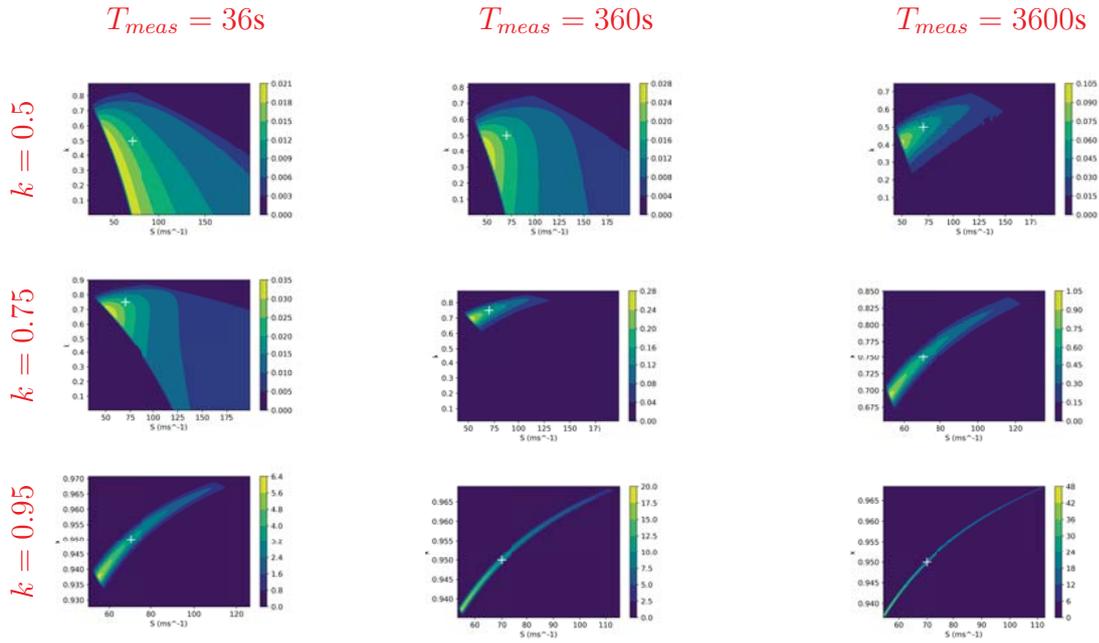


Figure 2: A posteriori distribution for (k, S) using 3P3M with explicit sampling

We can compare it to the result using the MCMC method

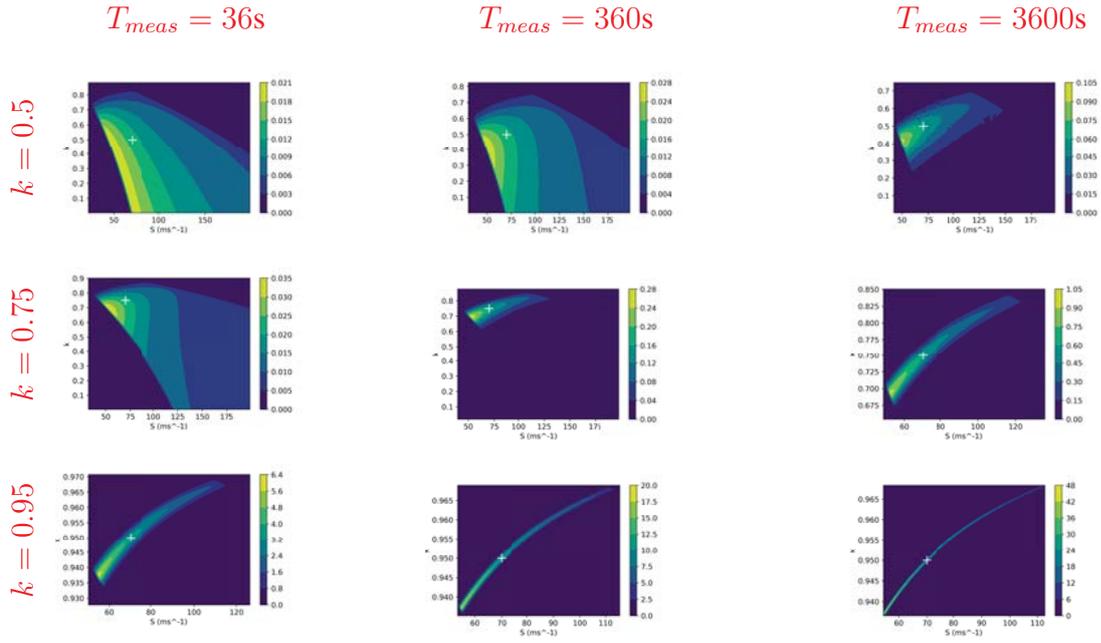


Figure 3: A posteriori distribution for (k, S) using 3P3M with MCMC sampling

We can also observe the a posteriori distribution for (k, ε_C)

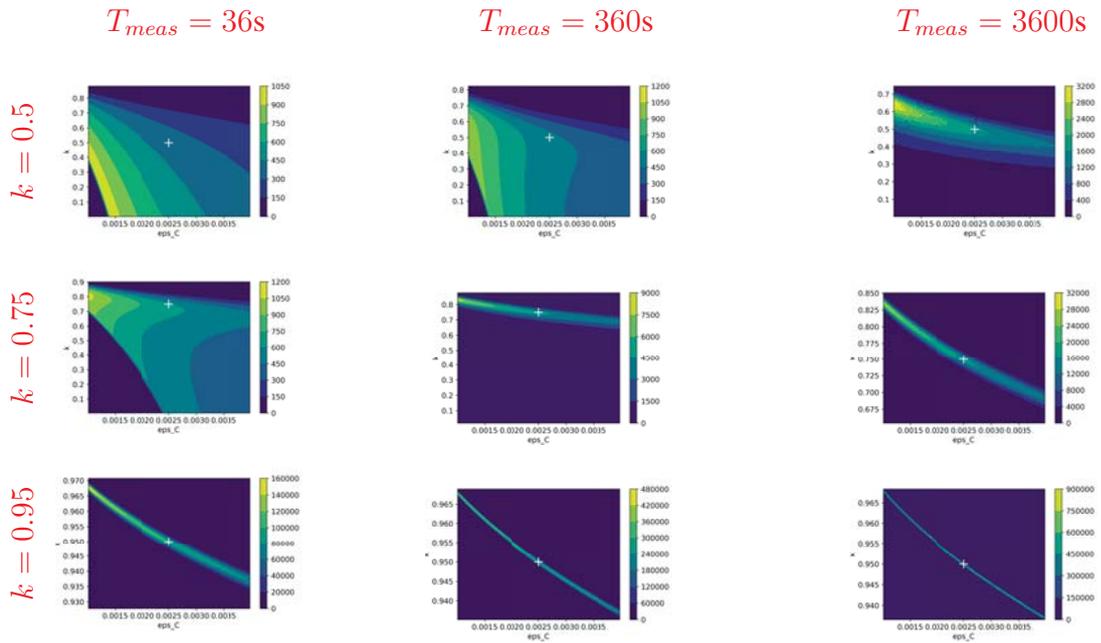


Figure 4: A posteriori distribution for (k, ε_C) using 3P3M with explicit sampling

We can compare these results to the MCMC method results

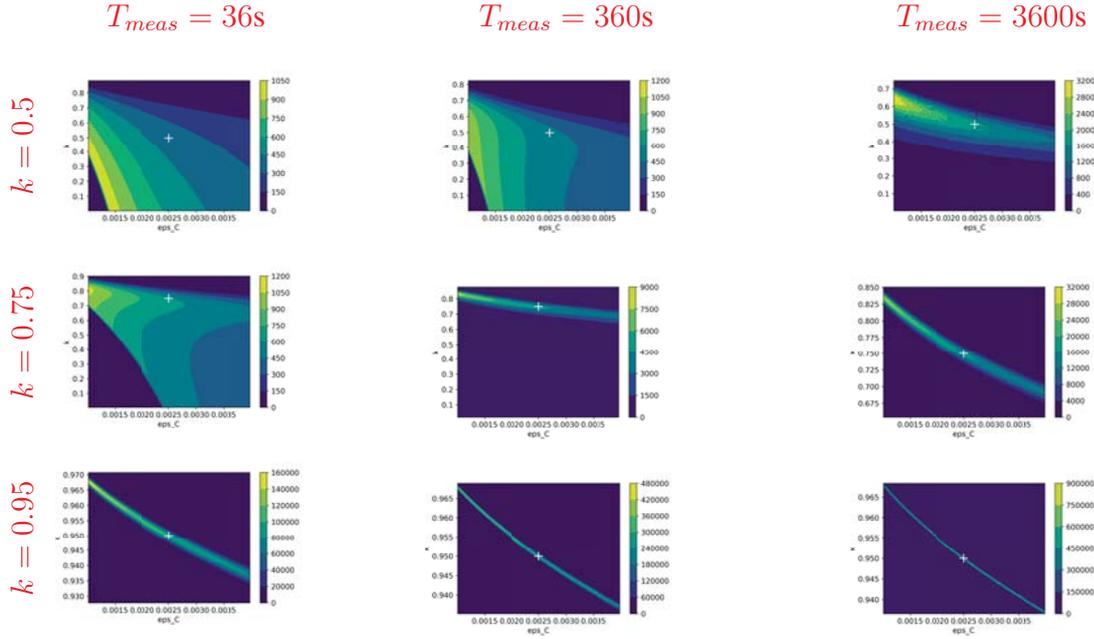


Figure 5: A posteriori distribution for (k, ε_C) using 3P3M with MCMC

5 Discussion

The explicit sampling of the distribution is really effective when the dimension of \mathbf{p} is up to three.

When the dimension is larger than four, the AM algorithm is efficient whatever the dimension of \mathbf{p} is.

We observe the higher the multiplication factor k the more the distribution is degenerate. The same effects appear when the time of measurement T_{meas} is large. The real value \mathbf{p}^* is in the support of the distribution, but this support is large.

6 Conclusion

To sum up, in the context of the neutron point model we have used the analytic expression of the three first simple moments which define the forward model of our inverse problem. Our observations are the estimation of the three first empirical moments of the neutron count distribution. Then using the Bayes principle, we have exposed the estimation of the a posteriori distribution of the parameters. Then we have implemented two sampling methods of this a posteriori distribution:

- The first method is a simple sampling with a regular grid whose cost dramatically increases with the dimension of the parameter.
- The second method is obtained by the use of the Adaptive Metropolis algorithm with Covariance Matrix Adaptation

On an example with synthetic data, we observed that the support of the distribution contains the true parameter. The distribution is more degenerate when the multiplication factor k is high, and also when T_{meas} is large.

The explicit sampling is well adapted when $\dim(\mathbf{p}^*) \leq 3$, but it is too expensive when this condition is not satisfied. Then the use of the AM-CMA approach is required.

The present work shows that the sampling (explicit or with AM-CMA) is satisfactory to retrieve the true for parameter for one time gate T when $\dim(\mathbf{p}^*) \leq 3$. Considering two time gates T_1 and T_2 will enable recovering a parameter of higher dimension.

REFERENCES

- [1] C. Andrieu, J. Thoms *A tutorial on adaptive MCMC*, Stat Comput, 18, 343–373, 2008
- [2] G. I. Bell and S. Glasstone, *Nuclear Reactor Theory*, 1970, Van Nostrand Reinhold Company
- [3] R. P. Feynman, F. de Hoffmann, R. Serbe, *Statistical fluctuations in the water boiler and the dispersion of neutrons emitted per fission*, 322, 10, 891–921, 1944
- [4] A. Gelman, G.O. Roberts, W. R. Gilks, *Efficient Metropolis Jumping Rules*, Bayesian Statistics, 5, 599-607, 1996
- [5] H. Haario, E. Saksman and J Tamminen, *Adaptive proposal distribution for random walk Metropolis algorithm*, Computational Statistics, 14, 1, 375-395, 1999
- [6] H. Haario, E. Saksman and J Tamminen, *An adaptive Metropolis algorithm*, Bernoulli, 7, 2, 223-242, 2001
- [7] P. Humbert, *Simulation and Analysis of List Mode Measurements on SILENE Reactor*, Journal of Computational and Theoretical Transport, 47, 4-6, 350-363, 2018
- [8] C. L. Müller, *Exploring the common concepts of adaptive MCMC and Covariance Matrix Adaptation schemes* Dagstuhl Seminar Proceedings 10361, Theory of Evolutionary Algorithms
- [9] I. Pazsit, L. Pal, *Neutron Fluctuations A Treatise on the Physics of Branching Processes*, 2008, Elsevier Ltd, London, New York, Tokyo
- [10] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. 2004, Springer Texts in Statistics. Springer-Verlag, New York
- [11] J. Saxby, *Numerical Solution of the Phase-Space Dependent Backward Master Equation for the Probability Distribution of Neutron Number in a Subcritical Multiplying Sample*, Imperial College London Department of Mechanical Engineering, 2017
- [12] T.J. Sullivan, *Introduction to Uncertainty Quantification*, 2015, Springer International Publishing
- [13] A. Tarantola, *Inverse Problem Theory and Methods for Model Parameter Estimation*, 2005, SIAM
- [14] J. Terrell, *Distribution of Fission Neutron Numbers*, Physical Review, 108, 3, 783-789, 1957
- [15] N. Tsoulfanidis, S. Landberger, *Measurements & Detection of radiation 4th edition*, 2015, CRC Press

MACHINE LEARNING AIDED STOCHASTIC SLOPE STABILITY ANALYSIS

Zhanpeng Liu¹, Di Wu², Daichao Sheng³, Behzad Fatahi⁴, Hadi Khabbaz⁵

^{1 2 3 4 5} University of Technology Sydney
School of Civil and Environmental Engineering, University of Technology Sydney, Sydney NSW
2007, Australia

e-mail: ¹zhanpeng.liu@student.uts.edu.au

²di.wu-1@uts.edu.au

³daichao.sheng@uts.edu.au

⁴behzad.fatahi@uts.edu.au

⁵hadi.khabbaz@uts.edu.au

Abstract

This paper presents the study in the machine learning aided stochastic slope stability analysis through the finite element method. The probability of failure of a dam with cohesive slope has been investigated. The numerical model has been built by the finite element method. An advanced machine learning algorithm called Extreme Learning Machine (ELM) is adopted to establish the regression model. The applicability and effectiveness of the presented approach are compared by the Monte-Carlo simulation method.

Keywords: Slope stability analysis, machine learning, finite element method, experimental design.

1 INTRODUCTION

Slope stability is one of the principal considerations in soil mechanics such as rainfall induced landslide analysis and strength limit design of dam embankment. A series of research can be found in the slope stability analysis [1-11]. Herein, D.V. Griffiths presented research in slope stability analysis by finite elements methods [1], followed by the probabilistic slope stability analysis [2,3]. Nonlinear behaviour includes the nonlinear failure criterion and the yielding criteria on the elasto-plastic analysis are investigated [4,5]. Reliability analysis for slope stability is presented by J.T. Christian et al. [6] and J.M. Duncan [7]. C.C. Huang et al. have conducted research in 3D slope stability analysis [8, 9]. Research shows, because of the uncertainty of the soil properties and the large-scale variation of the nature environment, the deterministic analysis is not always leading the results that reflect the real-world situation. Thus, the non-deterministic analysis is of favoured in such a design, especially the reliability analysis for the slope stability.

This study presents a machine learning aided stochastic slope analysis by using extreme learning machine (ELM). ELM is a single hidden layer forward neural network (SLFN) that originally proposed by G.B. Huang [11], based on the four-layered feedforward neural network versus three [12, 13]. Figure 1 illustrates the architecture of the classical ELM algorithm. The performance of the regression and multiclass classification is further discussed in [14, 15]. Researchers investigated on the improvements of the ELM algorithms and the effectiveness of its extension including bidirectional extreme learning machine (B-ELM) [16]; incremental extreme learning machine (I-ELM) [17-19]; on-line sequential extreme learning machine (OS-ELM) [20]; OS-ELM with kernel [21] and so on. G. Huang has summarised the current research work in ELM in a review article up to 2015 [22]. In this paper, a numerical investigation has been conducted for the dam embankment model demonstrated in Figure 2. The results are compared with the conventional Monte Carlo Simulation method to demonstrate the accuracy and the efficiency of the presented method.

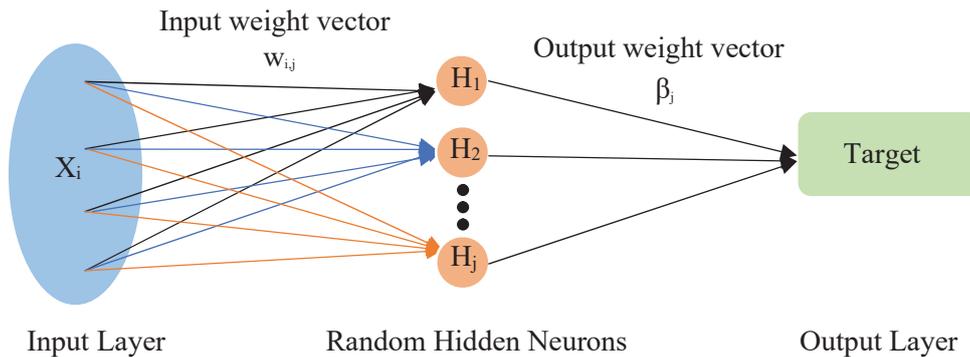


Figure 1: Classical ELM algorithm architecture

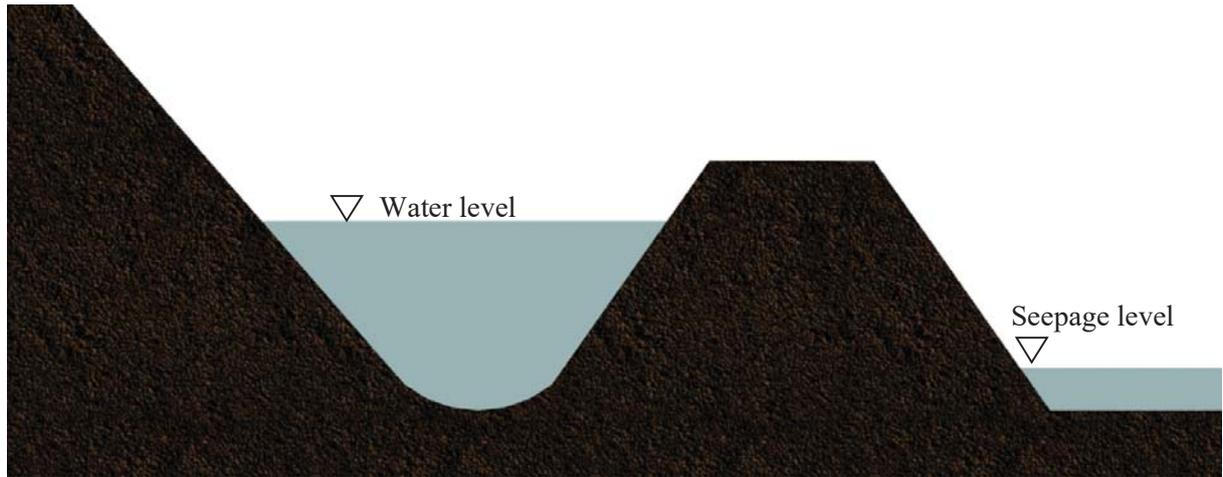


Figure 2: The illustration of the dam embankment

2 PRELIMINARIES

2.1 Deterministic slope stability analysis by FE models

The deterministic slope stability analysis has been widely investigated by finite element analysis [1-6], due to the page limitation, the fundamental of the finite element analysis will not be repeated. In this study, the slope stability analysis has been conducted by shear strength reduction method according to previous research [1-7]. The factor of safety (FOS) is used to reduce the cohesion thus results in a reduction of the soil shear strength; the FOS is formulated as:

$$\text{FOS} = \frac{c}{c'} \quad (1)$$

where c is the material cohesion and c' is parameterised cohesion.

The Mohr-coulomb yield function is adopted, the associated plasticity can be demonstrated as:

$$F = \frac{\sigma_1' + \sigma_3'}{2} \sin(\phi') - \frac{\sigma_1' - \sigma_3'}{2} - c' \cos(\phi') \quad (2)$$

where when $F > 1$, the soil is yielding thus the stresses are redistributed. The failure of slope is determined in displacements, at a certain FOS value, the result does not converge due to the increment of reduction of the soil shear strength, which indicated the slope failure due to the instability.

2.2 Reliability analysis through Extreme Learning Machine (ELM)

In this section, the algorithm of the extreme learning machine is introduced. As a single hidden layer forward neural networks, the goal is to minimise the cost function:

$$Y = \sum_{j=1}^N \left(\sum_{i=1}^{\tilde{N}} \beta_i g(w_i \cdot x_j + b_i) - t_j \right)^2 \quad (3)$$

for N samples and \tilde{N} hidden nodes. The input weight vector w is randomly generated based on a continuous probability density function in an interval $[-1, 1]$ [11], b is the bias vector. The $g(x)$ is an activation function, in this study, Sigmoid function is adopted:

$$g(x) = \frac{1}{1 + e^{-x}} \quad (4)$$

The hidden layer output matrix \mathbf{H} is formulated as:

$$\mathbf{H} = \begin{bmatrix} g(\mathbf{w}_1 \cdot \mathbf{x}_1 + b_1) & \dots & g(\mathbf{w}_{\tilde{N}} \cdot \mathbf{x}_1 + b_{\tilde{N}}) \\ \dots & \dots & \dots \\ g(\mathbf{w}_1 \cdot \mathbf{x}_N + b_1) & \dots & g(\mathbf{w}_{\tilde{N}} \cdot \mathbf{x}_N + b_{\tilde{N}}) \end{bmatrix} \quad (5)$$

leads to the output of ELM is given as follow:

$$f(x) = \sum_{i=1}^N \beta_i h_i(x) = \mathbf{H}\beta \quad (6)$$

where β is known as the output weight vector that connects the hidden nodes and the output nodes, β can be calculated by:

$$\beta = \mathbf{H}^\dagger \mathbf{T} \quad (7)$$

where \mathbf{H}^\dagger is the Moore-Penrose generalized inverse of the hidden layer output matrix \mathbf{H} [11].

3 NUMERICAL INVESTIGATION

For the purpose of demonstration, a numerical example is investigated by adopting the presented ELM algorithm and then compared with the Monte Carlo Simulation (MCS) method. In this example, the horizontal displacement and vertical displacement of a point P within the dam body are selected as targets for the ELM regression. The finite element model is illustrated in Figure 3. Soil properties are chosen to be random variables, the Young's modulus, Poisson's ratio and soil porosity are uniformly distributed [1], where $E \sim U(0.995 \times 10^5, 1.005 \times 10^5)$ kPa, $\nu \sim U(0.2487, 0.2512)$ and porosity $p \sim U(0.398, 0.402)$; the soil density, soil cohesion and parameter $\tan(\phi)$ are lognormal distributed [2, 7], where $\rho_{\text{mean}} = 2000 \text{ kg/m}^3$, $\rho_{\text{sd}} = 60 \text{ kg/m}^3$, $c_{\text{mean}} = 10 \text{ kPa}$, $c_{\text{sd}} = 3 \text{ kPa}$, $\tan(\phi_{\text{sat}})_{\text{mean}} = 0.5774$, $\tan(\phi_{\text{sat}})_{\text{sd}} = 0.0115$, and $\tan(\phi_{\text{unsat}})_{\text{mean}} = 0.3640$, $\tan(\phi_{\text{unsat}})_{\text{sd}} = 0.0073$. 100 training samples are generated by latin hyper cube sampling method, computed by FEM and trained by ELM. In result verification, 10,000 samples are generated by the MCS method and computed by FEM. The results are compared between the trained ELM and MCS. The accuracy of the presented method is illustrated in Figure 4 and Figure 5. For the results presented in Figures 4 and 5, the total computational time of the MCS is over 200 hours, versus the presented ELM in 2 hours. It worth to mention that, FEM computation dominates the time consumption of ELM, by using the trained ELM, the testing time of newly generated 10,000 samples is within 1 second. In this example, following computer configurations are used:

System: Microsoft Windows 10
 CPU: Intel Core i7 – 8665U 1.9 GHz
 RAM: 16.0 GB

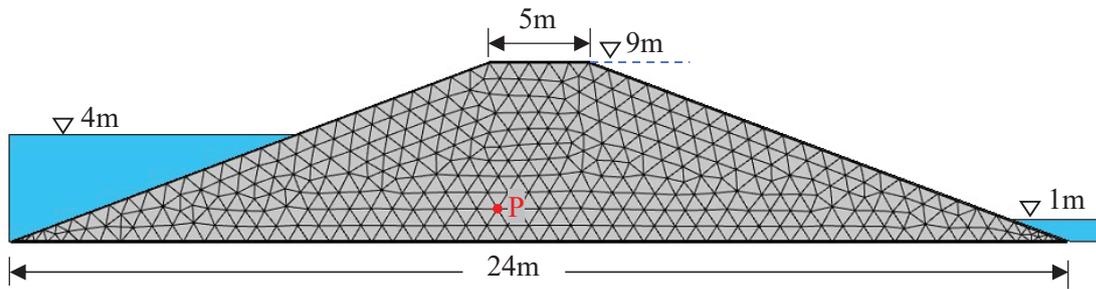


Figure 3: Numerical example in FEM

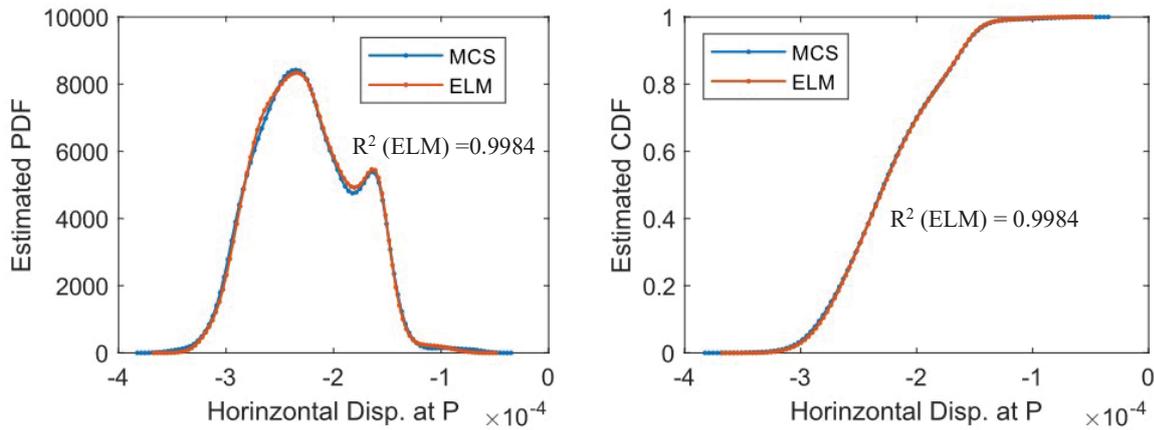


Figure 4: Horizontal displacement of the selected point. Unit: meter

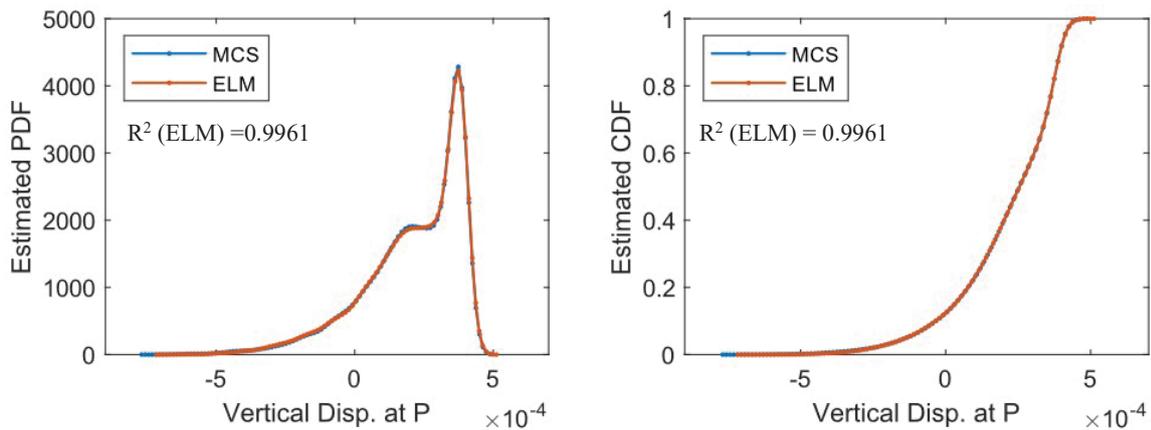


Figure 5: Vertical displacement of the selected point. Unit: meter

4 CONCLUSIONS

In this study, the machine learning aided stochastic slope stability analysis by extreme learning machine is conducted. Finite element method is adopted in the deterministic analysis. In order to verify the accuracy of the presented approaches, results are compared with Monte Carlo simulation. Results show the presented method agreed well with the Monte Carlo simulation while the computation efficiency is improved from 200 hours to 2 hours. Furthermore, the results have been illustrated and compared by probability density function and cumulative density function.

5 ACKNOWLEDGEMENT

This research is supported by an Australian Government Research Training Program Scholarship.

REFERENCES

- [1] D.V. Griffiths, P.A. Lane, Slope stability analysis by finite elements. *Géotechnique*, **49**, No. 3, 387 – 403, 1999.
- [2] D.V. Griffiths, G.A. Fenton, Probabilistic slope stability analysis by finite elements. *Journal of Geotechnical and Geoenvironmental Engineering*, **130:5**, 2004
- [3] D.V. Griffiths, J. Huang, G.A. Fenton. Probabilistic infinite slope analysis. *Computers and Geotechnics*. **38**, 577-584, 2011
- [4] X.L. Yang, J.H. Yin, Slope stability analysis with nonlinear failure criterion. *Journal of Engineering Mechanics*, **130(3)**, 267-273, 2004
- [5] H. Zheng, D.F. Liu, C.G. Li, Slope stability analysis based on elasto-plastic finite element method. *International Journal for Numerical Methods in Engineering*, **64**, 1871-1888, 2005.
- [6] J.T. Christian, C.C. Ladd, G.B. Baecher, Reliability applied to slope stability analysis. *Journal of Geotechnical Engineering*, **120(12)**, 2180-2207, 1994.
- [7] J.M. Duncan, Factors of safety and reliability in geotechnical engineering. *Journal of Geotechnical and Geoenvironmental Engineering*, **126(4)**, 307-316, 2000.
- [8] G. Tang, J. Huang, D. Sheng, S.W. Sloan. Stability analysis of unsaturated soil slopes under random rainfall patterns. *Engineering Geology*, **245**, 322-332, 2018.
- [9] C.C. Huang, C.C. Tsai, Y.H. Chen, Generalized method for three-dimensional slope stability analysis. *Journal of Geotechnical and Geoenvironmental Engineering*, **128:10**, 836-848, 2002.
- [10] C.C. Huang, C.C. Tsai, New method for 3D and asymmetrical slope stability analysis. *Journal of Geotechnical and Geoenvironmental Engineering*, **126(10)**, 917-927, 2000.
- [11] G.B. Huang, Q.Y. Zhu, C.K. Siew, Extreme learning machine: Theory and applications. *Neurocomputing*, **70**, 489-501, 2006
- [12] G.B. Huang, H.A. Babri, Upper Bounds on the Number of Hidden Neurons in Feedforward Networks with Arbitrary Bounded Nonlinear Activation Functions. *IEEE Transactions on Neural Networks*, Vol 9, No.1, 1998.
- [13] S. Tamura, M. Tateishi. Capabilities of a four-layered feedforward neural network: four layers versus three. *IEEE Transactions on Neural Networks*, Vol. 8, No.2, 1997.
- [14] G.B. Huang, H. Zhou, X. Ding, R. Zhang, Extreme Learning Machine for Regression and Multiclass Classification, *IEEE Transactions on Systems, Man, And Cybernetics—Part B: Cybernetics*. Vol. 42, No. 2, 2012.
- [15] G.B. Huang, C.K. Siew, Extreme learning machine: a new learning scheme of feedforward neural networks. *International Joint Conference on Neural Networks (IJCNN2004)*, Budapest, Hungary, July 25-29, 2004

- [16] Y. Yang, Y. Wang, X. Yuan, Bidirectional extreme learning machine for regression problem and its learning effectiveness. *IEEE Transactions on Neural Networks*, Vol. 23, No.9, 2012.
- [17] G.B. Huang, L. Chen, Convex incremental extreme learning machine. *Neurocomputing*, **70**, 3056-3065, 2007.
- [18] G.B. Huang, M.B. Li, L. Chen, C.K. Siew, Incremental extreme learning machine with fully complex hidden nodes. *Neurocomputing*, **71**, 576-583, 2008.
- [19] S. Song, M. Wang, Y. Lin. An improved algorithm for incremental extreme learning machine. *Systems science & control engineering*, **8:1**, 308-317, 2020.
- [20] G.B. Huang, N.Y. Liang, H.J. Rong, P. Saratchandran, N. Sundararajan. On-line sequential extreme learning machine. *the IASTED International Conference on Computational Intelligence (CI 2005)*, Calgary, Canada, July 4-6, 2005
- [21] S. Scardapane, D. Comminiello, M. Scarpiniti, A. Uncini, Online sequential extreme learning machine with kernels. *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 26 No. 9, 2015
- [22] G. Huang, G.B. Huang, S. Song, K. You, Trends in extreme learning machines: A review. *Neural Networks*. **61**, 32-48, 2015.

LIMIT REPRESENTATIONS OF IMPRECISE RANDOM FIELDS

Mona M. Dannert^{1,*}, Johannes L. Häufler¹ and Udo Nackenhorst¹

¹Leibniz University Hannover, Institute of Mechanics and Computational Mechanics
Appelstraße 9a, 30167 Hannover
*e-mail: mona.dannert@ibnm.uni-hannover.de

Keywords: Imprecise random fields, interval valued correlation length, Karhunen-Loève expansion, stochastic finite element method

Abstract. *In order to describe spatially uncertain parameters by random fields, the underlying autocorrelation structure in engineering structures is usually not known.. The idea of imprecise random fields is to acknowledge this lack of knowledge by adding epistemic uncertainties. Within this contribution the influence of the correlation length is studied. In particular, it is shown that there exist bounds that limit the case of having no idea at all. This “absolutely no idea p-box” is defined by white noise and the random variable corresponding to the mean value and standard deviation of the imprecise random field. By this, the limits of having “absolutely no idea” can be described without the need of Karhunen-Loève expansion and random field propagation. Then, at least for linear problems, every response in between can be estimated by linear interpolation without any need for sampling.*

1 INTRODUCTION

Stochastic finite element (FE) method aims to describe a models response depending on random input variables such as loads or material parameters. In this context, uncertainties are distinguished into aleatory and epistemic [5]. The former describe the irreducible, intrinsic randomness of a parameter and is classically described by probability theory. The latter is caused by a lack of knowledge or data. However, gaining enough information to reduce epistemic uncertainty is usually limited by finite resources or limited technical capabilities in engineering reality. To consider such epistemic uncertainties, several possibilistic approaches can be used, e.g. interval [9] or fuzzy [8] analyses. An honest approach usually considers both, aleatory and epistemic uncertainties. Alternative approaches on the treatment of these mixed uncertainties, e.g. evidence theory, fuzzy probabilities or probability boxes, have been reviewed for example in [1].

In probability theory, random fields can be used to describe spatially uncertain values in terms of their mean value, standard deviation and autocorrelation structure. While mean value and standard deviation can be estimated by experiments quite easily, the autocorrelation between the random field values at two different locations can hardly be measured. To describe such mixed aleatory and epistemic uncertainties, imprecise random fields have been introduced lately [6]. This approach can be understood as an extension of probability box (p-box) approach towards random fields. The random field parameters that cannot be determined precisely, e.g. the correlation length, can be described as interval [3] or fuzzy valued [10]. Propagating such imprecise random fields through an FE model, the quantity of interest is described by a p-box, meaning a lower and upper bound instead of a crisp distribution. However, by introducing a second loop over the epistemic uncertainties the sampling process can become very expensive. Discretising the individual random fields for each correlation length by Karhunen-Loève (KL) expansion furthermore leads to high-dimensional problems, especially when small correlation lengths are involved [3, 4].

This contribution focuses on the autocorrelation structure where the lack of information is incorporated by an interval valued correlation length. For this purpose, the concept of imprecise random fields and the KL expansion as a method to discretise random fields are introduced in Section 2. It is important to ensure that the imprecise response is not affected by the local or global error arising from the truncation of the KL expansion [4]. Therefore, an imprecise random field input is carefully investigated in Section 3 in terms of the affect of the correlation length and truncation order. Furthermore, the limits of the correlation length towards zero (white noise) and infinity (random variable) are studied. Afterwards, the propagation of imprecise random field input parameters is studied for a linear problem including load and material uncertainties within Section 4. Finally, the results are summarised and concluded in Section 5.

2 IMPRECISE RANDOM FIELDS

The concept of imprecise random fields allows to model one or several parameters of a random field by interval or fuzzy variables. This way, epistemic uncertainties can be added to a classically aleatory random field. Mixed uncertainties are usually propagated by a double loop approach. After discretising the epistemic parameters within an outer loop, the probabilistic problem resulting for each crisp parameter can be solved within the inner loop, e.g. by Monte Carlo (MC) sampling. In case of imprecise random fields this means that each resulting random field needs to be discretised. A well known method for this purpose is given by Karhunen-Loève (KL) expansion [7] which will be shortly summarised in the following subsection. Afterwards

the main idea of the probability box (p-box) approach is illustrated in order to describe and propagate imprecise random fields.

2.1 Random field discretisation

Random fields $X(\mathbf{z}, \omega)$ are parameters depending on space $\mathbf{z} \in \mathcal{D}^d$ and chance $\omega \in \Omega$. They can be described in terms of a mean value $\mu_X(\mathbf{z})$ and an autocovariance function $\text{Cov}(\mathbf{z}_1, \mathbf{z}_2) = \sigma_X(\mathbf{z}_1)\sigma_X(\mathbf{z}_2)\Gamma(\mathbf{z}_1, \mathbf{z}_2)$, where $\sigma_X(\mathbf{z})$ is the standard deviation and $\Gamma(\mathbf{z}_1, \mathbf{z}_2)$ describes the autocorrelation between the random variables assigned to two arbitrary locations \mathbf{z}_1 and \mathbf{z}_2 .

Within this contribution, the standard deviation is assumed to be constant, $\sigma_X(\mathbf{z}) = \sigma_X$. Then the random field can be described in terms of $\mu_X(\mathbf{z})$, σ_X and $\Gamma_X(\mathbf{z}_1, \mathbf{z}_2)$ and its series expansion reads [11]

$$X(\mathbf{z}, \omega) = \mu_X(\mathbf{z}) + \sigma_X \sum_{i=1}^{\infty} \sqrt{\lambda_i} \phi_i(\mathbf{z}) \xi_i(\omega), \quad (1)$$

where ξ_i are independent standard normal distributed random variables. The eigenpairs $\{\lambda_i, \phi_i\}$ are gained by solving

$$\int_{\mathcal{D}} \Gamma_X(\mathbf{z}_1, \mathbf{z}_2) \phi_i(\mathbf{z}_2) d\mathbf{z}_2 = \lambda_i \phi_i(\mathbf{z}_1), \quad (2)$$

where $\Gamma_X(\mathbf{z}_1, \mathbf{z}_2)$ can be decomposed as

$$\Gamma_X(\mathbf{z}_1, \mathbf{z}_2) = \sum_{i=1}^{\infty} \lambda_i \phi_i(\mathbf{z}_1) \phi_i(\mathbf{z}_2). \quad (3)$$

To propagate random field parameters through a model, e.g. a stochastic finite element (FE) problem, the infinite sum in Equation (1) needs to be truncated and the random field is approximated by

$$\hat{X}(\mathbf{z}, \omega) = \mu_X(\mathbf{z}) + \sigma_X \sum_{i=1}^T \sqrt{\lambda_i} \phi_i(\mathbf{z}) \xi_i(\omega). \quad (4)$$

The corresponding expanded autocorrelation function then reads

$$\hat{\Gamma}_X(\mathbf{z}_1, \mathbf{z}_2) = \sum_{i=1}^T \lambda_i \phi_i(\mathbf{z}_1) \phi_i(\mathbf{z}_2) \quad (5)$$

and maintains an error $\epsilon(\mathbf{z})$ depending on T

$$\epsilon(\mathbf{z}) = 1 - \sum_{i=1}^T \lambda_i \phi_i^2(\mathbf{z}). \quad (6)$$

With $\text{Cov}(\mathbf{z}_1, \mathbf{z}_2) = \sigma_X(\mathbf{z}_1)\sigma_X(\mathbf{z}_2)\Gamma(\mathbf{z}_1, \mathbf{z}_2)$, Equation (6) is the normalised equivalent to the often used error variance $\epsilon_{\sigma^2}(\mathbf{z})$, defined e.g. by [2]. Furthermore, the mean error over the whole domain, equivalent to the mean error variance $\bar{\epsilon}_{\sigma^2}$, can be estimated by

$$\bar{\epsilon} = 1 - \sum_{i=1}^T \lambda_i \int_{\mathcal{D}} \phi_i^2(\mathbf{z}) d\mathbf{z}. \quad (7)$$

In case of an analytical solution, the eigenfunctions are orthonormal, $\phi_i(\mathbf{z})\phi_j(\mathbf{z}) = \delta_{ij}$, and Equation (7) simplifies to [2]

$$\bar{\epsilon} = 1 - \frac{1}{|\mathcal{D}|} \sum_{i=1}^T \lambda_i. \quad (8)$$

Random fields are classically categorised with aleatory uncertainties, meaning that they describe the intrinsic randomness of a phenomena itself, which cannot be further reduced. However, the describing parameters may contain epistemic uncertainty caused by a lack of knowledge or data. For this reason, the concept of imprecise random fields is introduced in the following subsection.

2.2 Probability box approach

If one or several parameters cannot be determined precisely, the classically aleatory random field includes also epistemic uncertainties, e.g. by interval or fuzzy valued parameters [10]. To avoid further assumptions on the fuzziness, this work considers epistemic parameters to be interval valued. This leads to the concept of an imprecise random field [6],

$$[X](\mathbf{z}, \omega) = \mu_X^I(\mathbf{z}) + \sigma_X^I \sum_{i=1}^{\infty} \sqrt{\lambda_i^I} \phi_i^I(\mathbf{z}) \xi_i(\omega), \quad (9)$$

where the index I denotes the interval valued random field parameters. The interval valued eigenvalues and -functions originate from an interval valued correlation length L^I .

In this work imprecise random fields are propagated through an FE problem using the probability box (p-box) approach [1]. By this, the quantity of interest Y can be described by a left and right bound $[\bar{F}_Y, \underline{F}_Y]$ of the cumulative distribution function (CDF). If further information is available, e.g. the interval ranges μ_Y^I and σ_Y^I of the mean value and the standard deviation or the probability family \mathcal{F} , these can be added and the p-box is described by the quintuple $(\bar{F}_Y, \underline{F}_Y, \mu_Y^I, \sigma_Y^I, \mathcal{F})$.

The eigenvalues λ_i are not monotonically dependent on the correlation length L . For this reason, a pure vertex analysis does not necessarily guarantee to gain the outer bounds of the p-box in case an imprecise random field contains an interval valued correlation length. A straightforward possibility is to discretise L^I , to perform a stochastic analysis with each $L_i \in L^I$ and to determine the p-box bounds by the minimum and maximum of all results [3]. Alternatively, if the used model is monotonic, the relevant intermediate correlation lengths $L_i^* \in L^I$ of the imprecise random field input can be determined by optimisation a priori [6]. This may reduce the computational effort of propagating multiple random fields, especially when several imprecise random fields are involved.

Note that when the constant standard deviation is considered to be interval valued (as well), the choice to decompose the autocorrelation function instead of the autocovariance function - as it is usually done in literature - becomes beneficial in terms of the computational cost. As $\Gamma_X(\mathbf{z}_1, \mathbf{z}_2)$ is independent of σ_X , Equation (2) needs to be solved only once (per L_i) and not for each $\sigma_{X,i} \in \sigma_X^I$ (in combination with each L_i).

3 INVESTIGATION ON CRUCIAL RANDOM FIELD PARAMETERS

Within this contribution one-dimensional (1D) random fields depending on the spatial parameter $z \in \mathcal{D}$ are investigated. Furthermore, an exponential autocorrelation function $\Gamma_X(z_1, z_2)$ is considered,

$$\Gamma_X(z_1, z_2) = \exp \left\{ -\frac{|z_1 - z_2|}{L} \right\}, \quad (10)$$

which describes the decay of the autocorrelation in terms of the correlation length L . Note that the stochastic dimension N of the random field depends on the truncation order T , see Equation (4). Due to the non-differentiability of Equation (10) at $z_1 = z_2$, the corresponding random field can become very high-dimensional when L is small compared to the domain length l . However, the availability of an analytical solution, described e.g. in [11], justifies the effort for the purpose of this paper.

The parameters which describe a random field are investigated in this section. Equation (1) can be interpreted as an expanded standard normal distributed random field ($\mu_S = 0, \sigma_S = 1$) that is scaled by σ_X and shifted towards $\mu_X(z)$. The mean value and the standard deviation therefore do not influence the expansion itself. Beside the chosen autocorrelation function, the main effect on the random field is caused by the correlation length. Therefore, in this section a 1D standard normal distributed random field $S(z, \omega)$ defined on $\mathcal{D} = [0, 1]$ is investigated for the correlation length values $L = [0.01, 0.1, 1.0, 10.0]$ in Subsection 3.2. The results are compared to the limits $L \rightarrow 0$, which defines white noise, and $L \rightarrow \infty$, which is equal to a constant standard normal distributed random variable $S(\omega)$. However, the impact of the truncation order T on the expansion error needs to be studied before.

3.1 Influence of the truncation order

When different correlation lengths $L_i \in L^I$ are considered for an imprecise random field, special care must be taken regarding the truncation order T . To ensure that the p-box bounds are not affected by different errors within the input variance $\sigma^2\{\hat{X}(z, \omega)\} = \sigma_X^2 \hat{\Gamma}_X(z_1, z_2)$, the truncation needs to be chosen individually for each L_i , e.g. in terms of an equal mean error $\bar{\epsilon}_i$. As it can be seen in Figure 1a, the convergence of the latter and consequently the stochastic dimension $N = T$ is highly dependent on the correlation length ratio. For large L/l the corresponding $\bar{\epsilon}$ is already small for very few truncation terms. It is therefore practical to use the upper bound of L^I to decide on $\bar{\epsilon}$. However, by this approach one can be forced to accept extremely high dimensions for the lower bound of L^I when the range of the interval valued correlation length is large.

Table 1: Truncation orders T resulting to the mean errors $\bar{\epsilon} \approx 3.2\%$, $\bar{\epsilon} \approx 1.3\%$ and $\bar{\epsilon} \approx 0.8\%$ regarding an exponential autocorrelation function with different correlation length ratios L/l .

$L/l [-]$	$\bar{\epsilon} \approx 3.2\%$		$\bar{\epsilon} \approx 1.3\%$		$\bar{\epsilon} \approx 0.8\%$	
	$T [-]$	$\bar{\epsilon} [\%]$	$T [-]$	$\bar{\epsilon} [\%]$	$T [-]$	$\bar{\epsilon} [\%]$
10.0	1	3.2441	2	1.2977	3	0.7963
1.0	7	3.0997	16	1.3064	26	0.7948
0.1	63	3.2382	156	1.3029	254	0.7993
0.01	625	3.2420	1551	1.3068	2534	0.7998

In Table 1, for three possibly aimed errors $\bar{\epsilon} \approx 3.2\%$, $\bar{\epsilon} \approx 1.3\%$ and $\bar{\epsilon} \approx 0.8\%$ (guided by the lowest possible truncation terms $T = 1$, $T = 2$ and $T = 2$ of the maximum L/l) the closest possible error $\bar{\epsilon}_i$ and the corresponding value T_i are listed for the different considered L_i/l . As can be seen for $\bar{\epsilon} \approx 3.2\%$, not all L_i/l can match this error well. The next possible error referred to $L/l = 10$ is $\bar{\epsilon} \approx 1.3\%$. Here the errors of the different L_i/l are better comparable already but T has more than doubled. Furthermore, if further relatively large correlation length ratios, e.g. $L/l = 5$, were included, the corresponding error could again not match perfectly. For this reason, the truncation has to be investigated for each problem considered with imprecise random fields and the comparable error needs to be chosen individually depending on the involved values L_i to be propagated. Regarding the local error $\epsilon(z)$ depicted in Figure 1b, another difficulty becomes clear in terms of different correlation lengths to be considered. Although all L_i/l fulfil the same mean error $\epsilon(z) \approx 1.3\%$, the local error still varies significantly when T is small. Depending on the quantity of interest, this can lead to localisation effects [4].

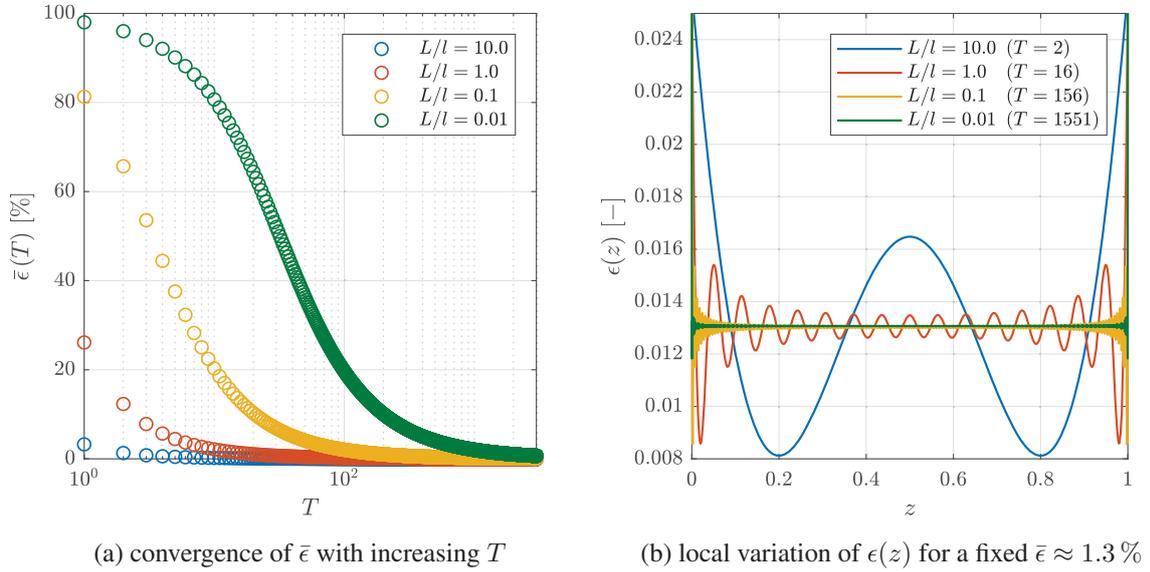


Figure 1: Influence of different correlation length ratios L/l and truncation orders T on the mean error $\bar{\epsilon}$ and the local error $\epsilon(z)$ of a 1D random field using an exponential autocorrelation function.

3.2 Influence of the correlation length

As already mentioned, the correlation length L is a parameter indicating how fast or slow the autocorrelation between the random field values $X(z_1, \omega)$ and $X(z_2, \omega)$ decays with the distance $|z_1 - z_2|$. When $L/l \rightarrow 0$, the values of the field are completely uncorrelated which is called white noise. On the other hand the random field converges towards a random variable for $L/l \rightarrow \infty$. In this subsection, the influence of L_i/l on a standard normal distributed random field $S(z, \omega)$ is studied and the convergence of the random field properties towards these two limits are investigated. The corresponding truncation orders T_i are chosen according to a mean error $\bar{\epsilon}_i \approx 0.8\%$, compare Table 1.

In Figure 2 the effect of the different considered correlation length ratios as well as the limits of L/l is visualised. On the left side, the autocorrelation function $\Gamma(z_1, z_2)$ is depicted in its closed form. It can be easily seen that the non-differentiability at $z_1 = z_2$ becomes more and more crucial for decreasing L/l . On the right, three standard normal distributed random field

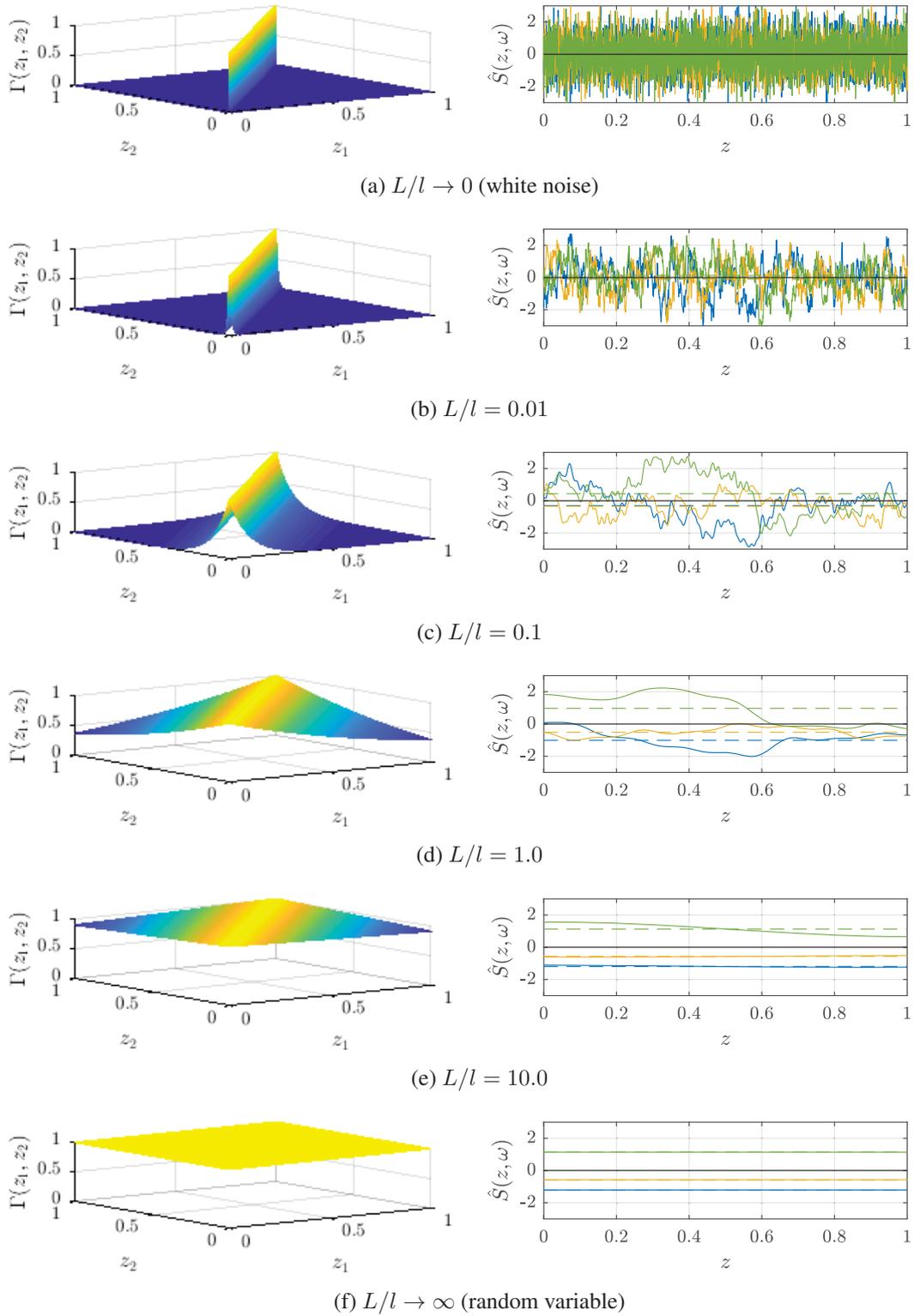


Figure 2: Influence of the correlation length ratio L/l considering a 1D standard normal distributed random field. Left: closed form of the exponential autocorrelation function $\Gamma(z_1, z_2)$, right: three random field realisations $\hat{S}_j = \hat{S}(z, \omega_j)$ (solid lines) and their corresponding mean values $\hat{\mu}_{S,j} = \mu\{\hat{S}_j\}$ (dashed lines).

realisations $\hat{S}_j = \hat{S}(z, \omega_j)$ are depicted for the different ratios L/l . Furthermore, the individual mean value $\hat{\mu}_{S,j} = \mu\{\hat{S}_j\}$ corresponding to the realisation j is given in a dashed line of the same colour. It can be seen that the variation of the random field increases with a decreasing L/l . However, the higher the variation is, the more likely $\hat{\mu}_{S,j}$ falls close to the input mean value $\mu_S = 0$ of the random field. Therefore, the convergence behaviour of $\hat{\mu}_{S,j}$ is investigated further in terms of the number n_{MC} of random field realisations.

The variation of the mean value $\hat{\mu}_{S,j}$ of an individual random field $\hat{S}_j = \hat{S}(z, \omega_j)$ seems to depend on the correlation length ratio L/l . In the following the mean value $\mu\{\hat{\mu}_{S,j}\}$ and standard deviation $\sigma\{\hat{\mu}_{S,j}\}$ of the individual random field mean values $\hat{\mu}_{S,j}$, $j = 1, \dots, n_{MC}$, are discussed in terms of an increasing sample size n_{MC} . As can be seen in Figure 3a, considering a sufficiently large sample size the mean value $\mu\{\hat{\mu}_{S,j}\}$ of all random field mean values converges towards the input mean value $\mu_S = 0$ that has been used in Equation (4) to create the realisations. On the contrary, the standard deviation $\sigma\{\hat{\mu}_{S,j}\}$ of all random field mean values is not generally converging towards the input standard deviation $\sigma_S = 1$ but towards individual values $\sigma \in [0, 1]$. Furthermore, $L/l \rightarrow 0$ describes the lower bound with $\sigma\{\hat{\mu}_{S,j}\}$ converging to zero, $L/l \rightarrow \infty$ the upper bound with $\sigma\{\hat{\mu}_{S,j}\}$ converging to one.

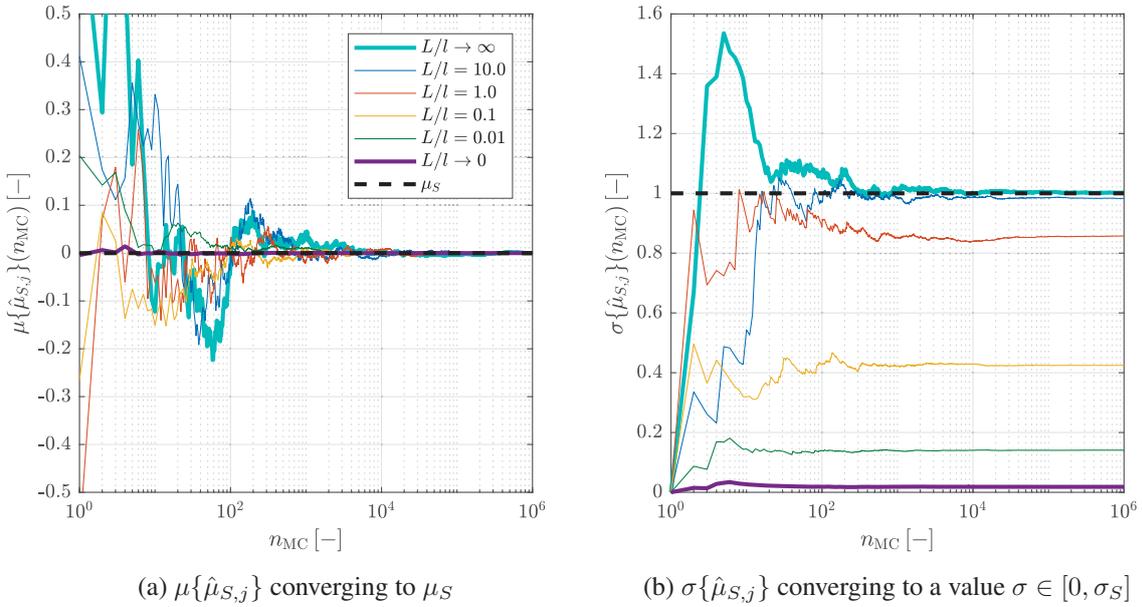


Figure 3: Convergence of the mean value $\mu\{\hat{\mu}_{S,j}\}$ and the standard deviation $\sigma\{\hat{\mu}_{S,j}\}$ of the individual mean values $\hat{\mu}_{S,j} = \mu\{\hat{S}_j\}$ of n_{MC} standard normal distributed random field realisations $\hat{S}_j = \hat{S}(z, \omega_j)$.

In this case, a standard normal distributed random field $\hat{S}(x, \omega)$ has been used. It has been found that $\mu\{\hat{\mu}_{S,j}\} \rightarrow \mu_S = 0$ and $\sigma\{\hat{\mu}_{S,j}\} \rightarrow \sigma \in [0, \sigma_S = 1]$. However, understanding an arbitrary random field $X(z, \omega)$ as a standard normal distributed random field $S(z, \omega)$ that has been scaled by σ_X and shifted towards $\mu_X(z)$, one can conclude in general that for $j = 1, \dots, n_{MC}$ and n_{MC} sufficiently large

$$\mu\{\hat{\mu}_{X,j}\} \rightarrow \mu_X \quad \text{independent of } L/l, \quad (11)$$

and $\sigma\{\hat{\mu}_{X,j}\}$ is bounded by the limits of L/l ,

$$\sigma\{\hat{\mu}_{X,j}\} \rightarrow \begin{cases} 0 & \text{for } L/l \rightarrow 0 \\ \sigma_X & \text{for } L/l \rightarrow \infty \end{cases}. \quad (12)$$

For this reason it can be worth the effort to once determine $\sigma\{\hat{\mu}_{S,j}\}$ of a standard normal distributed random variable as a function of the correlation length ratio L/l . Having this standardised result for a given autocorrelation function one can estimate $\sigma\{\hat{\mu}_{X,j}\}$ of any random field $X(z, \omega)$ by

$$\sigma\{\hat{\mu}_{X,j}\}(L/l) = \sigma_X \cdot \sigma\{\hat{\mu}_{S,j}\}(L/l), \quad (13)$$

where σ_X is the input standard deviation that is used to create random field realisations in Equation (4). Then, if the random field propagates linearly through the applied model, the mean value and standard deviation of a quantity of interest can be estimated immediately for any further L_i/l as soon as two correlation length values have been propagated.

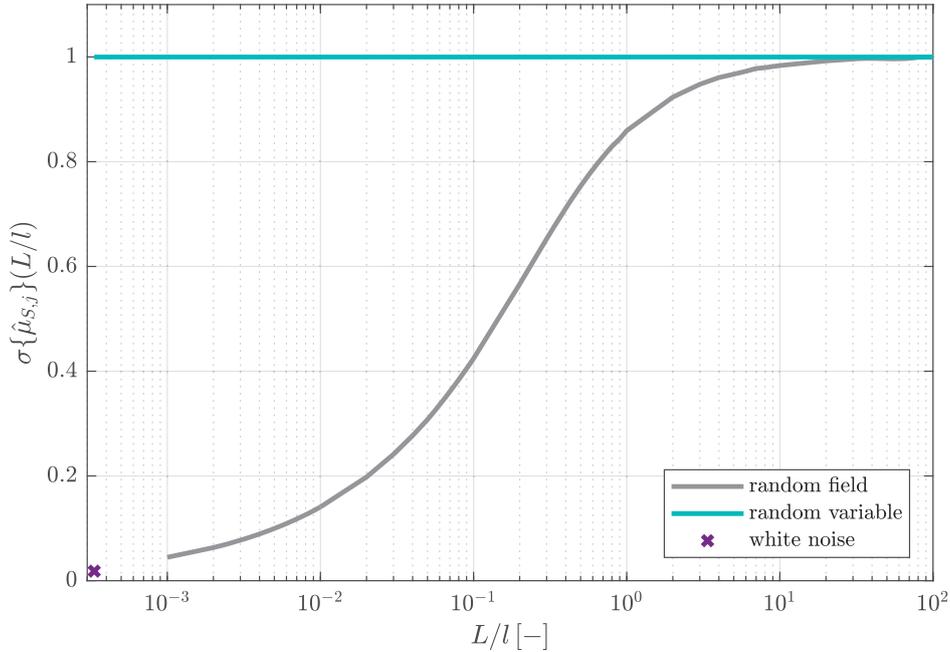


Figure 4: Standard deviation $\sigma\{\hat{\mu}_{S,j}\}$ of the individual mean values $\hat{\mu}_{S,j} = \mu\{\hat{S}_j\}$ of $n_{\text{MC}} = 10^6$ standard normal distributed random field realisations $\hat{S}_j(z, \omega)$ with respect to the correlation length ratio L/l .

In Figure 4 the standard deviation $\sigma\{\hat{\mu}_{S,j}\}$ resulting from $n_{\text{MC}} = 10^6$ individual random field mean values $\hat{\mu}_{S,j} = \mu\{\hat{S}_j\}$ is depicted as a function of L/l . Note that the results of a white noise property depend on the discretisation of the domain, here $n_{\text{el}} = 3000$, and therefore $\sigma\{\hat{\mu}_{S,j}\}$ is not exactly zero. However, it can be expected that for $n_{\text{el}} \rightarrow \infty$ it is $\sigma\{\hat{\mu}_{S,j}\} \rightarrow 0$. It can be estimated that for correlation lengths L which are larger than ten times the domain length l , the resulting random field starts to converge towards a random variable and the effort of discretising the random field by KL expansion might not be justified. On the lower bound, L/l is rather restricted by the feasibility in terms of the stochastic dimension than by converging towards white noise.

4 LINEAR BEAM STUDY

Considering a linear model $Y = \mathcal{M}(X)$ to propagate (imprecise) random fields $X(z, \omega)$, it can be assumed that the mean value $\mu\{Y\}$ and the standard deviation $\sigma\{Y\}$ depend linearly on $\mu\{\hat{\mu}_{X,j}\}$ and $\sigma\{\hat{\mu}_{X,j}\}$. Instead of simulating several $L_i \in L^I$ to determine the p-box of Y , the computational cost could therefore be reduced significantly by just propagating the limits of L , meaning white noise and a random variable, through the model to gain the p-box of “absolutely no idea”. Any further needed L_i/l could then be simply gained by linear interpolation. The standard deviation $\sigma\{\hat{\mu}_{X,j}\}$ corresponding to a specific random field $X(z, \omega)$ with any standard deviation σ_X defined on an arbitrary \mathcal{D} can be gained by sampling, which is much cheaper when the random field does not need to be propagated. Alternatively, it even can be estimated by the standardised results depicted in Figure 4 by reading $\sigma\{\hat{\mu}_{S,j}\}$ corresponding to L/l from the graph and multiplying this value with σ_X as given in Equation (13).

This assumption is further studied within this section. For this purpose, the beam problem depicted in Figure 5 is considered assuming a linear elastic material. To gain a good representation of white noise, the beam is discretised by $n_{\text{el}} = 3000$ 1D beam elements. The investigated random fields are assumed as a function along the beam length $l = 1$ m and to be constant within the cross section $A = 0.01$ m² (1D random field). The maximum deflection $w_{\text{max}} = w(z = 0.5$ m) in the middle of the both-sided supported beam is the quantity of interest. For a deterministic simulation using a Young’s modulus $E = 210 \cdot 10^9 \frac{\text{N}}{\text{m}^2}$ and a constant line load $q_0 = 5000 \frac{\text{N}}{\text{m}}$, the quantity of interest results in $w_{\text{max}} = 3.72 \cdot 10^{-5}$ m.

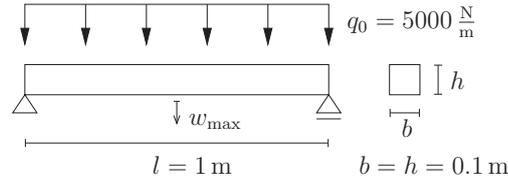


Figure 5: Linear-elastic beam with Young’s modulus $E = 210 \cdot 10^9 \frac{\text{N}}{\text{m}^2}$ under constant line load, deterministic maximal deflection: $w_{\text{max}} = 3.72 \cdot 10^{-5}$ m.

In the following subsections, both, the line load and the Young’s modulus are considered as imprecise random fields in first two studies independently, and in a third study combined.

4.1 Investigation on different imprecise random field input parameters

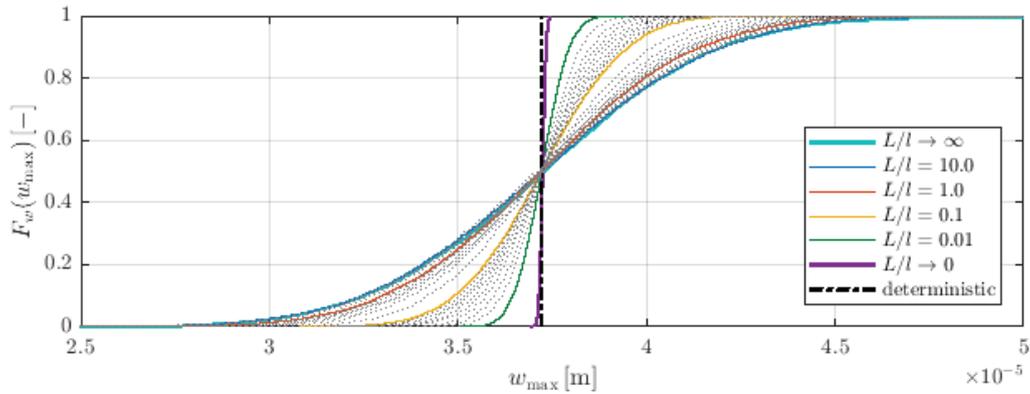
In the following, two studies on the correlation length ratio L/l are performed, each with one parameter considered as imprecise random field input with the parameters given in Table 2. For both studies, the discretised correlation length intervals $L^{(1)}/l = [0.01 : 0.01 : 0.1]$, $L^{(2)}/l = [0.1 : 0.1 : 1.0]$ and $L^{(3)}/l = [1.0 : 1.0 : 10.0]$ as well as $L/l \rightarrow 0$ (white noise, abbreviated by WN) and $L/l \rightarrow \infty$ (random variable, abbreviated by RV) are investigated choosing $\bar{\epsilon} = 0.8$ %. A closer look is spend on the results of the values $L^{(*)}/l = [\text{WN}, 0.01, 0.1, 1.0, 10.0, \text{RV}]$. Each L_i/l simulation is performed using brute force MC with $n_{\text{MC}} = 10000$ samples.

In the first study, the line load q is modelled as an imprecise random field with $\mu_q = q_0 = 5000 \frac{\text{N}}{\text{m}}$ and $\sigma_q = 0.1\mu_q = 500 \frac{\text{N}}{\text{m}}$. As q is in the numerator of the beam deflection solution, it can be assumed that the propagation of L_i/l through the FE model is linear. The Young’s modulus E , considered with $\mu_E = E = 210 \cdot 10^9 \frac{\text{N}}{\text{m}^2}$ and $\sigma_E = 0.05\mu_E = 10.5 \cdot 10^9 \frac{\text{N}}{\text{m}^2}$ within the second study, can be found in the denominator of the beam deflection solution. Therefore, the results are expected to depend inversely on $E(z, \omega)$ and might not be Gaussian distributed anymore.

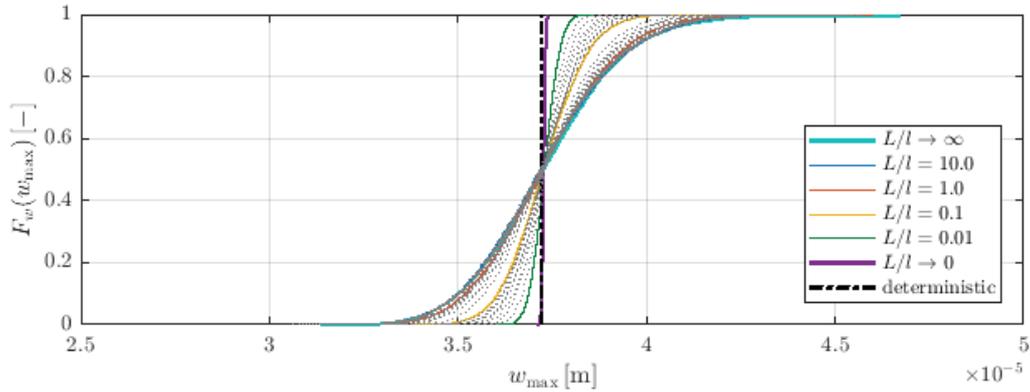
Table 2: Random field parameters considered for two studies including each one random field input parameter.

	random field input	mean value	standard deviation
study 1	line load $q(z, \omega)$	$\mu_q = 5000 \frac{\text{N}}{\text{m}}$	$\sigma_q = 0.1\mu_q = 500 \frac{\text{N}}{\text{m}}$
study 2	Young's modulus $E(z, \omega)$	$\mu_E = 210 \cdot 10^9 \frac{\text{N}}{\text{m}^2}$	$\sigma_E = 0.05\mu_E = 10.5 \cdot 10^9 \frac{\text{N}}{\text{m}^2}$

The CDFs resulting from each L_i/l are depicted in Figure 6 for both studies. The axis describing w_{\max} is scaled uniquely such that both p-boxes can be directly compared qualitatively. The deterministic result is given as a vertical dash-dot line, the results of $L^{(*)}/l$ in different colours and these of all other L_i/l as grey dotted lines. Both results appear to be Gaussian distributed, which corroborates the assumption of a linear dependency between input and output.



(a) study 1: line load as input random field



(b) study 2: Young's modulus as input random field

Figure 6: CDFs for different correlation length ratios L/l bounded by white noise $L/l \rightarrow 0$ and a random variable $L/l \rightarrow \infty$ considering one random field input parameter.

In the first study, the input standard deviation is assumed to be ten percent of the input mean value, $\sigma_q = 0.1\mu_q$, while it is only five percent, $\sigma_E = 0.05\mu_E$, in the second study. For this reason, the resulting p-box is much wider in Figure 6a than in Figure 6b. The CDF gained by a random variable spans the widest range of w_{\max} . With decreasing L/l the CDFs become steeper. The CDF of the white noise is very close to the deterministic value. It can be assumed that it will converge towards a vertical line for $n_{el} \rightarrow \infty$. By this, one can estimate an “absolutely no idea

p-box” by even just one stochastic simulation, the one assuming just a random variable. The white noise result defining the second part of the p-box bound can be considered as a vertical line $\mu\{w_{\max}^{\text{RV}}\}$. In addition to the qualitative impression gained by Figure 6, the mean value and standard deviation of both, input random fields and the quantity of interest according to each $L^{(*)}/l$ can be compared quantitatively in Table 3.

Table 3: Mean value $\mu\{\hat{\mu}_{X,j}\}$ and standard deviation $\sigma\{\hat{\mu}_{X,j}\}$ of the considered input random fields as well as mean value $\mu\{w_{\max}\}$ and standard deviation $\sigma\{w_{\max}\}$ of the maximum beam deflection w_{\max} resulting from a propagation through a linear FE model for different L/l .

(a) study 1: line load as input random field				
L/l [-]	input realisations		quantity interest	
	$\mu\{\hat{\mu}_{q,j}\} [\frac{\text{N}}{\text{m}}]$	$\sigma\{\hat{\mu}_{q,j}\} [\frac{\text{N}}{\text{m}}]$	$\mu\{w_{\max}\} [\text{m}]$	$\sigma\{w_{\max}\} [\text{m}]$
WN	5.0001e+03	9.1424e+00	3.7223e-05	7.5621e-08
0.01	4.9986e+03	7.0213e+01	3.7213e-05	5.8664e-07
0.1	4.9998e+03	2.1207e+02	3.7214e-05	1.7917e-06
1.0	5.0002e+03	4.2623e+02	3.7221e-05	3.2960e-06
10.0	5.0013e+03	4.9965e+02	3.7231e-05	3.7368e-06
RV	5.0039e+03	5.0130e+02	3.7251e-05	3.7319e-06
deterministic	5e+03	-	3.72e-05	-

(b) study 2: Young’s modulus as input random field				
L/l [-]	input realisations		quantity interest	
	$\mu\{\hat{\mu}_{E,j}\} [\frac{\text{N}}{\text{m}^2}]$	$\sigma\{\hat{\mu}_{E,j}\} [\frac{\text{N}}{\text{m}^2}]$	$\mu\{w_{\max}\} [\text{m}]$	$\sigma\{w_{\max}\} [\text{m}]$
WN	2.1000e+11	1.9224e+08	3.7249e-05	4.3169e-08
0.01	2.0998e+11	1.4680e+09	3.7299e-05	3.3463e-07
0.1	2.1001e+11	4.4876e+09	3.7293e-05	9.9669e-07
1.0	2.0994e+11	8.9484e+09	3.7310e-05	1.6989e-06
10.0	2.0000e+11	1.0341e+10	3.7295e-05	1.8649e-06
RV	2.0990e+11	1.0457e+10	3.7313e-05	1.8716e-06
deterministic	2.1e+11	-	3.72e-05	-

As it was expected based on the behaviour of random fields investigated in Subsection 3.2, the mean values $\mu\{w_{\max}\}$ of the maximum beam deflection turn out to lay close to the deterministic result, independent of the correlation length. The standard deviation $\sigma\{\hat{\mu}_{X,j}\}$ of the input random fields have been determined by the generated samples. Alternatively, $\sigma\{\hat{\mu}_{X,j}\}$ can be determined by Equation (13). The values of both options are compared in Table 4 for both considered random field inputs, the line load $q(z, \omega)$ and the Young’s modulus $E(z, \omega)$. Note that $\sigma\{\hat{\mu}_{S,j}\}$ has been determined by $n_{\text{MC}} = 10^6$ samples, while the $q(z, \omega)$ and $E(z, \omega)$ have been sampled only $n_{\text{MC}} = 10^4$ times each. Still, the results are comparable already. Furthermore, it can be seen that Equation (12) holds true for both input parameters E and q .

Regarding the standard deviation $\sigma\{w_{\max}\}$ of the quantity of interest, it can be seen that the value is very small for white noise and increases with increasing L/l for both studies in Ta-

Table 4: Comparison of the standard deviation $\sigma\{\hat{\mu}_{X,j}\}$ of the input random fields determined by $n_{MC} = 10000$ samples with the result gained by factorising $\sigma\{\hat{\mu}_{S,j}\}$ of a standard normal distributed random field $\hat{S}(z, \omega)$ by σ_X .

L/l [-]	$\sigma\{\hat{\mu}_{S,j}\}$ [-]	line load $q(z, \omega)$ [$\frac{N}{m}$]		Young's modulus $E(z, \omega)$ [$\frac{N}{m^2}$]	
		$\sigma_q \cdot \sigma\{\hat{\mu}_{S,j}\}$	$\sigma\{\hat{\mu}_{q,j}\}$	$\sigma_E \cdot \sigma\{\hat{\mu}_{S,j}\}$	$\sigma\{\hat{\mu}_{E,j}\}$
WN	0.0183	9.1500e+00	9.1424e+00	1.9215e+08	1.9224e+08
0.01	0.1408	7.0400e+01	7.0213e+01	1.4784e+09	1.4680e+09
0.1	0.4243	2.1212e+02	2.1207e+02	4.4552e+09	4.4876e+09
1.0	0.8593	4.2965e+02	4.2623e+02	9.0227e+09	8.9484e+09
10.0	0.9836	4.9180e+02	4.9965e+02	1.0328e+10	1.0341e+10
RV	0.9999	4.9995e+02	5.0130e+02	1.0499e+10	1.0457e+10

bles 3a and 3b. To investigate a possible linear dependence between input and output, $\sigma\{w_{\max}\}$ is plotted versus $\sigma\{\hat{\mu}_{X,j}\}$ in Figure 7. The axis denoting $\sigma\{w_{\max}\}$ is scaled equally for both studies. This way, the different percentages in the input standard deviations, $\sigma_q = 0.1\mu_q$ and $\sigma_E = 0.05\mu_E$ become visible again. The values $L^{(*)}/l$ are highlighted in red while all other L_i/l pairs are depicted as grey crosses. The blue line represents the assumed linear dependence between white noise, for which it is $\sigma\{w_{\max}\} \rightarrow 0$ for $n_{el} \rightarrow \infty$, and the random variable. It can be seen that the dependence between input and output standard deviation is not perfectly linear, as the results lay above the blue line. Assuming a linear dependence and interpolating any L/l response from the CDF gained by a random variable would therefore underestimate the real standard deviation.

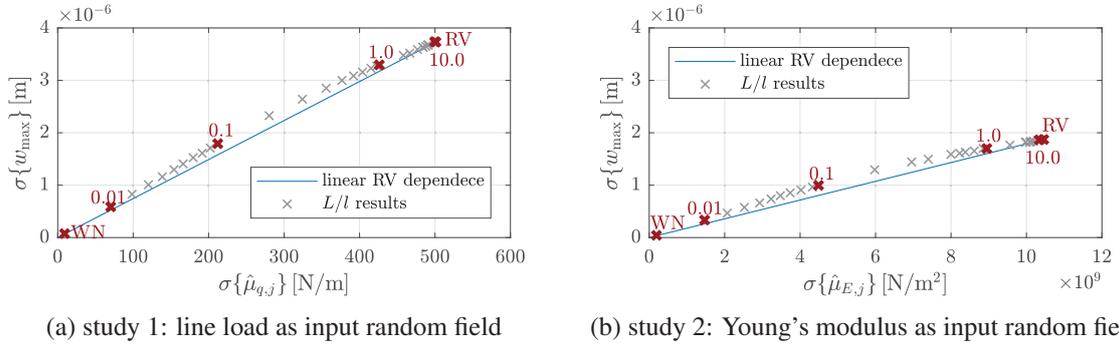


Figure 7: Dependence of the output standard deviation $\sigma\{w_{\max}\}$ on the input standard deviation $\sigma\{\hat{\mu}_{X,j}\}$ of the individual mean values $\hat{\mu}_{X,j} = \mu\{\hat{X}_j\}$ with $n_{MC} = 10000$ random field realisations $\hat{X}_j(z, \omega)$ when one input random field is considered.

The CDFs resulting from a linear interpolation of L_i/l within the “absolutely no idea p-box” are compared to the ones gained by sampling and propagation in Figures 8 and 9 for both studies. The underestimated standard deviation is clearly visible by the dashed lines, which denote the interpolated CDFs, being slightly steeper than the corresponding CDF gained by sampling (solid line). However, with respect to the spectrum resulting from “having no idea at all” about the correlation length, the linear interpolation leads to a good estimate. Furthermore, if it is supposed to represent the lower bound of L , the estimate returns a slightly more conservative but save bound.

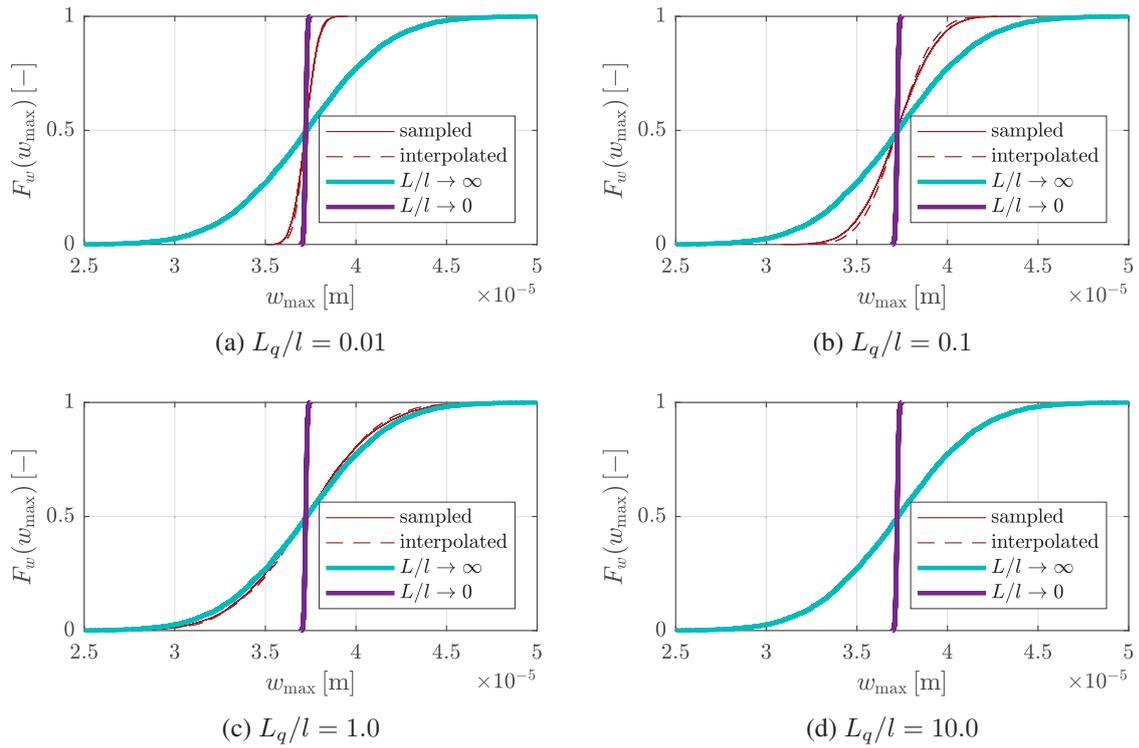


Figure 8: Comparison of the CDFs gained by sampling and interpolation within the “absolutely no idea p-box” for different correlation lengths L_q/l considering the line load as an imprecise random field input.

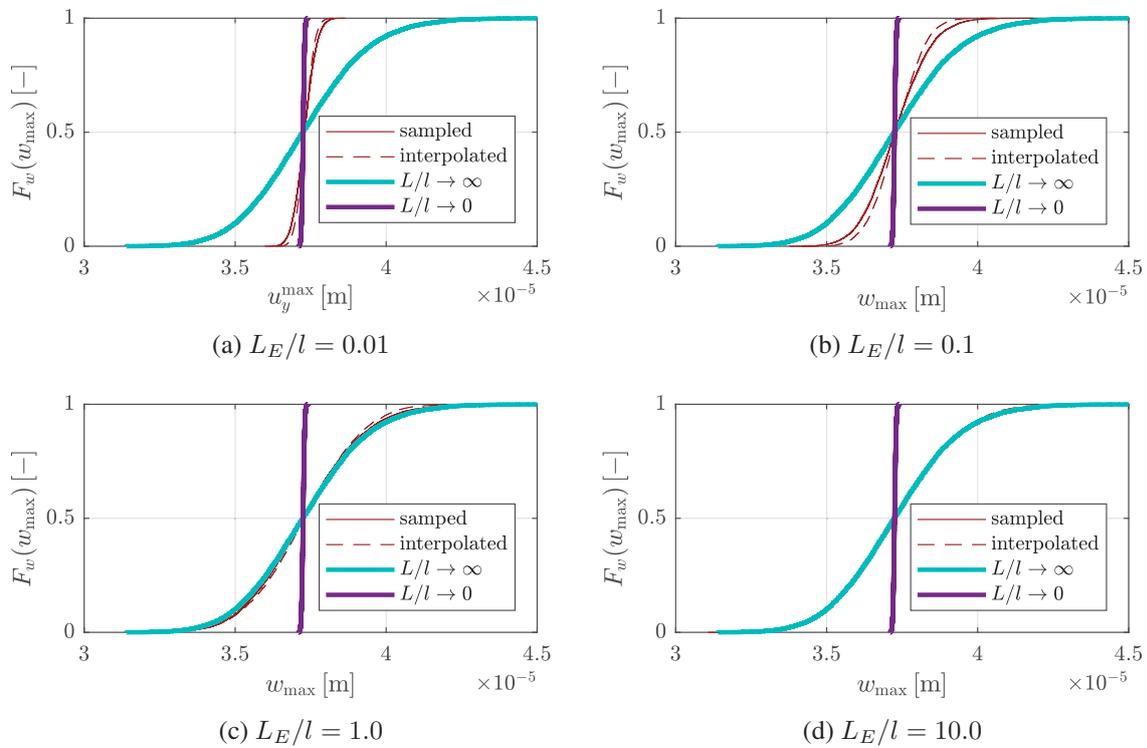


Figure 9: Comparison of the CDFs gained by sampling and interpolation within the “absolutely no idea p-box” for different correlation lengths L_E/l considering the Young’s modulus as an imprecise random field input.

4.2 Investigation on the interference of two imprecise random field input parameters

The more parameters are considered as imprecise random fields the more expensive a simulation becomes. If the underlying interval valued correlation lengths need to be discretised, each combination of L_i/l corresponding to each parameter needs to be propagated. In this case, a cheap estimate gained by linear interpolation can still become valuable. The beam problem defined in Figure 5 is simulated again but with both parameters, the line load $q(z, \omega)$ and the Young's modulus $E(z, \omega)$, considered as imprecise random fields. The corresponding mean values μ_q and μ_E as well as standard deviations σ_q and σ_E are still chosen as given in Table 2. For both parameters, the correlation length values $L^{(*)}/l = [\text{WN}, 0.01, 0.1, 1.0, 10.0, \text{RV}]$ are chosen and each combination $L_{q,i}/l \times L_{E,i}/l$ is propagated. The corresponding random fields are truncated such that $\bar{\epsilon}_i = 0.8\%$. For the propagation of each L_i/l combination, $n_{\text{MC}} = 30000$ samples are generated.

The resulting CDFs for the quantity of interest w_{max} are depicted in Figure 10. The combination of twice white noise and twice a random variable are depicted in bold lines while the vice versa combinations are depicted in bold dashed lines. For the sake of clarity, the combinations where it is $L_{q,i}/l = L_{E,i}/l$ are depicted in coloured lines, all other combinations are plotted in grey dotted lines.

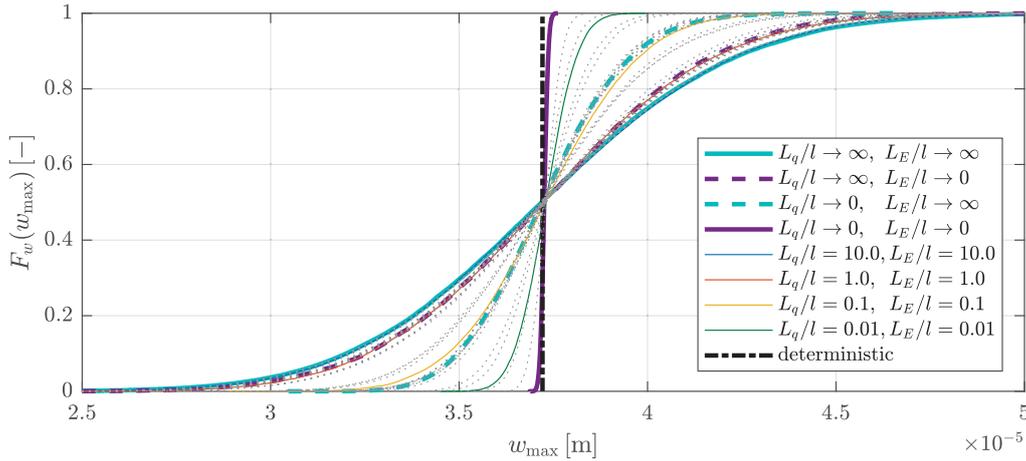


Figure 10: CDFs for different combinations of correlation length ratios L_q/l and L_E/l bounded by the combinations of white noise $L_q/l \rightarrow 0, L_E/l \rightarrow 0$ and random variables $L_q/l \rightarrow \infty, L_E/l \rightarrow \infty$ considering two random field input parameters.

As before, the mean value $\mu\{w_{\text{max}}\}$ is not affected but the standard deviation $\sigma\{w_{\text{max}}\}$. It can be seen that all L_i/l combinations lay within the p-box defined by the white noise combination and the random field combination. Furthermore, the former seems again to converge towards the deterministic result. The “absolutely no idea p-box” can therefore be defined by only propagating the combination of both parameters being a random variable.

The resulting standard deviation $\sigma\{w_{\text{max}}\}$ depending on the input combination of $\sigma\{\hat{\mu}_{q,j}\}$ and $\sigma\{\hat{\mu}_{E,j}\}$ is depicted in Figure 11. The blue surface is spanned by the results corresponding to $[0, \sigma\{w_{\text{max}}^{\text{RV},q}\}] \times [0, \sigma\{w_{\text{max}}^{\text{RV},E}\}]$. The results $\sigma\{w_{\text{max}}\}$ gained by propagating the pairs $L_{q,i}/l, L_{E,i}/l$ are marked by a cross, while the interpolated value corresponding to this input is marked by a dot on the surface. Furthermore, the interpolated and simulated values corresponding to each other are connected by a line. This way the distance between the simulation cross and the interpolation surface is visualised.

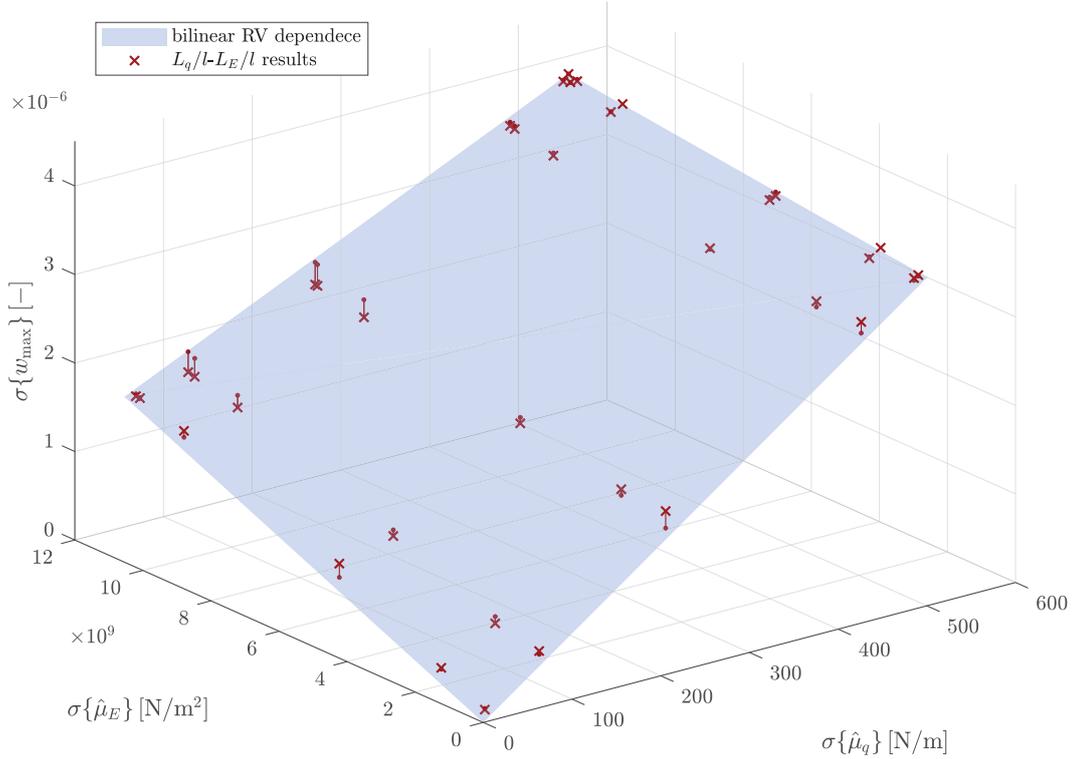


Figure 11: Dependence of the output standard deviation $\sigma\{w_{\max}\}$ on the input standard deviations $\sigma\{\hat{\mu}_{q,j}\}$ and $\sigma\{\hat{\mu}_{E,j}\}$ of the individual mean values $\hat{\mu}_{q,j} = \mu\{\hat{q}_j\}$ and $\hat{\mu}_{E,j} = \mu\{\hat{E}_j\}$, respectively, with $n_{\text{MC}} = 30000$ random field realisations $\hat{q}_j(z, \omega)$ and $\hat{E}_j(z, \omega)$ when two imprecise random fields are considered.

Compared to the two studies discussed in Subsection 4.1, the bilinear interpolation surface matches most of the simulation results even better. The computational cost could therefore be reduced drastically by only propagating the parameters modelled by random variables and avoiding the propagation of several correlation length combinations. Additionally, this would completely save the cost to determine the KL expansion, which can become expensive when no analytic solution is available. The cost of each individual realisation can be furthermore reduced significantly when the random property is constant for each realisation. Finally, propagating one or several random variables means a low stochastic dimension which enables more sophisticated sampling techniques than brute force MC sampling. As discussed in Subsection 3.1, the propagation of just one random field can become highly dimensional. Therefore sophisticated sampling methods often suffer from the curse of dimensionality when they are used to propagate random fields.

5 CONCLUSION AND PERSPECTIVES

In this contribution imprecise random fields described by interval valued correlation lengths have been investigated. In a first study, the influence of the correlation length L on a standard normal distributed random field has been studied in general. To describe the variability of a random field $X(z, \omega)$ corresponding to L , the mean value $\mu\{\hat{\mu}_{X,j}\}$ and standard deviation $\sigma\{\hat{\mu}_{X,j}\}$ of the individual random field realisations mean values $\hat{\mu}_{X,j}$ have been introduced. It

could be shown that the former converges towards the mean value μ_X used to define the random field, while the converged value of the latter depends on L . However, the bounds of $\sigma\{\mu_{\hat{X},j}\}$ are defined by the limits of the correlation length, $L \rightarrow 0$ describing white noise and $L \rightarrow \infty$ standing for a random variable. For the applied linear elastic example, these limits propagate to a p-box which includes all other solutions corresponding to any L . As $L \in (0, \infty)$ represents all possible correlation lengths when no information is available, this p-box has been called “absolutely no idea p-box”.

In a second step the dependence between input and output of the simple linear mechanical model has been investigated in terms of different imprecise random field input parameters. As the mean value of the quantity of interest Y is barely affected by the correlation length, the focus has been on the standard deviation. It has been shown that the dependence between the standard deviation $\sigma\{Y\}$ of the quantity of interest and $\sigma\{\mu_{\hat{X},j}\}$ is not perfectly linear. However, determining it by assuming a linear dependence within the “absolutely no idea p-box” has shown to result in a good estimate. Furthermore, as the real standard deviation $\sigma\{Y\}$ is underestimated, a linear interpolation of the lower interval bound results in a conservative p-box, when $L \in [\tilde{L}, \infty)$ is considered. According to the fact that the correlation length (as well as the autocorrelation function itself) is usually unknown, interpolating within an “absolutely no idea p-box” can be a computational cheap method in terms of engineering application. As the white noise converges towards a vertical line of the mean value $\mu\{Y\}$ corresponding to the random field, only the random variable needs to be propagated to determine this limit representation of the p-box, avoiding the need for computationally expensive random field discretisation. By that, the stochastic dimensions are reduced drastically and perhaps, more efficient low dimensional sampling schemes could be applied to further reduce the computational cost for engineering analysis.

For engineering applications the suggested approach appears very attractive. Further investigations are needed to investigate nonlinear problems. Here, for some parameters the dependence between input and output can become more complex. Still, a first linear estimate can be used to reduce the sampling effort.

REFERENCES

- [1] M. Beer, S. Ferson and V. Kreinovich, Imprecise probabilities in engineering analyses. *Mech. Syst. Signal Pr.* **37**, 4–29, 2013.
- [2] W. Betz, I. Papaioannou, D. Straub, Numerical methods for the discretization of random fields by means of the Karhunen-Loève expansion. *Comput. Methods Appl. Mech. Engrg.* **271**, 109–129, 2014
- [3] M.M. Dannert, R.M.N. Fleury, A. Fau and U. Nackenhorst, Non-linear Finite Element Analysis under Mixed Epistemic and Aleatory Uncertain Random Field Input. M. Beer and E. Zio eds. *Proceedings of the 29th European Safety and Reliability Conference (ES-REL)*, Hanover, Germany, September 22-26, 2019.
- [4] M.M. Dannert, M.G.R. Faes, R.M.N. Fleury, A. Fau, U. Nackenhorst and D. Moens, Imprecise random field analysis for non-linear concrete damage analysis. *Mech. Syst. Signal Pr.* **150**, 107343, 2021
- [5] A. Der Kiureghian and O. Ditlevsen, Aleatory or epistemic? Does it matter? *Struct. Saf.* **31** (2), 105–112, 2009

- [6] M. Faes and D. Moens, Imprecise random field analysis with parametrized kernel functions. *Mech. Syst. Signal Pr.* **134**, 106334, 2019.
- [7] R.G. Ghanem and P.D. Spanos, *Stochastic Finite Elements: A Spectral Approach*. New York: Springer, 1991.
- [8] M. Hanss, *Applied Fuzzy Arithmetic: An Introduction with Engineering Applications*, Springer, 2005
- [9] R. Moore, R. Kearfott and M. Cloud, *Introduction to Interval Analysis*, Society for Industrial and Applied Mechanics, 2009
- [10] F.N. Schietzold, A. Schmidt, M.M. Dannert, A. Fau, R.M.N. Fleury, W. Graf, M. Kaliske, C. Könke, T. Lahmer and U. Nackenhorst, Development of fuzzy probability based random fields for the numerical structural design. *GAMM-Mitteilungen* **42**, e201900004, 2019.
- [11] B. Sudret and A. Der Kiureghian, Stochastic finite element methods and reliability - A state-of-the-art report, *Tech. rep., report no. UCB/SEMM-2000/08*, Department of Civil & Environmental Engineering, University of California, Berkeley, 2000.

NUMERICAL SIMULATION OF A 3D PRINTED WALL STRUCTURE DURING THE PROCESS OF PRINTING CONSIDERING UNCERTAINTY

Meron Mengesha², Albrecht Schmidt^{1,2}, Luise Göbel¹,

Tom Lahmer^{1,2}, and Carsten Könke¹

¹ Materials Research and Testing Institute (MFPA) at the Bauhaus-Universität Weimar, Germany
e-mail: meron.wondafrash@gmail.com

² Institute of Structural Mechanics, Bauhaus-Universität Weimar, Germany

Abstract

3D concrete printing technology is getting increasing recognition in the construction industry. The extrusion-based printing method represents the most popular and promising one among the 3D printing techniques for concrete. However, mostly time-consuming trial-and-error explorations, i.e. mainly experimental studies have been performed so far.

By utilizing numerical simulation, a fundamental understanding of the relations between process - process parameters - product properties could be achieved. Also, they enable us to study the dependencies of properties of the printed product on process parameters and material behavior. The extrusion-based 3D concrete printing process can be reliably controlled and optimized by taking into account the uncertain nature of the process and material parameters.

In this study, the Finite Element (FE) method combined with a pseudo-density approach, following the soft-killing approaches in topology optimization is applied. The numerical simulations allow to reliably estimate the strength-based failure mechanisms that might occur during the 3D concrete printing of a wall structure by varying one of the printing process parameter printing velocity.

Keywords: 3D concrete printing, Uncertainty, SFEM simulation, Pseudo-density approach, Random Process, Reliability, Strength-based failure

1 INTRODUCTION

Very recently, additive manufacturing techniques for concrete technology have gained wide attention. They indicated their potential to become a serious supplement to conventional concrete casting in molds. 3D concrete printing (3DCP) is one of the fastest evolving technologies in construction engineering, which is illustrated by the rapid growth of both research and industry projects carried out worldwide [1]. The main reason for this is it directly addresses the challenges related to the sustainability and productivity of the construction industry.

However, the current practice is based on the trial-and-error procedure, which makes the research of the 3DCP process expensive and time-consuming [2].

One of the reasons is that there exist significant knowledge gaps regarding the relations between the design, material, and process parameters. The quality of the fabricated product is significantly influenced by these parameters which also exhibit interdependency [3].

Therefore, it is of vital importance to establish a relation between the process parameters and the printed product to avoid unreliability and failure [1, 4]. By implementing a numerical simulation of the 3DCP process, a more fundamental understanding of the relations between the printing process, the process parameters, and the properties of the printed product could be achieved.

Since the technology is new, the deterministic approach, i.e. applying safety factors to account for the uncertainty of the system, may not be applicable. Accordingly, the Stochastic Finite Element Method (SFEM) incorporating the spatially varying pseudo-density approach is proposed to include the uncertain nature of the process and material parameters of the extrusion-based 3D concrete printing. Also in the numerical modeling along with the progressing printing process, a previously generated finite element (FE) mesh is activated layer by layer that includes all material parameters. These vary spatially and temporarily due to the time dependency of the curing process. The numerical simulations allow to reliably estimate strength-based failure mechanisms that might occur during the 3D concrete printing of a wall structure.

2 STATE OF THE ART

To obtain a stable and reliable printed product, two criteria have to be considered and controlled during the manufacturing: the overall failure probability and the geometrical dimensions of the single layers. Non-sufficient strength, stiffness, or stability may already cause the failure of the structure during the printing process. These properties are strongly dependent on the printing process parameters, e.g. printing velocity, temperature, nozzle diameter, as well as on the concrete mixture.

In this context, Van der Putten et al. [21] studied the effects of the linear printing speed and the time gap between two subsequent layers on the microstructure of printed concrete. Accordingly, the two parameters significantly influence the surface roughness, the compressive strength, and the interlayer bonding strength. Therefore, it is of vital importance to establish a relation between the process parameters and the mechanical properties of the printed product in order to avoid unreliability and failure [1,4].

Structural reliability analysis aims at computing the probability of failure, by accounting for different sources of uncertainties. These include inherent randomness of the material or lack of data [10], geometrical imperfection, random loading [11], uncertainties due to human error, and adopted model [12]. For example, for printable concrete Wolfs et al. [4] showed that the mechanical characteristics of printable concrete are random in nature with different values of coefficient of variation minimum of 3.8% to a maximum of 23%.

Since it includes safety definition and uncertainties in the analysis the probability of failure is a more reliable and complete measure of safety [19].

To compute the failure probability, it is necessary to formulate a limit state function $g(x)$ which, for instances can include the displacements, stresses, or strains where X is a vector of basic random variables, which describe the randomness in the geometry, material properties, and loading, etc. [15,19].

$$P_f = \int_{g(x) \leq 0} f_X(x) dx \quad (1)$$

where f_X is the joint probability density function (PDF) of random vector X and $g(x)$ is the limit state function, with $g(x) \leq 0$ denoting the failure domain and $g(x) > 0$ is the safe domain. Making its direct estimation for Eq. (1) is computationally expensive in the general case, but various approximation methods have been developed to evaluate the failure probability, here the method used is the Adaptive Kriging Monte Carlo Simulation (AK-MCS), Which saves the costly evaluation of the actual limit state function [15]. It is based on Monte Carlo Simulation (MCS) and Kriging meta-model [24], by using the Kriging meta-model to approximate the limit state function, which is then combined with MCS to evaluate the probability of failure [22].

The limit state function for the extrusion-based 3D printing processes of the wall is elastic buckling and plastic collapse. The elastic buckling mechanism reflects failure caused by a loss of geometrical stability, while plastic collapse is characterized by the maximum stress reaching the material yield strength [23].

The value of the yield strength is dependent on the type of failure criterion adopted for the printing material. Two representative failure criteria are pressure-dependent shear failure following the Mohr-Coulomb theory and compressive failure described by the maximal stress theory [23].

To determine the maximum shear stress developed as a result of the self-weight, deterministic FEM is typically restricted to average values of the input variables, it fails to consider the uncertainties and leads to a rough representation of reality [10,13,14].

To account for the various uncertainties arising in the model description (geometry, material properties, or loading) encountered in engineering practice, researchers have been trying to extend the standard FEM into the Stochastic Finite Element Method (SFEM) by incorporating random variables into the mathematical and computational formulations [10, 12, 15]. It has been named the Random Finite Element Method (RFEM) and the Probabilistic Finite Element Method (PFEM) [10].

SFEM consists mainly of discretization of stochastic fields, a FEM analysis part, and estimation of system response statistics [16]. Different approaches are available for the discretization of a stochastic field, some of them are the midpoint method, the interpolation method, the local average method. Once the stochastic fields are generated, Monte Carlo simulation is the most straightforward method, since it only needs repeated execution of an existing deterministic solution by utilizing many realizations of the random variables [16,18,19].

2.1 Compressive failure

Plastic collapse is reached when the compressive strength becomes lower than the vertical compressive stress. Wolfs et al. [4], experimentally determined the evolution of the compressive strength over the time of the curing process and have developed equations based on the average result for 3D printable concrete.

$$\sigma_y (r, t) = \sigma_{y,0} + \sigma_{y,1} t_r \quad (2)$$

where $\sigma_{y,0}$ is the initial yield strength of the printable concrete at the moment it leaves the printing nozzle, $\sigma_{y,1}$ is the gradient of temporal increment, and t_r is the curing time at a specific location.

However, experimental variations were observed in a series of compression and shear tests conducted at different ages of the fresh concrete [4]. It is shown that the experimentally obtained results exhibit a large scatter with coefficients of variation ranging from 13 to 21 % for the compressive strength. Furthermore, as indicated in Eq. (2) the experiments have revealed that the studied material properties increase linearly over time. From these findings, it is obvious that reliability-oriented modeling needs to account for both temporal changes and the randomness of the material parameters.

Random fields are more frequently applied in structural engineering analyses related to concrete. A parameter considering spatial variability is mainly determined by the mean, the variance, and the scale of fluctuation [28,31]. For concrete structures in some literature, the correlation properties are usually taken into account by utilizing exponential functions [30]. Bottenbruch et al. [29], have done a numerical investigation on spatial correlation of concrete and identified that it is significant along with the layers for pouring concrete in layers by slip forming.

Nonetheless, in most literature, the correlation length is assumed and these assumptions are not consistent with each other [32], the reason for this is the lack of available data [30,31]. The probability of failure can be underestimated resulting in an unconservative design if this value is ignored or implicitly assumed to be infinite [12,16,33,35]. To avoid this the modeling needs to account for the correlation length.

The compressive stress $\sigma_p (r,t)$ depends on the height of the wall $h (r)$ to be printed. This compressive stress acting on any of the layers can be written as follows [8]:

$$\sigma_p (r) = \rho_c g h(r) \quad (3)$$

Where ρ_c is the material density, g is the acceleration of gravity. Similarly, Wolfs et al. [4] experimentally determined the density of the printable concrete.

3 NUMERICAL MODELING OF 3D CONCRETE PRINTING

During the extrusion process the rheological properties of the concrete change. The material should be flowable at the pumping and extruding stage and after deposition, it should gain rapid strength to have a stable shape and to carry the subsequent layers [5, 6].

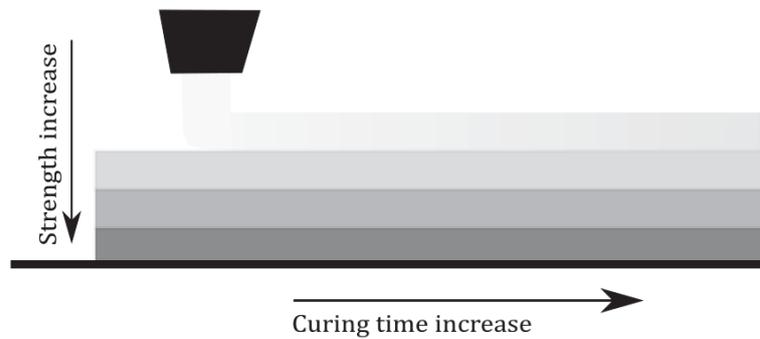


Figure 1: Gradient of the strength development of 3D printed concrete wall.

To ensure the bond strength between layers the time gap between consecutive layers should be kept as low as possible [7]. This time dependency of the evolution of the yield stress is one of the main components in numerical modeling.

Fig.1 shows the gradient of the concrete strength due to the different ages and resting times of the layers. Compared to the other layers, the concrete of the bottom layer is more mature than in the subsequent layers. To analyze the shape stability or buildability of the printed structure at a given time, this gradient of the strength over the height of the structure should be considered [8]. As each layer is activated in the numerical modeling this spatial variation is considered.

For each of the printed layer, the curing time (t_c) can be calculated based on the printing velocity (v_p), counting the number of layers from bottom to top according to the printing progress and assuming that there is no time gap between layers, the i^{th} layer will have a curing time of:

$$t_{ci} = (-i + (N + 1)) \frac{L}{v_p}, \quad (4)$$

where N is the total number of layers and L is the length of the printed layer.

The numerical model is based on the finite element method (FEM) following the layer-wise production process, i.e. each layer is activated separately. However, the problem arises when the layer thickness of the concrete and the size of the FE mesh do not coincide because the size of the FE is much lower than the thickness of the layers, see Fig.2.

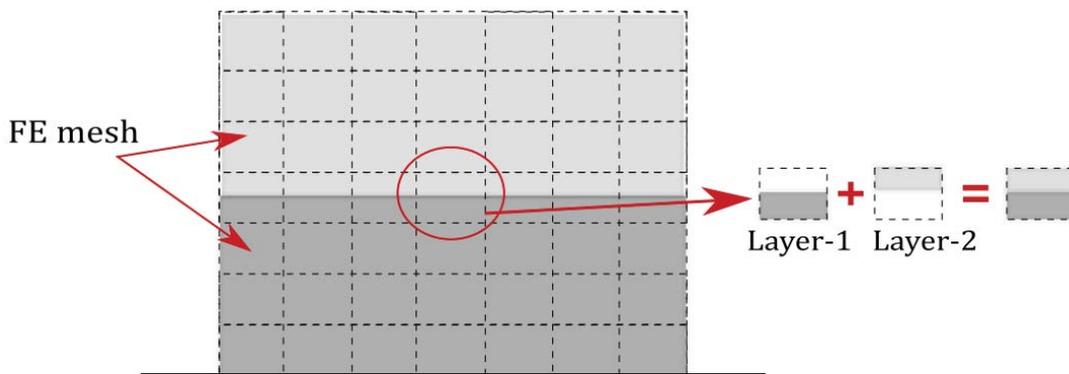


Figure 2: 3D Concrete printed layers and the FE mesh

This has been addressed by introducing a novel modeling approach, applying the FE while considering a pseudo density approach similar to the soft-killing approach in topology optimization, for more detail, see [9].

In addition to the above-mentioned modeling procedure, as mentioned in the previous section uncertainties arising in the model description (geometry, material properties, or loading) are also included by utilizing structural reliability analysis.

Numerical simulation is performed using the commercially available programming language MATLAB, and UQLab (which is an open-source scientific module, a framework for Uncertainty quantification developed at ETH Zurich), to investigate the buildability of 3D concrete printing wall structures during the printing process.

3.1 Numerical example

The wall is modeled as a two-dimensional structure and constructed in a layer-wise process to perform failure analyses. Finite Element analysis with 4-node bi-linear finite elements is applied. The boundary condition considered is the bottom layer fixed as a result of friction on the printed bed [17]. The geometry of the numerical simulation includes wall width $w = 43.5\text{mm}$, wall-length $L = 1000\text{mm}$, the thickness of each layer $t = 10\text{mm}$, the Poisson ratio of the printing material (which is assumed to be constant during the curing process) taken as 0.3. The analysis was limited to a number of 17 layers to avoid the elastic buckling according to a parametric model developed by Suiker et al. [2]. Deterministic uniform load configuration is used only considering the self-weight as a result of the layer-wise production process and activated as the printing progresses.

3.1.1 Numerical simulation of the compressive failure

The temporal changes and the randomness of the compressive strength are represented by random variables [30], generated from the normal distribution [25,26,27]. This random field is characterized statistically by three parameters defining its first moments, the mean, the standard deviation, and the correlation length. Therefore, results obtained from Eq. (2) are taken as the mean value for each Finite Elements(FE), for this process one of the discretization methods i.e. midpoint method is used [16].

In particular, this numerical simulation is focused on buildability (part of overall failure probability) of the 3D printed concrete wall by varying one of the printing process parameter i.e printing velocity. Also, different values of the correlation length have been investigated in the numerical simulation.

The mean and standard deviation can conveniently be combined in terms of the dimensionless coefficient of variation, taking the result from Wolfs et al. [4], to be 0.168. For example, for printing velocity of 1.4 m/mm, the temporal development of compressive strength at the bottom layer is described as a Gaussian random field with a mean based on Eq. (2) is 7.8 kPa and the standard deviation is 1.31. For all the other layers temporal development of compressive strength can be calculated similarly.

In this research, a ‘‘Markovian’’ correlation function is used where the spatial correlation is assumed to decay exponentially with distance [28]. It has the form of:

$$\rho_{(\tau)} = \exp\left(\frac{-2|\tau_{ij}|}{\theta}\right) \quad (5)$$

where θ is the correlation length and τ_{ij} is the separation distance between two finite elements.

Realizations of the random compressive strength fields are produced using covariance matrix decomposition. It is a direct method of producing a homogeneous random field [36]. A covariance matrix is formulated for a single layer, then the covariance matrix is decomposed into a lower and upper triangular matrix via a LU Decomposition, to generate correlated normally-distributed random variables for each layer.

Additionally, the layer-wise production process of the concrete structure is included in the FE model while considering a pseudo-density approach [9].

$$\sigma_y(\mathbf{x}) = \sigma_{y,\min} + \rho^p(\mathbf{x})(\sigma_y - \sigma_{y,\min}) \quad (6)$$

Where $\sigma_y(x)$ denotes the resulting compressive strength of the concrete, $\sigma_{y,\min} > 0$ is a lower bound to avoid zero entries, σ_y refers to the nominal compressive strength, spatially-varying pseudo-density $\rho(x) \in [0,1]$, and $p > 1$ denotes a power-law correlation that is implemented to achieve density values closer to the lower and upper bounds of the design variables.

Based on Wolfs et al. [4] experimental result concrete density is taken as a random variable with normal distribution [25,26,27], the mean value of 2020 kg/m^3 , and the coefficient of variation is 1.96% for the determination of the compressive stress.

To generate the random field for the concrete density, here also the Markovian correlation function Eq. (5) and the covariance matrix decomposition are applied. Correspondingly, material density is modeled considering a pseudo-density approach.

$$\rho_c(x) = \rho_{c,\min} + \rho^p(x)(\rho_c - \rho_{c,\min}) \quad (7)$$

where $\rho_c(x)$ denotes the resulting concrete density, $\rho_{c,\min}(x) > 0$ is a lower bound to avoid zero entries, ρ_c is the nominal concrete density, ρ and p are similar to Eq.(6).

In a random field, the value assigned to each cell or finite element, in this case, is itself a random variable, thus the mesh which has 1050 finite elements, contains 1050 random variables. For both the compressive strength and the compressive stress for each analysis, 3000 realizations were performed with input shown in Table 1, by varying the printing velocity (v_p) and Correlation length (θ).

Parameter	Values considered
v_p (m/min)	0.7, 1.4, 2.1, 3
θ (m)	0.1, 1, 10, 100

Table 1: Parameters varied in the numerical example while holding the other parameters constant.

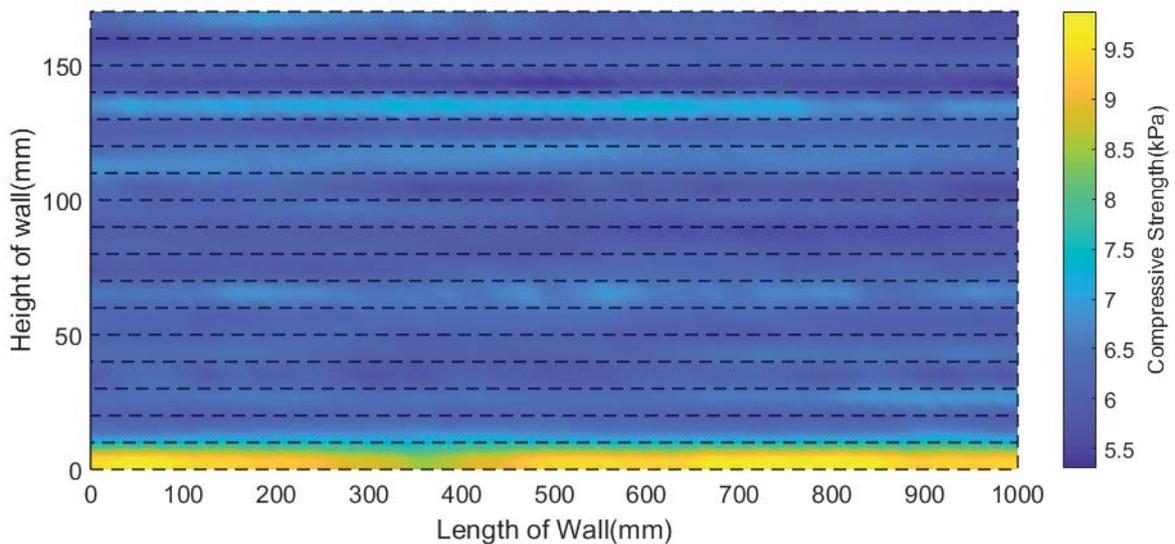


Figure 3: Single realization of the compressive strength for the printing velocity of 2.1 m/mm and correlation length 10m.

A single realization of the compressive strength is shown in Fig.3, both the randomness and time dependency are observed, and the effect of the correlation length is also included in

the realization. Correlation between the properties of the neighboring freshly printed elements is noticed.

The strength-based failure of the structure is studied by comparing the temporal evolution of the compressive yield stress with the increasing hydrostatic pressure caused by subsequently placed concrete layers (compressive stress). Gravity-induced stresses increase with the height of the printed product, whereby the maximum stress values occur in the bottom layer. For each realization, the mean and standard deviation is taken to perform the failure probability at the bottom layer.

Combining the written MATLAB script and UQLab, the result for the printing velocity of 2.1m/min is shown in Fig.4.

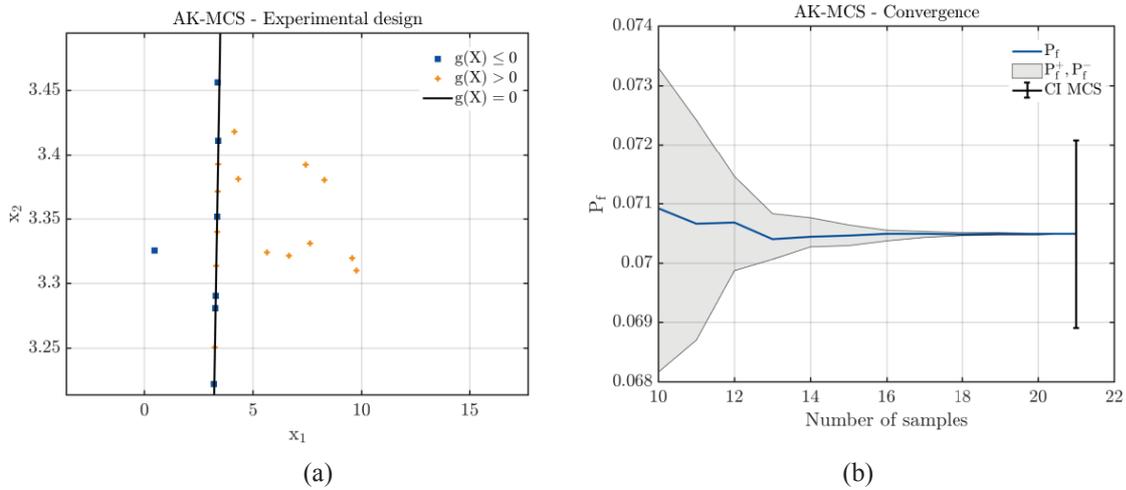


Figure 4: UQLab output for Adaptive Kriging Monte Carlo Simulation printing velocity of 2.1 m /min and correlation length of 0.1 m.

Fig.4 shows graphical visualization of the convergence of the AK-MCS analysis (Fig.4b), a Kriging surrogate model from a small initial sampling of the input vector produces an experimental design iteratively refined close to the currently estimated limit-state the surface $g(x) = 0$ (Fig.4a).

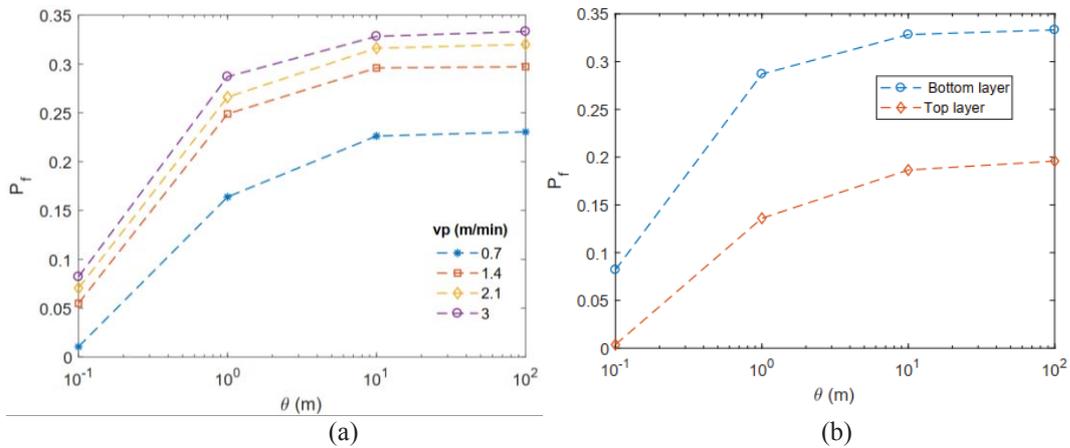


Figure 5: Effect of printing velocity and correlation length on the Probability of failure of compressive strength.

The numerical simulation output, see Fig.5a, indicates that with increasing printing velocity, the probability of failure of the buildability of the printed wall increase. This fact was expected because the 3D concrete printing process is a time-dependent process. Increasing the printing velocity means lowering the curing time, which also results in a lower compressive strength. Another important observation from Fig.5a is the influence of spatial correlation length on the probability of failure. Different correlations cause a large difference in results. If the spatial correlation length for selected printing velocity increases, then the probability of failure also increases.

Furthermore, it can be observed the probability of failure of the top layer is different. The reasons for that are: (1) for the 3D printed wall the concrete age at the top layer is smaller compared to the other layers which affect the compressive strength and (2) to check the accuracy of the numerical simulation. Fig. 5b shows that the probability of failure is higher at the bottom than the other layers as expected even if the curing time for the top layer is smaller than the bottom layer but it is subjected to higher stress, the aforementioned result is for 3m/min printing velocity.

4 CONCLUSION

A numerical modeling procedure of a 3D concrete printed (3DCP) wall is proposed, in which uncertainties implemented via the Stochastic Finite Element Method (SFEM) and a spatially varying pseudo-density approach are considered. To minimize the computational cost, Adaptive Kriging Monte Carlo Simulation (AK-MCS) is utilized.

The aforementioned method enables to predict the probability of failure of the 3D printed concrete wall. The influence of the printing velocity and the spatial correlation length on the probability of failure was studied. For a simple 2D example of a wall, it was shown clearly that the influence of spatial correlation length on the probability of failure is significant. Different correlation lengths cause a difference in the results. However, in practice correlation lengths are often assumed based on literature and these assumed values for the correlation length of concrete properties are not consistent with each other [32]. The suggested values for the correlation length could be the initial values for further investigations.

In the future, additional failure mechanisms, the influence of additional mechanical properties, and process parameters will be studied. It will give rise to a more realistic 3DCP model.

ACKNOWLEDGMENT

The work has been financially supported by different institutions which are highly acknowledged. Among them are DAAD (Ethiopian - German Exchange of Ph.D. candidates), DFG (German Research Foundation) priority program 1886 “Polymorphic uncertainty modeling for the numerical design of structures” and the Federal State of Thuringia, Germany.

REFERENCES

- [1] R. A. Buswell, W. R. Leal de Silva, S. Z. Jones, and J. Dirrenberger. 3d printing using concrete extrusion: A roadmap for research. *Cement and Concrete Research*, 112:37-49, 2018.

- [2] Suiker, A.: Mechanical performance of wall structures in 3D printing processes: Theory, design tools, and experiments. *International Journal of Mechanical Sciences* 137, 145–170,2018.
- [3] Wolfs, R. J. M. Experimental characterization and numerical modeling of 3D printed concrete: controlling structural behavior in the fresh and hardened state. *Eindhoven: Technische Universiteit Eindhoven*,2019
- [4] R. J. M. Wolfs, F. P. Bos, and T. A. M. Salet. Early age mechanical behaviour of 3d printed concrete: Numerical modeling and experimental testing. *Cement and Concrete Research*, 106:103-116, 2018.
- [5] Roussel, N.: Cement and Concrete Research Rheological requirements for printable concretes. *Cement and Concrete Research*, 1–10 (2018). 2018.
- [6] Marchon, D., Kawashima, S., Bessaies-Bey, H., Mantellato, S., Ng, S.: Hydration and rheology control of concrete for digital fabrication: Potential admixtures and cement chemistry. *Cement and Concrete Research* 112: 96–110,2018.
- [7] Le, T.T., Austin, S.A., Lim, S., Buswell, R.A., Law, R., Gibb, A.G., Thorpe, T.: Hardened properties of high-performance printing concrete. *Cement and Concrete Research* 42:558–566,2012.
- [8] Perrot, A.: 3D Printing of Concrete. *ISTE Ltd and John Wiley & Sons, Inc, Great Britain and the United States by*, first edn,2019.
- [9] Mengesha M., Schmidt A., Göbel L., Lahmer T. Numerical Modeling of an Extrusion-Based 3D Concrete Printing Process Considering a Spatially Varying Pseudo-Density Approach. *Second RILEM International Conference on Concrete and Digital Fabrication. DC 2020. RILEM Bookseries, vol 28*, 2020. Springer International Publishing.
- [10] Arregui-Mena, J. D., Margetts, L., & Mummery, P. M. Practical Application of the Stochastic Finite Element Method. *Archives of Computational Methods in Engineering (Vol. 23)*,2016.
- [11] Brenner, C. E., & Bucher, C. G. Stochastic response of uncertain systems. *Archive of Applied Mechanics*, 62(8), 507–516,1992.
- [12] Gomes, H. M., & Awruch, A. M. Reliability of reinforced concrete structures using stochastic finite elements. *Engineering Computations (Swansea, Wales)*, 19(7–8), 764–786,2002.
- [13] Sudret, B., & Kiureghian, A. Der.Stochastic finite element methods and reliability. *Ucb/Semm-2000*, (October), 189,2000.
- [14] Andrea, B., & Brunel, J. F. SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA Bratislava 2013.
- [15] Aldosary, M., Wang, J., & Li, C. Structural reliability and stochastic finite element methods: State-of-the-art review and evidence-based comparison. *Engineering Computations (Swansea, Wales)*, 35(6), 2165–2214,2018.
- [16] Bergman, L. A., Shinozuka, M., Bucher, C. G., Sobczyk, K., Dasgupta, G., Spanos, P. D., ... Zhang, R. A state-of-the-art report on computational stochastic mechanics. *Probabilistic Engineering Mechanics*, 12(4), 197–321,1997.

- [17] Gordon A. Fenton and D. V. Griffiths. Risk Assessment in Geotechnical Engineering. *John Wiley & Sons*, 2008.
- [18] Papadrakakis, M., & Papadopoulos, V. Robust and efficient methods for stochastic finite element analysis using Monte Carlo simulation. *Computer Methods in Applied Mechanics and Engineering*, 134(3–4), 325–340,1996.
- [19] Sataloff, R. T., Johns, M. M., & Kost, K. M. (n.d.). Reliability-Based Design in Geotechnical Engineering. *Taylor & Francis e-Library*, 2008.
- [20] R. A. Buswell, W. R. Leal de Silva, S. Z. Jones, and J. Dirrenberger. 3D printing using concrete extrusion: A roadmap for research. *Cement and Concrete Research*, 112(May):37-49, 2018.
- [21] J. van der Putten, G. de Schutter, and K. van Tittelboom. The effect of print parameters on the (micro)structure of 3d printed cementitious materials. In *Timothy Wangler and Robert J. Flatt, editors, First RILEM International Conference on Concrete and Digital Fabrication -Digital Concrete 2018*, pages 234-244, Cham, 2019. Springer International Publishing.
- [22] Marelli, S., Schöbi, R., Sudretm, B.: UQLab user manual - Structural reliability (Rare event estimation), *Report # UQLab-VI.3-107* ,2019.
- [23] Wolfs, R.J., Suiker, A.S.: Structural failure during extrusion-based 3D printing processes. *International Journal of Advanced Manufacturing Technology* 104(1-4), 565-584 ,2019.
- [24] Echard, B., Gayton, N., & Lemaire, M. AK-MCS: An active learning reliability method combining Kriging and Monte Carlo Simulation. *Structural Safety*, 33(2), 145–154.,2011.
- [25] De Araujo, J.M.: Probabilistic analysis of reinforced concrete columns. *Advances in Engineering software* 32, 871-879 ,2001.
- [26] El-Reedy, M.A.: Reinforced Concrete Structural Reliability. *CRC Press* ,2012.
- [27] Seo, D., Shin, S., Han, B.: Reliability-based structural safety evaluation of reinforced concrete members. *Journal of Asian Architecture and Building Engineering* 9(2), 471-478 ,2010.
- [28] Vanmarcke, B. E., Asce, M., & Grigoriu, M. Stochastic finite element analysis of simple beams, *109(5)*, 1203–1214,1984.
- [29] Bottenbruch, H., Pradlwarter, H.J., Schuëller, G.I.: The influence of spatial correlation of concrete strength on the failure probabilities of reinforced concrete chimneys. *Materials and Structures* 22(4), 255-263 ,1989.
- [30] Vanmarcke, E., Shinozuka, M., Nakagiri, S., Schuëller, G. I., & Grigoriu, M. Random fields and stochastic finite elements. *Structural Safety*, 3(3–4), 143–166,1986.
- [31] Yang, Y., Peng, J., Zhang, J., & Cai, C. S. . A new method for estimating the scale of fluctuation in reliability assessment of reinforced concrete structures considering spatial variability. *Advances in Structural Engineering*, 21(13), 1951–1962,2018.
- [32] Criel, P., Caspeele, R., & Taerwe, L. Bayesian updated correlation length of spatial concrete properties using limited data. *Computers and Concrete*, 13(5), 659–677,2014

- [33] Griffiths, D. V., & Lane, P. A. Slope stability analysis by finite elements. *Geotechnique*, 49(3), 387–403,1999.
- [34] Maiti, & Bidinger. CISM COURSES AND LECTURES. *Journal of Chemical Information and Modeling*, 53(9), 1689–1699,1981.
- [35] Giulio Maier - Milan, Jean Salençon - Palaiseau, & Wien, W. S.-. *cism courses and lectures. the International Center for Mechanical Sciences, SpringerWien NewYork*,2007.
- [36] De Schutter, G., Lesage, K., Mechtcherine, V., Nerella, V., Habert, G., Agusti-Juan, I.: Vision of 3D printing with concrete | Technical, economic and environmental potentials. *Cement and Concrete Research* 112, 25-36 ,2018.

EFFICIENT DISCRIMINATION BETWEEN BIOLOGICAL POPULATIONS VIA NEURAL-BASED ESTIMATION OF RÉNYI DIVERGENCE¹

Anastasios Tsourtis², Georgios Papoutsoglou² and Yannis Pantazis²

²Institute of Applied and Computational Mathematics,
Foundation for Research and Technology - Hellas, Greece
e-mail: {tsourtis, papoutsoglou, pantazis}@iacm.forth.gr

¹This research is co-financed by Greece and the European Union (European Social Fund-ESF) through the Operational Programme “Human Resources Development, Education and Lifelong Learning 2014-2020” in the context of the project “Characterizing Population Dynamics with Applications in Biological Data” (MIS 5050686).

Keywords: Statistical populations, Rényi divergence, Variational representation, Neural networks

Abstract. *The advent of single-cell or single-molecule sampling techniques allowed the study of abnormalities in small subsets of cell populations as well as subtle differences of evolving biological phenomena. A major challenge in this effort is the quantification of statistical differences between the probability distributions of measured quantities; particularly when small perturbations or small distributional changes need to be detected. Here, we propose to use as a discriminative tool the Rényi divergence whose key advantage is its ability to highlight differences between probability tails. In addition, we describe an algorithm which is based on a variational representation formula for the Rényi divergence and implicitly estimates it by solving an optimization problem. We evaluate the discrimination performance on both synthetic and real datasets. The proposed algorithm is able to detect distributional differences which are below 0.5% and quantify the trade-off between number of samples and neural network complexity. The comparison with existing density ratio approaches reveals that the proposed method is significantly better when the dimension of the data is moderately high (e.g., larger than 10).*

1 INTRODUCTION

Population datasets emerge in many scientific fields such as biology [1, 2], ecology [3], epidemiology [4] and molecular motion in biochemistry [5, 6] to name a few. Particularly in biology typical population datasets now consist of tens of thousands of samples measuring dozens of quantities of interest. For instance, high dimensional single-cell technologies, such as flow and mass cytometry [7], are able to capture the abundance of up to 40 proteins on thousands of cells simultaneously. Such moderate to high dimensional datasets cannot be screened out manually (e.g. through scatter plots) and computational approaches are required for the detection of clusters and differences in the data. Despite the recent proposal of computational tools that handle single-cell population data [2, 8], a major challenge in the characterization of cellular heterogeneity still remains. Indeed, it is often the case that rare sub-populations exist in the samples which are very difficult to be detected due to their low abundance levels. For instance, stem and progenitor cells are underrepresented in the total cell population therefore; they are rarely detected using general-purpose methodologies over large populations of cells.

Statistical quantities such as the mean value and the covariance matrix are not sufficient discriminative metrics because they do not capture the complete probabilistic characteristics of the two populations. On the other hand, a probability distance or a divergence could capture all the statistical information induced by the observed sample distributions. Additionally, probability distances which are sensitive to small perturbations and be able to detect small distributional changes are ideal choices. In this paper, we suggest using Rényi divergence as an approach to discriminate between two population datasets. Rényi divergence has the advantage that its hyper-parameter controls how much weight to put on the tails of the distributions thus it can become very sensitive to rare sub-populations inside the population datasets (see Figure 2 for three examples).

The estimation of the Rényi divergence becomes feasible with the use of a variational representation [9, 10, 11]. Variational representations essentially transform the estimation of a divergence to an optimization problem over an infinite-dimensional function space. Then, the function space is approximated by a neural network parametrized space in a similar fashion to [12, 13, 14, 15]. Thus, we present and then evaluate an algorithm that estimates the Rényi divergence which we named NERD (Neural-based Estimation of Rényi Divergence) algorithm. The utilization of neural networks offers additional advantages such as the ability to handle high dimensional data as well as any type of input data with the trade-off being the requirement for a large sample size; a limitation which is already alleviated in practice by the production of large amounts of single-cell measurements per experiment.

We first evaluate NERD algorithm on synthetic data where the ground truth is known. NERD is capable of handling high-dimensional data better than state-of-the-art methods such as ITE [16]. We assess the behavior of the estimator as a function of various hyperparameters such as the number of samples, the rarity of the sub-population and the choice of function space. We numerically show that NERD algorithm is capable of accurately estimating the Rényi divergence in high dimensions given enough sample size. We also compute the discriminative capabilities of NERD between single-cell populations. The two populations consist of cells from healthy participants as well as healthy cells contaminated by a small portion of “sick” cells. We show that NERD algorithm can discriminate confidently when the percentage of rare subpopulation is above 0.2% and the number of available samples is above 40K.

2 DEFINITION AND PROPERTIES OF THE RÉNYI DIVERGENCE

Let Q and P be two probability measures (or distributions) on a measurable space (Ω, \mathcal{M}) . The Rényi divergence of order $\alpha > 0$ with $\alpha \neq 1$ of Q with respect to P is defined as [17, 18]

$$\mathcal{R}_\alpha(Q||P) := \frac{1}{\alpha(\alpha-1)} \log \mathbb{E}_P \left[\left(\frac{dQ}{dP} \right)^\alpha \right] \quad (1)$$

when Q and P are mutually absolutely continuous¹ with respect to each other, otherwise, $\mathcal{R}_\alpha(Q||P) = \infty$. The ‘ratio’ $\frac{dQ}{dP}$ is the Radon-Nikodym derivative of Q with respect to P which always exists due to the imposed absolute continuity condition. The defining properties of a divergence are that (a) it is non-negative and (b) it equals to zero if and only if $Q = P$. Despite not being a distance since it is neither symmetric nor satisfies the triangular inequality, divergences are widely used for the comparison of probability distributions.

In some studies, the definition of Rényi divergence utilizes the factor $\frac{1}{\alpha-1}$ (cf. [19, 20, 9]) instead of $\frac{1}{\alpha(\alpha-1)}$, nevertheless, we prefer the definition (1) due to the symmetry property

$$\mathcal{R}_\alpha(Q||P) = \mathcal{R}_{1-\alpha}(P||Q)$$

when $0 < \alpha < 1$. Using this symmetry property, the definition of Rényi divergence is straightforwardly extended to $\alpha < 0$ (e.g., $\mathcal{R}_{-1}(Q||P) := \mathcal{R}_2(P||Q)$).

The definition of Rényi divergence is extended to $\alpha = 1$ where the limit equals to the Kullback-Leibler divergence defined by

$$D_{KL}(Q||P) := \int \log \frac{dQ}{dP} dQ \quad (2)$$

when $Q \ll P$, otherwise $D_{KL}(Q||P) = +\infty$ as well as to $\alpha = 0$ where the limit equals to the reverse² Kullback-Leibler divergence. Interestingly, several other divergences are linked to Rényi divergence. Rényi divergence has an one-to-one and onto correspondence with α -divergence [21] where Rényi divergence can be obtained as an affine transformation of the logarithm of the α -divergence. Rényi divergence is also related to Hellinger distance [22] as well as to χ^2 -divergence [22] for particular values of α . Figure 1 summarizes those relationships. Further properties of the Rényi divergence can be found for instance in [23, 20, 9].

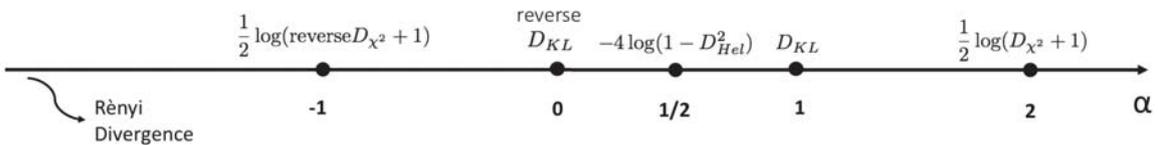


Figure 1: Rényi divergence as a function of its order α and its connections to other divergences. The case $\alpha = 0.5$ relates with the Hellinger distance while the cases $\alpha = 1$ & 2 relate with Kullback-Leibler and (Pearson’s) χ^2 divergence, respectively. Rényi divergence is reverse symmetric around 0.5 thus the cases $\alpha = 0$ & -1 relate with reverse Kullback-Leibler and reverse (i.e., Neyman’s) χ^2 divergence, respectively.

¹We say that Q is absolutely continuous with respect to P if for every measurable set $A \in \Omega$, $P(A) = 0 \Rightarrow Q(A) = 0$. It is written as $Q \ll P$.

²In the sense that the order of Q and P has been reversed.

2.1 Rényi Divergence Highlights Distributions' Tail Differences

Existing literature has shown that Rényi divergence is capable of efficiently bounding the probability of rare events and more generally of risk-sensitive observables of a distribution [18, 24, 25] through its order parameter. Intuitively, the order parameter as a power factor of the density ratio leverages the amount of weight put on the tails of the distributions. For instance, in [24], to discriminate between rare events from distributions with infinitesimal small differences the order had to be sent to infinity.

We demonstrate this sensitivity property of the Rényi divergence through a series of examples. First, we consider two zero-centered univariate Gaussian distributions with different standard deviations³, $Q \equiv \mathcal{N}(0, \sigma_1^2)$ and $P \equiv \mathcal{N}(0, \sigma_0^2)$. The Rényi divergence of Q with respect to P is given by [26]

$$\mathcal{R}_\alpha(Q||P) = \begin{cases} \frac{1}{\alpha} \log \frac{\sigma_0}{\sigma_1} + \frac{1}{2\alpha(\alpha-1)} \log \frac{\sigma_0^2}{\alpha\sigma_0^2 + (1-\alpha)\sigma_1^2} & \text{if } \alpha\sigma_0^2 + (1-\alpha)\sigma_1^2 > 0 \\ +\infty & \text{otherwise} \end{cases} \quad (3)$$

As α approaches to the ‘finiteness’ limit $\frac{\sigma_1^2}{\sigma_1^2 - \sigma_0^2}$, the Rényi divergence takes exponentially-large values resulting in an unequivocal discrimination between the two distributions. Going one step further, if $\sigma_1^2 = \sigma_0^2(1 + \epsilon)$ with ϵ being a small number then α should be of order $O(\epsilon^{-1})$ in order to efficiently discriminate between the two distributions. Figure 2(a) demonstrates this behavior for two values of ϵ . Analogous discriminative capacity is observed when the Rényi divergence between a Gaussian distribution with full covariance matrix and a Gaussian with diagonal covariance structure is calculated. In this second example, let Q be a zero-mean Gaussian with covariance matrix $\Sigma_1 = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ and P a zero-mean Gaussian with covariance matrix equal to the identity matrix (i.e., independent components). Figure 2(b) presents the Rényi divergence as a function of its order. As it is evident from the plot, there are both positive and negative α 's of order $O(\rho^{-1})$ that assign very large values to the Rényi divergence implying that even very small correlations between variables could be detected when $|\alpha|$ becomes sufficiently large. Additionally, both Gaussian examples show that large values for the order may result to infinite Rényi divergence and there is a finiteness limit for α that should not be exceeded. Therefore, caution must be placed on the choice of the order value. As a rule of thumb, the ‘‘closer’’ the two distributions are the larger the value of α can (or must) be set.

In this paper, we suggest exploiting this sensitivity property and distinguish between statistical populations of data that differ slightly by containing samples from rare sub-populations. Rare sub-populations are hard to detect exactly because of their rarity. Therefore, we aim to search for the highest value for Rényi divergence by tuning α . As a third and more relevant motivation example, Figures 2(c) & (d) present the Rényi divergence between a mixture of two Gaussians ($Q \equiv (1-w)\mathcal{N}(\mu_0, \sigma_0^2) + w\mathcal{N}(\mu_1, \sigma_1^2)$) with w corresponding to the percentage of the less probable population and a Gaussian ($P \equiv \mathcal{N}(\mu_0, \sigma_0^2)$). Under this particular setting, there is an optimal α that maximizes the Rényi divergence. Additionally, the smaller the percentage of the less probable population the larger the value of the optimal α is. This is consistent with the two previous examples in the sense that distributions with smaller differences require larger values of α in order to obtain larger Rényi divergence values.

³In this paper, we always consider P to be the baseline (or unperturbed) distribution while Q is the different or perturbed one.

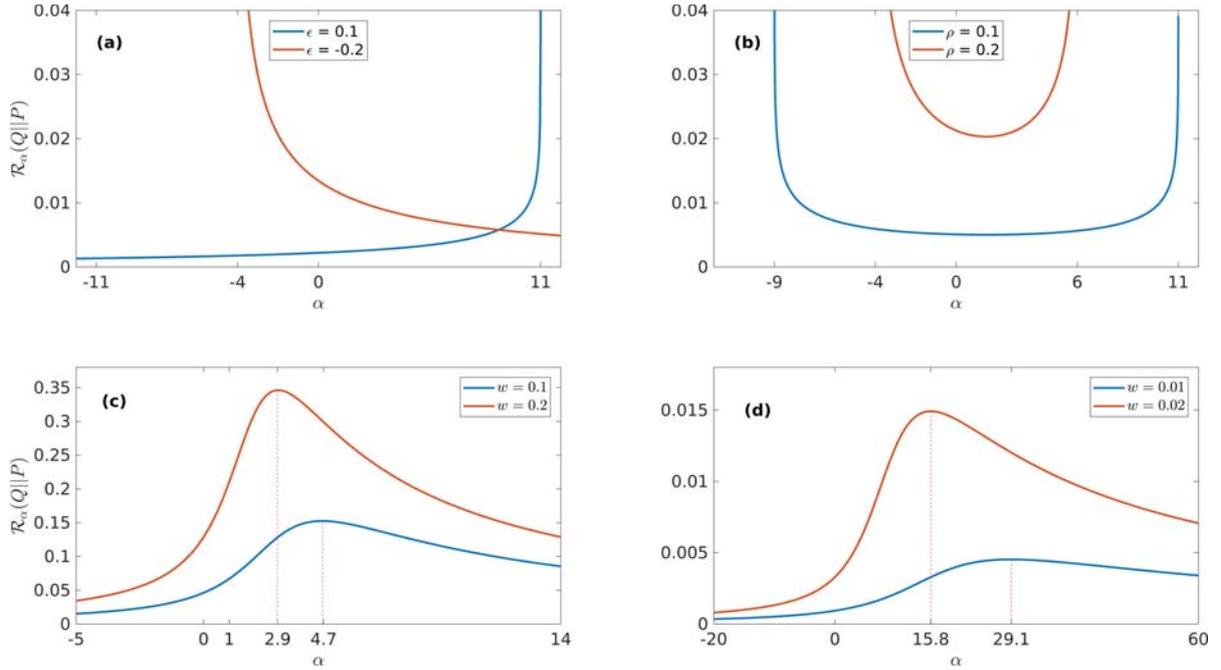


Figure 2: Rényi divergence of Q (perturbed) with respect to P (unperturbed) as a function of α between: **(a)** two 1D zero-mean Gaussian distributions with different variances ($\sigma_0^2 = 1, \sigma_1^2 = 1 + \epsilon$), **(b)** two 2D zero-mean Gaussian distributions with different covariance structure (ρ : correlation coefficient between the two elements of the perturbed Gaussian) and **(c)-(d)** between a mixture of two Gaussians and a Gaussian distribution (w : the percentage of the second mode in the mixture).

2.2 A Variational Representation Formula for the Rényi Divergence

By definition, the estimation of Rényi divergence requires either the knowledge of the densities of the probabilities involved or an approximation of their ratio. An alternative approach is to transform the estimation problem into an optimization problem via the utilization of a variational representation. A variational representation formula is essentially a lower bound of the divergence for which the optimal solution gives rise to the divergence value. It consists of two mathematical ingredients: the function space where the optimal solution will be searched for and, the representation expression, called here the ‘objective functional’, whose optimization leads to the value of the divergence.

The following theorem, proved in [11], states that Rényi divergence is the solution of a variational optimization with an objective functional which is the difference of two risk-sensitive observables (i.e., the expression inside the curly brackets in (4)).

Theorem 1. *Let P, Q be two probability measures on (Ω, \mathcal{M}) and $\alpha \in \mathbb{R} \setminus \{0, 1\}$. Then,*

$$\mathcal{R}_\alpha(Q||P) = \sup_{g \in \mathcal{M}_b(\Omega)} \left\{ \frac{1}{\alpha - 1} \log \mathbb{E}_Q[e^{(\alpha-1)g}] - \frac{1}{\alpha} \log \mathbb{E}_P[e^{\alpha g}] \right\}, \quad (4)$$

where $\mathcal{M}_b(\Omega)$ is the space of all (real-valued) measurable and bounded functions from Ω to \mathbb{R} and we assume the conventions $+\infty - \infty = -\infty$ and $-\infty + \infty = -\infty$.

The optimal solution under appropriate conditions provided in [11] can be explicitly written as $g^* = \log \frac{dQ}{dP}$, and, the aim of this paper is to approximate g^* as accurately as possible. Finally, taking $\alpha \rightarrow 1$, we recover the Donsker-Varadhan variational formula for the Kullback-Leibler

divergence [27] which is given by

$$D_{KL}(Q||P) = \sup_{g \in \mathcal{M}_b(\Omega)} \{ \mathbb{E}_Q[g] - \log \mathbb{E}_P[e^g] \} . \quad (5)$$

Hence, equation (4) can be seen as a generalization of the Donsker-Varadhan formula to the Rényi divergence. Similarly, the Donsker-Varadhan formula with the order of Q and P reversed is obtained when $\alpha \rightarrow 0$.

3 NEURAL-BASED ESTIMATION OF RÉNYI DIVERGENCE (NERD)

The variational formula in (4) is still not fully practical because (a) the expectations cannot be explicitly computed since Q and P are not known, and (b) the infinite dimensional space of test functions (i.e., of g 's) needs to be restricted to a parametric representation that can be handled by a computer. Regarding the first issue, the expectations are replaced by their statistical averages using a finite number of samples. This approximation fits well with our setting since we only have access to samples from the distributions of interest. Moreover, as the number of samples tends to infinity, the statistical averages converge to the respective expectation values.

For the latter issue, we concentrate to $\Omega = \mathbb{R}^d$ and parametrize the space of all measurable and bounded functions with neural networks of bounded activation function for the output layer. Letting $\theta \in \mathbb{R}^p$ be the parameter vector with the weights and biases of the neural network, our aim is to optimize $g_\theta : \mathbb{R}^d \rightarrow [-M, M]$ where M is a user-defined clipping factor. In our experiments we enforce the boundedness condition via the use of $M \tanh\left(\frac{\cdot}{M}\right)$ as the activation function of the output layer. The error induced by this second approximation can be controlled using (a) Lusin's theorem [28] where the space of all measurable and bounded functions is replaced with all continuous and bounded functions with arbitrary accuracy and (b) the fact that a large enough neural network is a universal approximator of continuous and bounded functions [29, 30].

The parameters of the neural network are estimated using *stochastic gradient ascent* because we are searching for the solution that maximizes the objective functional. The pseudo-code of the neural-based Rényi divergence estimator is provided in Algorithm 1. When $\alpha = 1$ or 0 , we apply the finite sampling approximation formulas stemming from the Donsker-Varadhan variational representation (5).

3.1 Statistical Properties of NERD

The asymptotic consistency of NERD has been shown in [11, Theorem 2]. However, stability and consistency results as well as bias-variance trade-offs for finite number of samples is an open and active problem even for the Kullback-Leibler case [31, 32, 33]. The main encumbrance stems from the fact that both terms in the objective function of Rényi's variational representation are sensitive to tail events and the variance of the estimator could grow exponentially with the true value of the divergence [33]. A partial solution proposed in [33] sets a small clipping factor M applied to the output of the final layer which results in reduced variance for the estimator at the cost of larger bias especially when the value of Rényi divergence is high since the maximum possible value for the estimator is $2M$. As it is already presented, NERD algorithm has adopted the clipping operator. In the following section, we propose a different approach to reduce the variance by utilizing a different, more regularized function space for the optimization problem.

Algorithm 1 Neural Estimation of Rényi Divergence (NERD)

Input: Sample matrix $X \in \mathbb{R}^{N \times d} \sim Q$, sample matrix $Y \in \mathbb{R}^{N \times d} \sim P$, order parameter α , neural network $g_\theta(\cdot)$, batch size m and learning rate λ_{lr}

Output: Rényi divergence estimate: \hat{R}_α^N

1: $\theta \leftarrow \text{Initialize_Neural_Network}()$

2: **while** not converged **do**

3: Choose randomly m samples from X : $\{x_i\}_{i=1}^m$ and from Y : $\{y_i\}_{i=1}^m$

4: Compute the variational expression:

$$R(\theta) = \frac{1}{\alpha - 1} \log \frac{1}{m} \sum_{i=1}^m e^{(\alpha-1)g_\theta(x_i)} - \frac{1}{\alpha} \log \frac{1}{m} \sum_{i=1}^m e^{\alpha g_\theta(y_i)} \quad (6)$$

5: Update the neural net's parameters:

$$\theta \leftarrow \theta + \lambda_{lr} \nabla_\theta R(\theta)$$

6: **end while**

7: Compute the variational estimate using all samples:

$$\hat{\mathcal{R}}_\alpha^N = \frac{1}{\alpha - 1} \log \frac{1}{N} \sum_{i=1}^N e^{(\alpha-1)g_\theta(x_i)} - \frac{1}{\alpha} \log \frac{1}{N} \sum_{i=1}^N e^{\alpha g_\theta(y_i)} \quad (7)$$

3.2 Using Lipschitz Continuous Functions as Test Functions

The space of test functions can be selected differently. The cost of choosing a subset of $\mathcal{M}_b(\Omega)$ is that a lower bound –but not necessarily strictly lower– for the divergence is obtained. Given that we are interested in alleviating the impact of finite sampling on the approximated risk-sensitive observables, we propose to use Lipschitz continuous functions with Lipschitz constant K as the function space over which the optimal solution will be sought for. The 1-Wasserstein distance, which is also defined on the Lipschitz function space but uses a different objective functional, has shown significantly better stability and convergence properties during the training of GANs [14, 15]. Thus, we anticipate improved statistical properties such as reduced variance in our experiments. Theoretically, it has also been shown that the function space replacement from measurable to Lipschitz functions retains the divergence property for the α -divergence [34] hence it is also retained for the Rényi divergence.

From an implementation perspective, the only difference for NERD algorithm is the removal of the clipping function and the addition of a gradient penalty term in (6). The new formula is given by

$$R(\theta) = \frac{1}{\alpha - 1} \log \frac{1}{m} \sum_{i=1}^m e^{(\alpha-1)g_\theta(x_i)} - \frac{1}{\alpha} \log \frac{1}{m} \sum_{i=1}^m e^{\alpha g_\theta(y_i)} + \lambda_{GP} \frac{1}{m} \sum_{i=1}^m \max(0, \|\nabla_x g_\theta(z_i)\|^2 - K) , \quad (8)$$

where $z_i = u_i x_i + (1 - u_i) y_i$ and $u_i \sim \mathcal{U}(0, 1)$ for $i = 1, \dots, m$. We remark that this is the one-sided gradient penalty and it is only activated when the square of the gradient's norm is above K . The two-sided gradient penalty which is valid for the Wasserstein distance is not applicable

for the Rényi divergence since the norm of the gradient is not everywhere equal to one for the optimal test function.

4 RESULTS

In this Section, we test the accuracy of the proposed algorithm on two synthetic examples as well as its discriminative efficacy on one real biological dataset. Our aim is to numerically evaluate the performance of NERD algorithm on the statistical estimation of Rényi divergence and also explore the Rényi’s order parameter that leads to the most efficient discrimination between two sample distributions with small sub-population differences. Our results are compared against a state-of-the-art density ratio approximation algorithm implemented by the Information Theoretical Estimators (ITE) toolbox [16].

4.1 Experimental setup

In Section 3, we presented two variants of the NERD algorithm depending on the chosen space of test functions: i) the space of continuous and bounded test functions referred to as NERD_{C_b} , and ii) the space of Lipschitz continuous test functions referred to as NERD_{Lip} . The boundedness condition of NERD_{C_b} is enforced through a bounding factor $M > 0$ on the activation function of the final layer. In contrast, the Lipschitz continuous condition is enforced through the addition of a regularization term that depends on two hyper-parameters: the Lipschitz constant $K > 0$ and the regularization coefficient λ_{GP} . Both M and K are capable of affecting the trade-off between estimation bias and estimation variance with smaller values favoring reduced variance with the cost of increased bias. Table 1 summarizes the value ranges of all (hyper-)parameters that appear in our numerical experiments.

Neural network hyperparameters were set following a similar rationale as in [12] where the Kullback-Leibler case was studied and the implementation is carried out using TensorFlow2⁴. Specifically, we employ fully-connected feed-forward neural networks with $l = 3$ layers, variable number of units per layer and $\tanh(\cdot)$ as activation function for the hidden layers. The number of units per layer is primarily dependent on the dimension of the data while the number of trainable parameters is typically of order $\mathcal{O}(10^3)$. Given that the sample size of the the studied datasets is between $\mathcal{O}(10^4) - \mathcal{O}(10^5)$ we anticipate no overfitting. For the sake of fairness, both NERD variants share the same architecture (i.e., hidden layers, number of units per layer, activation function) except the activation function of the output layer which is different.

We apply Adam optimizer [35] as the training algorithm with its default hyperparameter values. The learning rate is set to $\lambda_{lr} = 0.0005$ for the synthetic examples while the number of iterations was $N_{it} = 20000$. Due to slower convergence, the respective values for the real dataset are $\lambda_{lr} = 0.01$ and $N_{it} = 60000$. Moreover, we set a large value to the batch size so that samples from the tails are included in the statistical average with high probability at each step.

We also set the hyperparameter of the ITE-based estimator that defines the number, k , of nearest neighbors. Since the computational cost increases non-linearly with k , ITE becomes prohibitive for high dimensions and large sample sizes. We found that setting $k = 20$ is a balanced choice for approximating the Rényi divergence in our experiments.

Finally, our primal goal in the synthetic examples is to assess the accuracy of the estimator, hence, we fix the order of the Rényi divergence to $\alpha = 0.5$. Such order value provides a stable statistical behavior with low variance for the estimator relative to the other values of α . In contrast, for the real dataset example, we provide results for a range of α values excluding

⁴Code will be available upon acceptance.

Table 1: Parameters’ symbols, their categorization and range in our experiments.

Parameter	Explanation	Association	Range
N	No. samples	Data set	$\mathcal{O}(10^4) - \mathcal{O}(10^5)$
d	Dimension	Data set	[1, 50]
w	Sub-population proportion	Data set	[0.002, 0.2]
ρ	Correlation coefficient	Data set	[0, 0.9]
α	Order	Rényi divergence	[0.1, 0.9]
k	No. nearest neighbors	ITE	20
M	Boundedness const.	NERD (bounded)	[1, 50]
K	Lipshcitz const.	NERD (Lipschitz)	[1, 10]
λ_{GP}	Gradient penalty	NERD (Lipschitz)	0.1
N_{it}	No. iterations (training steps)	Training alg.	[10000, 150000]
m	Batch size	Training alg.	4000
λ_{lr}	Learning rate	Training alg.	[0.0005, 0.01]
l	No. hidden layers	Neural network	3
θ	Vector w/ weights & biases	Neural network	—
p	Dimension of θ	Neural network	$\mathcal{O}(10^2) - \mathcal{O}(10^3)$

$\alpha \in \{0, 1\}$ wherein the variance of the estimator might become very large [32, 33].

4.2 Rényi Divergence Estimation on Synthetic Data

4.2.1 Between a Gaussian Mixture Model (GMM) and a Gaussian

In our first example we consider Q to be a 1-D bimodal distribution (mixture of two Gaussians) and P a 1-D Gaussian distribution. The first mode of Q will be referred to as the ‘main’ mode and the second one as the ‘rare’ mode, inspired by biological data terminology of main and rare cell sub-populations. The mean and variance of P and of the main mode of Q were set equal to one another. Specifically,

$$\begin{aligned} Q &= (1 - w)\mathcal{N}(\mu_0, \sigma_0) + w\mathcal{N}(\mu_1, \sigma_1) \\ P &= \mathcal{N}(\mu_0, \sigma_0) \end{aligned} \quad (9)$$

where the μ ’s and the σ ’s denote the means and variances of the distributions and w is the probability of the rare mode. In our simulations, we set $\mu_0 = 0$, $\sigma_0 = 1$ for the main mode and $\mu_1 = 1$, $\sigma_1 = \frac{1}{4}$ for the rare mode. The upper panels of Figure (3) illustrate the convergence of all estimators as the sample size increases. As expected, larger sample sizes reduce the variance of the estimators and $N = 50000$ is sufficient for this example. Additional experimentation not shown here revealed that similar results are obtained for other values of μ_1 , σ_1 and α .

The lower panels in Figure 3 depict the effect of the probability of the rare mode w on the estimation of the Rényi divergence. Since the percentage of samples between the main and the rare mode of Q is controlled by w , we consider the range between 0.3% and 10% as being representative of the frequencies found in cases of rare cell populations in disease-like situations. Apparently, as w decreases it becomes harder to differentiate between Q and P . Interestingly, both NERD variants are more accurate both in terms of variance and bias in discriminating between slightly different sample distributions for the same sample size. On the other hand, the ITE estimator has undeniable difficulties with small values for w due to its large variance.

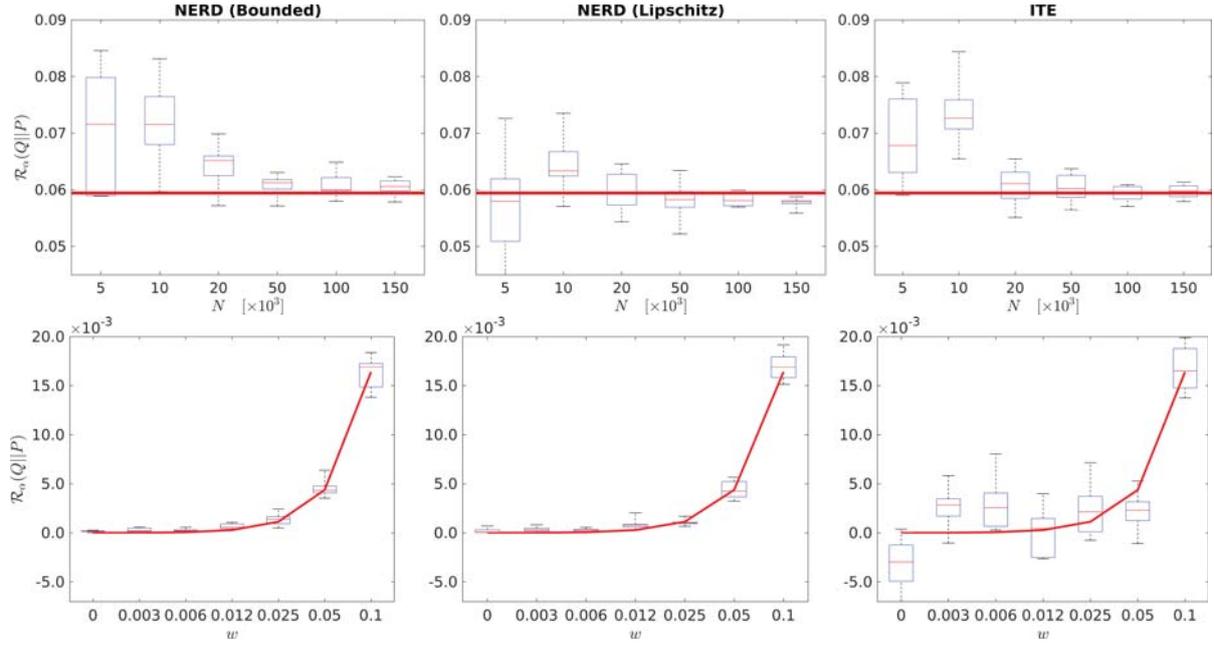


Figure 3: Estimated Rényi divergence between a GMM with two modes and a Gaussian distribution using both variants of NERD algorithm and ITE with the order fixed at $\alpha = 0.5$. *Upper panels:* As the sample size N increases, all methods converge to the exact Rényi divergence value (red solid line) with decreasing variance. Here, we set $w = 0.2$ and quantiles in the box-plots are estimated over 10 independent runs. *Lower panels:* Estimated Rényi divergence as a function of the sub-population proportion w . Here, the sample size is $N = 40000$. Evidently, both variants of NERD exhibit less bias and reduced variance relative to ITE-based estimator for $w < 0.05$.

4.2.2 Between Two High-dimensional Gaussians

In this example, we let Q and P be two zero-mean multivariate (standardized) Gaussian random variables of dimension d with different covariance matrices. We impose the element-wise correlation $\text{corr}(x_i, x_{\frac{d}{2}+j}) = \delta_{i,j}\rho$ to the samples $x \sim Q$ where $i, j = 1, \dots, \frac{d}{2}$ and $\delta_{i,j}$ is Kronecker's delta. In contrast, no correlation is assumed for the samples $y \sim P$. We test how the estimation accuracy of both variants of NERD changes with increasing dimension as well as correlation coefficient. This setting is quite challenging because as dimension increases the probability mass concentrates in a ball around the origin whose radius is exponentially decreasing with respect to the dimension. Moreover, as the correlation coefficient increases the intersection between the supports of the sample distributions is significantly reduced which could result in large estimation errors.

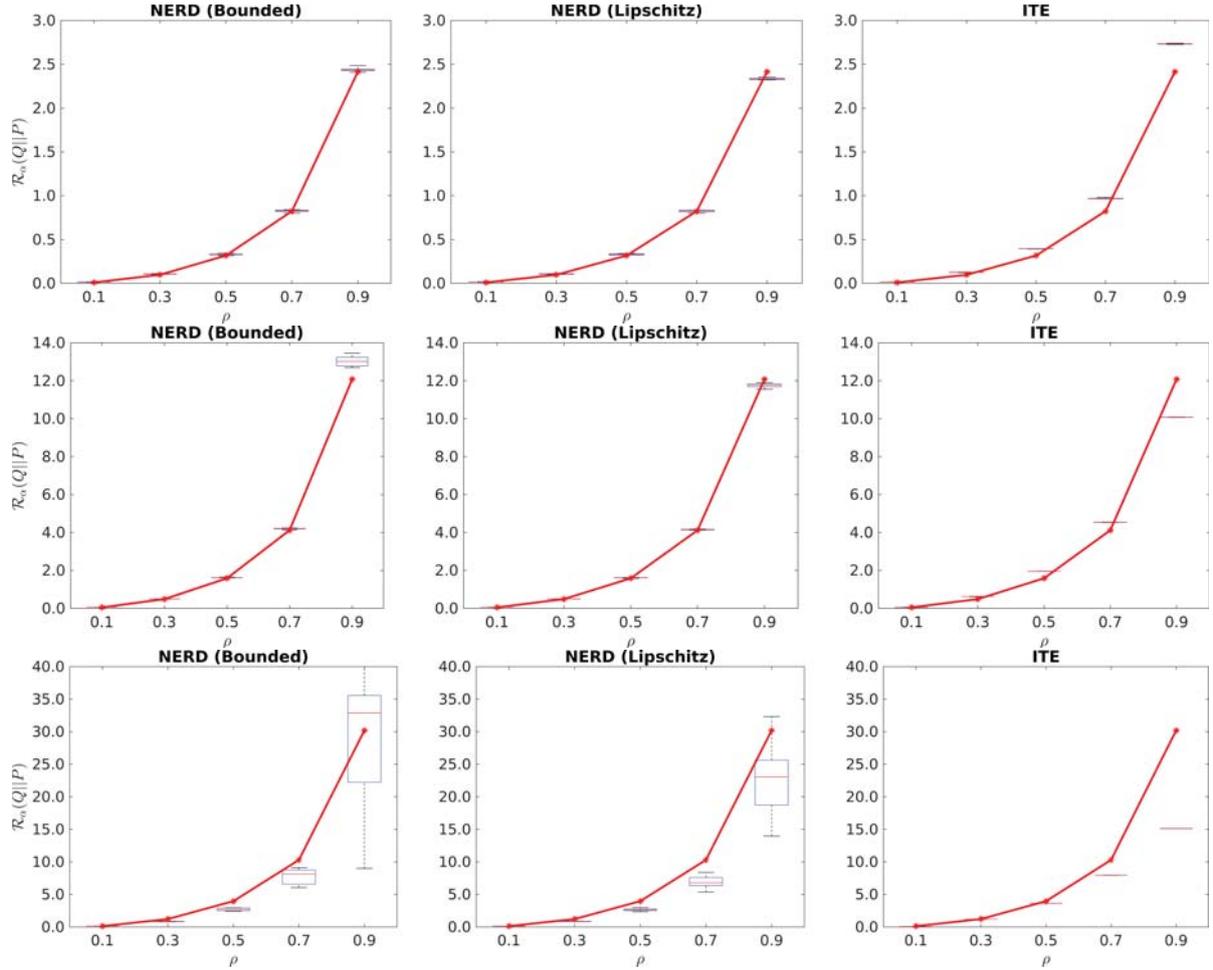


Figure 4: Rényi divergence between two multivariate Gaussian distributions as a function of the correlation coefficient ρ and the dimension d . The red solid line represents the exact Rényi divergence, whereas each boxplot corresponds to the 2nd and 3rd quantiles of the estimator over 10 independent iterations. In all cases, we set $M = 50$ and $K = 5$. *Upper panels:* We compare two $d = 4$ -dimensional Gaussians and draw $N = 50000$ samples for each. Both NERD variations are close to the exact Rényi divergence value, whereas ITE slightly overestimates it. *Middle panels:* With dimension being $d = 20$ and sample size being $N = 150000$, NERD with Lipschitz continuous functions provide the most accurate estimates relative to the others. $N_{it} = 20k$. *Lower panels:* For $d = 50$ and $N = 300000$ samples, the variance for both NERD variations is high and increases as the correlation coefficient increases. ITE-based estimator has no variability but its estimate is entirely inaccurate.

Figure 4 presents the estimation results for the three methods considered in this paper as a function of dimension and correlation coefficient ρ . We consider three values for the dimension: $d = 4$ (upper row of panels), $d = 20$ (middle row of panels), and $d = 50$ (lower row of panels) while we range $\rho \in [0.1, 0.9]$. When $\rho < 0.5$ our results indicate that all three methods provide satisfactory divergence estimations. When ρ increases, both NERD variants accurately estimate the Rényi divergence for $d \leq 20$ with small variance. In contrast, the variance increases significantly as ρ increases (lower panels) revealing that even larger sample size is required. This finding is in partial agreement with the results in [33] where it is shown that variance of this variational-based estimators may grow exponentially with the value of the Rényi divergence. On the other hand, we found that the NERD_{Lip} estimator has less variance relative to NERD_{C_b} especially for $\rho = 0.9$ revealing that the restriction of the function space to Lipschitz continuous functions as presented in Section 3.2 is beneficial from a statistical estimation perspective.

Finally, despite having very low variance in all cases, ITE-based estimation is inaccurate for large values of ρ even for $d = 4$.

Concerning the computational cost in CPU time, the most important factors are the sample size N and the dimension d . The training of neural network’s parameters scales linearly with N and in most cases with d too while ITE approach which is based on k -nearest neighbors does not scale efficiently neither with N nor with d . Despite being architecture and dimension dependent, we advocate that the break even point in terms of computational cost between the NERD and ITE approaches is for sample size N approximately between $5 \cdot 10^4$ and 10^5 . We also remark that NERD_{Lip} is approximately twice as slow as NERD_{C_b} due to the additional computational cost induced by the regularization term.

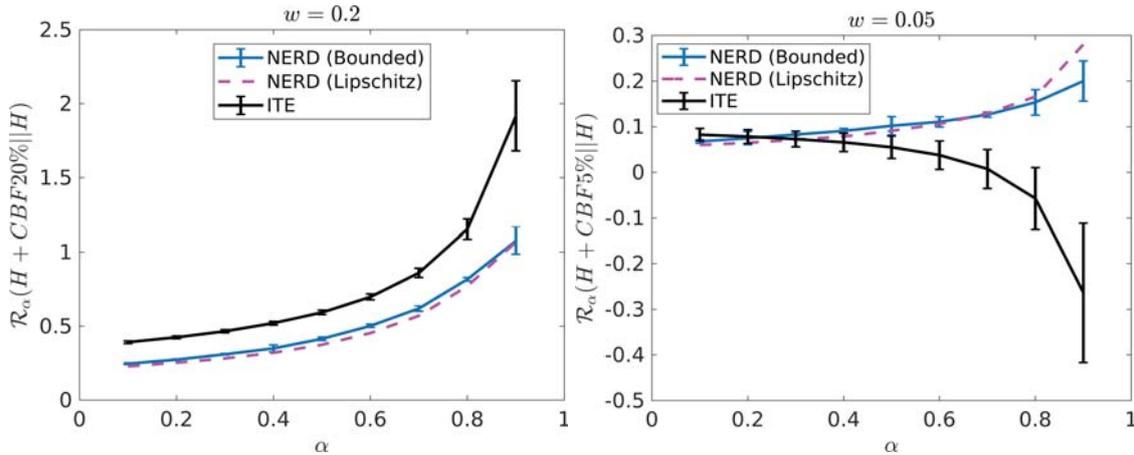


Figure 5: Rényi divergence estimates of NERD_{Lip} ($K = 1$), NERD_{C_b} and ITE when comparing healthy against disease-contaminated distributions, over varying α when the abundance of disease samples is set to $w = 0.2$ (left panel) and $w = 0.05$ (right panel). Errorbars are computed over 5 iterations. Both NERD variants generate similar consistent results while ITE estimates have shift and scaling issues making it untrustworthy.

4.3 Detecting Sub-populations in Single-Cell Datasets

Using data from [36], we test the efficacy of NERD to discriminate distributions from real biological settings⁵. Specifically, we consider single cell mass cytometry measurements on 16 bone marrow protein markers ($d=16$) coming from healthy and disease individuals with acute myeloid leukemia. The dataset consists of more than 150K healthy and 25K disease cell samples. Before analysis, data were transformed using the inverse hyperbolic sine $\text{arcsinh}()$ transformation with a cofactor of 5, which is typical in order to have comparable supports across dimensions. Following [8] we mix healthy and disease samples at decreasing frequencies. For this, we first split the healthy samples randomly into two equally sized subsets X and Y . Then, we replace a predefined percentage of samples in X with disease samples; that is, $\{20\%, 5\%, 1\%, 0.5\% \text{ and } 0.2\%\}$ of cells. The resulting distributions Q_X and P_Y reflect the properties of settings where rare, disease-associated cell populations must be detected from otherwise healthy samples.

Figure 5 shows the Rényi divergence estimates for various $\alpha \in (0, 1)$ and two values for the sick cells percentage. Both NERD variants and ITE estimate positive values for the divergence thus they do discriminate the healthy distribution P_X from Q_Y when $w = 0.2$ (left panel). When

⁵Data were accessed from <https://community.cytobank.org/cytobank/experiments/46098/illustrations/121588>

$w = 0.05$ (right panel), NERD algorithm continues to produce positive values and be able to discriminate between the two distributions, however, ITE estimates are negative for several values of α . It seems that ITE approach has shift and scaling issues. Those estimation errors can be partially reduced by increasing k at the cost of significantly higher computational effort. We additionally remark that the curve of the NERD-estimated Rényi divergence as a function of its order is in accordance with Figure 2(c)-(d) in the sense that as α increases the value of the divergence does also increase. Even though the distributions of the biological data are not normal, this consistency in the behavior of Rényi divergence suggests that NERD algorithm correctly estimates the divergence value.

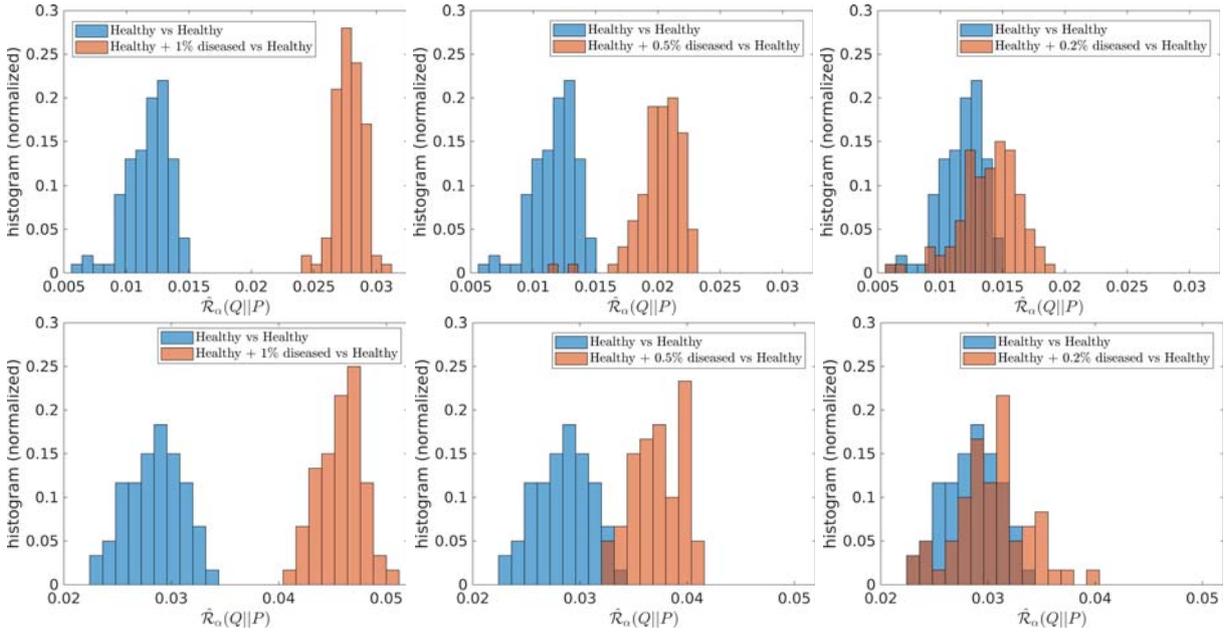


Figure 6: Histograms of repeated NERD estimates (60 runs) when trying to discriminate between two healthy datasets $\hat{\mathcal{R}}_\alpha(P_Y||P_Y)$ (blue color) and between a healthy dataset and a dataset contaminated with sick sub-population with proportion $w\%$: $\hat{\mathcal{R}}_\alpha(Q_X||P_Y)$ (orange color). We consider two sample sizes: $N = 78797$ samples per distribution (upper panels) and $\frac{N}{2}$ samples (lower panels). As the rare sub-population of sick cells decreases in numbers, it is harder to discriminate between the healthy and diseased distributions.

Finally, we investigate the limits of NERD algorithm in discriminating between healthy and sick cell contaminated distributions. Apparently, as w approaches 0, it becomes more difficult to distinguish the two distributions because the number of sick cells amounts to few dozens. Similarly, sample size is an important factor in order to generate statistically significant outcomes. Figure 6 presents histograms of repeated estimates of Rényi divergence with $\alpha = 0.5$ computed via NERD_{Lip} algorithm for the healthy vs healthy case (blue color) and diseased vs healthy case (orange color). In the upper row of panels, we use all available data while the sample size is halved in the lower row of panels. As it is evident from the x-axis, the estimates of Rényi divergence for the healthy vs healthy case is doubled for the lower panels revealing that sample size is indeed a crucial factor for accurately estimating the divergence which is zero for this case. Moreover, histograms are separated for values of w above 0.005 for both sample sizes. On the other hand, there is overlap of the histograms when $w = 0.002$ especially for the halved sample size. This is also evident from the Kolmogorov-Smirnov (KS) test computed on the histograms. Table 2 reports the p-value and the statistic of the KS test using build-in

function `kstest2`. Overall, we conclude that sick cell proportion below $w = 0.002$ cannot be detected with NERD for $\alpha = 0.5$ and sample size below $N = 40000$. Nevertheless, when we increase α to 0.8, differences between the histograms start to emerge (last row in Table 2) showing that larger values for α could assist in discriminating even rarer sub-populations.

Table 2: p-values and statistic for the KS test. Apart from one case, KS test suggests that the two distributions are different.

α	w	p-value	KS stat	samples
$\alpha = 0.5$	1%	1.5×10^{-45}	1	$N \approx 79K$
$\alpha = 0.5$	0.5%	9.4×10^{-44}	0.98	N
$\alpha = 0.5$	0.2%	3.6×10^{-12}	0.51	N
$\alpha = 0.5$	1%	7.8×10^{-28}	1	$N/2$
$\alpha = 0.5$	0.5%	4.9×10^{-26}	0.96	$N/2$
$\alpha = 0.5$	0.2%	0.0068	0.3	$N/2$
$\alpha = 0.8$	0.2%	1.4×10^{-5}	0.433	$N/2$

5 CONCLUSIONS

In this paper, we propose an efficient discrimination approach based on Rényi divergence to quantify the difference between population datasets and answer whether or not two sample population come from the same distribution. The estimation of Rényi divergence is performed via the optimization of functions parametrized by neural networks. We investigated the performance of the presented algorithm (NERD) on several synthetic and real biological datasets. We showed that both NERD variants accurately estimate the Rényi divergence and discriminate rare sub-populations in the data given sufficiently-large number of samples. Therefore, its potential to be used as a screening and/or detection tool in single-cell applications is high. As future work, we target towards devising novel techniques that reduce the estimator’s variance and require smaller sample sizes.

REFERENCES

- [1] Sean C. Bendall, Erin F. Simonds, Peng Qiu, El-ad D. Amir, Peter O. Krutzik, Rachel Finck, Robert V. Bruggner, Rachel Melamed, Angelica Trejo, Olga I. Ornatsky, Robert S. Balderas, Sylvia K. Plevritis, Karen Sachs, Dana Pe’er, Scott D. Tanner, and Garry P. Nolan. Single-cell mass cytometry of differential immune and drug responses across a human hematopoietic continuum. *Science*, 332(6030):687–696, 2011.
- [2] Smita Krishnaswamy, Matthew H. Spitzer, Michael Mingueneau, Sean C. Bendall, Oren Litvin, Erica Stone, Dana Pe’er, and Garry P. Nolan. Conditional density-based analysis of t cell signaling in single-cell data. *Science*, 346(6213), 2014.
- [3] Valery Tereshko. Reaction-diffusion model of a honeybee colony’s foraging behaviour. In Marc Schoenauer, Kalyanmoy Deb, Günther Rudolph, Xin Yao, Evelyne Lutton, Juan Julian Merelo, and Hans-Paul Schwefel, editors, *Parallel Problem Solving from Nature PPSN VI*, pages 807–816, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.
- [4] Sebastian Anita and Vincenzo Capasso. Reaction-diffusion systems in epidemiology, 2017.

- [5] Eugenio Marco, Robert L. Karp, Guoji Guo, Paul Robson, Adam H. Hart, Lorenzo Trippa, and Guo-Cheng Yuan. Bifurcation analysis of single-cell gene expression data reveals epigenetic landscape. *Proceedings of the National Academy of Sciences*, 2014.
- [6] Hong Zhan, Ramunas Stanciasauskas, Christian Stigloher, Kevin K Dizon, Maelle Jospin, Jean-Louis Bessereau, and Fabien Pinaud. In vivo single-molecule imaging identifies altered dynamics of calcium channels in dystrophin-mutant *c. elegans*. *Nature communications*, 5:4974, 2014.
- [7] Laura de Vargas Roditi and Manfred Claassen. Computational and experimental single cell biology techniques for the definition of cell type heterogeneity, interplay and intracellular dynamics. *Current Opinion in Biotechnology*, 34:9–15, 2015. Systems biology • Nanobiotechnology.
- [8] Lukas M Weber, Malgorzata Nowicka, Charlotte Soneson, and Mark D. Robinson. diffeyt: Differential discovery in high-dimensional cytometry via high-resolution clustering. *Communications Biology*, 2(183), 2019.
- [9] Tim van Erven and Peter Harremoës. Rényi divergence and Kullback-Leibler divergence. *IEEE Transactions on Information Theory*, 60(7):3797–3820, 2014.
- [10] Venkat Anantharam. A variational characterization of rényi divergences. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pages 893–897, 2017.
- [11] Jeremiah Birrell, Paul Dupuis, Markos A. Katsoulakis, Luc Rey-Bellet, and Jie Wang. Variational Representations and Neural Network Estimation for Rényi Divergences. *arXiv:2007.03814*, 2020.
- [12] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual information neural estimation. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 531–540, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.
- [13] XuanLong Nguyen, Martin J. Wainwright, and Michael I. Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.
- [14] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of Wasserstein GANs. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 5767–5777, 2017.
- [15] Yannis Pantazis, Dipjyoti Paul, Michail Fasoulakis, Yannis Stylianou, and Markos A. Katsoulakis. Cumulant gan. *arXiv:2006.06625v2*, 2020.
- [16] Barnabás Póczos, Zoltán Szabó, and Jeff Schneider. Nonparametric divergence estimators for independent subspace analysis. In *2011 19th European Signal Processing Conference*, pages 1718–1722, 2011.
- [17] Friedrich Liese and Igor Vajda. *Convex statistical distances*, volume 95 of *Teubner-Texte zur Mathematik [Teubner Texts in Mathematics]*. BSB B. G. Teubner Verlagsgesellschaft, Leipzig, 1987. With German, French and Russian summaries.
- [18] Rami Atar, Kamaljit Chowdhary, and Paul Dupuis. Robust bounds on risk-sensitive functionals via rényi divergence. *SIAM/ASA Journal on Uncertainty Quantification*, 3:18–33, 2015.
- [19] Alfréd Rényi. On measures of entropy and information. In *Proc. of the 4th Berkeley Symposium on Mathematical Statistics and Probability*, volume I, pages 547–561, Berkeley, CA, 1961. University of California Press.

- [20] Igor Vajda. Distances and discrimination rates for stochastic processes. *Stochastic Processes and Applications*, 35:47–57, 1990.
- [21] Tom Minka. Divergence measures and message passing. Technical Report MSR-TR-2005-173, January 2005.
- [22] Alexandre B Tsybakov. Introduction to nonparametric estimation. *Springer Science & Business Media*, 2008.
- [23] Leila Golshani, Einollah Pasha, and Gholamhossein Yari. Some properties of Rényi entropy and Rényi entropy rate. *Information Sciences*, 179:2426–2433, 2009.
- [24] Paul Dupuis, Markos A. Katsoulakis, Yannis Pantazis, and Luc Rey-Bellet. Sensitivity analysis for rare events based on Rényi divergence. *Ann. Appl. Probab.*, 30(4):1507–1533, 08 2020.
- [25] Rami Atar, Amarjit Budhiraja, Paul Dupuis, and Ruoyu Wu. Robust bounds and optimization at the large deviations scale for queueing models via rényi divergence, 2020.
- [26] Manuel Gil, Fady Alajaji, and Tamas Linder. Rényi divergence measures for commonly used univariate continuous distributions. *Information Sciences*, 249(Complete):124–131, 2013.
- [27] Monroe D. Donsker and S. R. Srinivasa Varadhan. Asymptotic evaluation of certain markov process expectations for large time. iv. *Communications on Pure and Applied Mathematics*, 36(2):183–212, 1983.
- [28] Gerald B. Folland. *Real analysis: Modern Techniques and Their Applications*. Wiley, New York, 1999.
- [29] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359 – 366, 1989.
- [30] Zhou Lu, Hongming Pu, Feicheng Wang, Zhiqiang Hu, and Liwei Wang. The expressive power of neural networks: A view from the width. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17*, page 6232–6240, Red Hook, NY, USA, 2017. Curran Associates Inc.
- [31] Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5171–5180. PMLR, 09–15 Jun 2019.
- [32] David McAllester and Karl Stratos. Formal limitations on the measurement of mutual information. In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 875–884. PMLR, 26–28 Aug 2020.
- [33] Jiaming Song and Stefano Ermon. Understanding the limitations of variational mutual information estimators. In *International Conference on Learning Representations*, 2020.
- [34] Jeremiah Birrell, Paul Dupuis, Markos A. Katsoulakis, Yannis Pantazis, and Luc Rey-Bellet. (f, γ) -divergences: Interpolating between f -divergences and integral probability metrics, 2021.
- [35] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *The International Conference on Learning Representations (ICLR)*, 2015.

- [36] Jacob H. Levine, Erin F. Simonds, Sean C. Bendall, Kara L. Davis, El Ad D. Amir, Michelle D. Tadmor, Oren Litvin, Harris G. Fienberg, Astraea Jager, Eli R. Zunder, Rachel Finck, Amanda L. Gedman, Ina Radtke, James R. Downing, Dana Pe'er, and Garry P. Nolan. Data-Driven Phenotypic Dissection of AML Reveals Progenitor-like Cells that Correlate with Prognosis. *Cell*, 162(1):184–197, 2015.

IDENTIFICATION OF UNKNOWN PARAMETERS AND PREDICTION WITH HIERARCHICAL MATRICES

A. Litvinenko^{1*}, R. Kriemann², and V. Berikov^{3,4}

¹ RWTH Aachen, Aachen, Germany
e-mail: litvinenko@uq.rwth-aachen.de

² Max Planck Institute for Mathematics in the Sciences (MiS) in Leipzig
e-mail: rok@mis.mpg.de

³Sobolev Institute of Mathematics, Novosibirsk, Russia
e-mail: berikov@math.nsc.ru

⁴Novosibirsk State University, Novosibirsk, Russia

Keywords: Computational statistics, parameter inference, prediction, hierarchical matrix, data analysis, Matérn covariance, random field, spatial statistics

Abstract. *Statistical analysis of massive datasets very often implies expensive linear algebra operations with large dense matrices. Typical tasks are an estimation of unknown parameters of the underlying statistical model and prediction of missing values. We developed the \mathcal{H} -MLE procedure, which solves these typical tasks. The unknown parameters can be estimated by maximizing the joint Gaussian log-likelihood function, which depends on a covariance matrix. To decrease the high computational cost, we approximate the covariance matrix in the hierarchical (\mathcal{H} -) matrix format, which has only a log-linear computational cost. The \mathcal{H} -matrix technique allows inhomogeneous covariance matrices and almost arbitrary locations. Especially, \mathcal{H} -matrices can be applied in cases when the matrices under consideration are dense and unstructured.*

For validation purposes, we implemented three machine learning methods: the kNN, random forest, and deep neural network. The best results (for the given datasets) were obtained by the kNN method with three or seven neighbors depending on the dataset. The results computed with the \mathcal{H} -MLE method were compared with the results obtained by the kNN method.

The developed \mathcal{H} -matrix code and all datasets are freely available online.

Contents

1	Introduction	2
2	\mathcal{H}-matrix approximation of covariance matrices and the log-likelihood	4
3	Prediction Errors	5
4	Machine learning methods to make predictions	5
5	Numerical results, obtained by the \mathcal{H}-MLE method	6
5.1	Tests 1a and 1b: Parameter identification and prediction, 8 datasets with $n = 90,000 + 10,000$	7
5.2	Test-2a: Parameter identification and prediction, $n = 90,000 + 10,000$	8
5.3	Test-2b: Parameter identification and prediction, $n = 900,000 + 100,000$	9
6	Numerical results, obtained by the machine learning methods	11
7	Conclusion	12
1	Introduction	

The number of measurements that should be statistically analyzed increases from year to year. The involved statistical methods often contain intensive operations with large dense matrices. In case these measurements are distributed irregularly across the given domain, the efficient algorithms like the Fast Fourier Transformation and similar are not applicable. Therefore, new efficient methods are needed.

To make these expensive computations possible, we suggest to use the hierarchical matrix (\mathcal{H} -matrix) technique [19, 18, 29, 26]. We will demonstrate how to solve statistical inference and prediction tasks appearing very often in spatial statistics. This work is an extension of our previous research [32, 31]. The first novelty is that we simultaneously identify four unknown parameters and not three. This new parameter is the regularization term - the nugget τ^2 . Another novelty is the new research on how well we can make statistical predictions with \mathcal{H} -matrix approximations. Finally, we compare our identified parameters and predicted values with the actual results and with results obtained by other methods. We summarize the strong and weak sides of the \mathcal{H} -matrix technique.

Assumptions. Let (s_1, \dots, s_n) be the set of locations. We model the set of measurements as a realization from a stationary Gaussian spatial random field. Specifically, we let $\mathbf{Z} = \{Z(s_1), \dots, Z(s_n)\}^\top$, where $Z(s)$ is a Gaussian random field indexed by a spatial location $s \in \mathbb{R}^d$, $d = 2, 3$. Then, we assume that \mathbf{Z} has zero mean and a stationary parametric covariance function $C(\mathbf{h}; \boldsymbol{\theta}) = \text{cov}\{Z(s), Z(s + \mathbf{h})\}$, where $\mathbf{h} \in \mathbb{R}^d$ is a spatial lag vector and $\boldsymbol{\theta} \in \mathbb{R}^4$ the unknown parameter vector of interest. Statistical inferences about $\boldsymbol{\theta}$ are often based on the Gaussian log-likelihood function:

$$\mathcal{L}(\boldsymbol{\theta}) = -\frac{n}{2} \log(2\pi) - \frac{1}{2} \log |\mathbf{C}(\boldsymbol{\theta})| - \frac{1}{2} \mathbf{Z}^\top \mathbf{C}(\boldsymbol{\theta})^{-1} \mathbf{Z}, \tag{1}$$

where the covariance matrix $\mathbf{C}(\boldsymbol{\theta})$ has entries $C(s_i - s_j; \boldsymbol{\theta})$, $i, j = 1, \dots, n$. The maximum likelihood estimator of $\boldsymbol{\theta}$ is the value $\hat{\boldsymbol{\theta}}$ that maximizes (1). When the sample size n is large, the evaluation of (1) becomes challenging. Indeed, the storage of the n -by- n covariance matrix \mathbf{C}

requires $\mathcal{O}(n^2)$ units of memory. Computation of the inverse and log-determinant of $\mathbf{C}(\boldsymbol{\theta})$ cost $\mathcal{O}(n^3)$ FLOPs. Hence, parallel and scalable methods that can reduce this high cost are needed.

Similar works for the case when measurements are located on a rectangular grid can be resolved via the fast Fourier transformation (FFT) method [47, 9, 16, 45, 11] with the computing cost $\mathcal{O}(n \log n)$. However, the FFT method does not work for data measured at irregularly spaced locations or requires expensive, non-trivial modifications.

Other recent ideas include the low-tensor rank methods [30, 36, 22], covariance tapering [14, 24, 42, 43], likelihood approximations [44, 12], Gaussian Markov random-field approximations [13], Vecchia framework [46, 23], the nearest-neighbor Gaussian process models [10], the low-rank update [40], multiresolution Gaussian process models [37], equivalent kriging [27], and Bayesian-like approach [34, 35].

An \mathcal{H} -matrix approximation of covariance matrices was done in [28, 41, 4, 20, 3, 6, 39]. The inverse of the covariance matrix was approximated in [2, 3, 5]. The \mathcal{H} -matrix technique for the parameter estimation was proposed in [3, 2]. There are many implementations of \mathcal{H} -matrices exist: HLIB (<http://www.hlib.org/>), \mathcal{H}^2 (<https://github.com/H2Lib>), HLIBPro (<https://www.hlibpro.com/>), and some others. In this work, we are using the HLIBPro library. For extended details, we refer to our earlier works [31, 32]. The data, which we used in this work were generated in the ExaGeoStat library [1] (<https://github.com/ecrc/exageostat>) without using \mathcal{H} -matrices.

Matérn covariance functions: We consider the Matérn family [33], which has gained widespread interest in recent years [17]. The Matérn covariance depends only on the distance $\mathbf{h} := \|\mathbf{s} - \mathbf{s}'\|$, where \mathbf{s} and \mathbf{s}' are any two spatial locations:

$$C(\mathbf{h}; \boldsymbol{\theta}) = \frac{\sigma^2}{2^{\nu-1}\Gamma(\nu)} \left(\frac{h}{\ell}\right)^\nu K_\nu\left(\frac{h}{\ell}\right) + \tau^2 \mathbf{I}, \quad (2)$$

with parameters $\boldsymbol{\theta} = (\sigma, \ell, \nu, \tau)^\top$. Here σ^2 is the variance, τ^2 the nugget, $\nu > 0$ controls the smoothness of the random field, with larger values of ν corresponding to smoother fields, and $\ell > 0$ the spatial range parameter that measures how quickly the correlation of the random field decays with distance. A larger ℓ corresponds to a faster decay. \mathcal{K}_ν denotes the modified Bessel function of the second kind of order ν .

Prediction: Estimating the unknown parameters $\boldsymbol{\theta}$ is only an intermediate step. Once it is done, the estimation $\hat{\boldsymbol{\theta}} \approx \boldsymbol{\theta}$ is used for prediction at new locations. Let $I_1 := (\mathbf{s}_1, \dots, \mathbf{s}_n)$ be locations with known values \mathbf{Z}_1 , and $I_2 = (\mathbf{s}_{n+1}, \dots, \mathbf{s}_{n+m})$ be the new locations with unknown values $\mathbf{Z}_2 = \{Z(\mathbf{s}_{n+1}), \dots, Z(\mathbf{s}_{n+m})\}^\top$ to be predicted. Here $(Z(\mathbf{s}_1), \dots, Z(\mathbf{s}_n), Z(\mathbf{s}_{n+1}), \dots, Z(\mathbf{s}_{n+m}))^\top$ is a Gaussian random field indexed by spatial locations with indices from the index set (I_1, I_2) . We assume that vector $(\mathbf{Z}_1, \mathbf{Z}_2)$ is zero mean and has a stationary parametric covariance function. After discretisation we can get the following block covariance matrix

$$\begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix}, \quad (3)$$

where $\mathbf{C}_{11} \in \mathbb{R}^{n_1 \times n_1}$, $\mathbf{C}_{12} \in \mathbb{R}^{n_1 \times n_2}$, $\mathbf{C}_{21} \in \mathbb{R}^{n_2 \times n_1}$, and $\mathbf{C}_{22} \in \mathbb{R}^{n_2 \times n_2}$. Now, the unknown vector \mathbf{Z}_2 can be computed by the following formula [8]

$$\mathbf{Z}_2 = \mathbf{C}_{21} \mathbf{C}_{11}^{-1} \mathbf{Z}_1. \quad (4)$$

We can also say that \mathbf{Z}_2 has the conditional distribution with the mean value $\mathbf{C}_{21} \mathbf{C}_{11}^{-1} \mathbf{Z}_1$ and the covariance matrix $\mathbf{C}_{22} - \mathbf{C}_{21} \mathbf{C}_{11}^{-1} \mathbf{C}_{12}$.

2 \mathcal{H} -matrix approximation of covariance matrices and the log-likelihood

The \mathcal{H} -matrix technique is defined as a recursive partitioning of a given matrix into sub-blocks. The majority of these sub-blocks are approximated by low-rank matrices on the fly (without computing any dense sub-matrices). And only a minor number of sub-blocks are calculated as dense matrices without any approximation. Details about block partitioning and heuristic algorithms used for low-rank approximation are not so trivial. Therefore, we skip them here and refer to [26, 32, 31].

The \mathcal{H} -matrix approximation error depends on the type of the covariance matrix, its smoothness, covariance length, computational geometry, nugget, and the dimensionality of the problem. For some matrices, the problem may become ill-posed since even tiny perturbations in the covariance matrix $\mathbf{C}(\boldsymbol{\theta})$ may result in considerable perturbations in the log-determinant and the log-likelihood. The usage of $\tau^2\mathbf{I}$ regularisation helps partially to resolve this issue.

Storage and complexity. We let $\mathbf{C}(\boldsymbol{\theta}) \in \mathbb{R}^{n \times n}$ be approximated by an \mathcal{H} -matrix $\tilde{\mathbf{C}}(\boldsymbol{\theta}; k)$ or $\tilde{\mathbf{C}}(\boldsymbol{\theta}; \varepsilon)$. In the first case we fix the maximal rank k in each sub-block (the approximation accuracy will vary from sub-block to sub-block). In the second case we fix the accuracy ε in each sub-block (the ranks of sub-blocks will vary). The \mathcal{H} -Cholesky decomposition of $\tilde{\mathbf{C}}(\boldsymbol{\theta}; k)$ costs $\mathcal{O}(k^2 n \log^2 n)$. The solution of the linear system $\tilde{\mathbf{L}}(\boldsymbol{\theta}; k)\mathbf{v}(\boldsymbol{\theta}) = \mathbf{Z}$ costs $\mathcal{O}(k^2 n \log^2 n)$. The log-determinant $\log |\tilde{\mathbf{C}}(\boldsymbol{\theta}; k)| = 2 \sum_{i=1}^n \log \{\tilde{L}_{ii}(\boldsymbol{\theta}; k)\}$ is available for free. The cost of computing the log-likelihood function $\tilde{\mathcal{L}}(\boldsymbol{\theta}; k)$ is $\mathcal{O}(k^2 n \log^2 n)$ and the cost of computing the MLE $\hat{\boldsymbol{\theta}}$ in m iterations is $\mathcal{O}(mk^2 n \log^2 n)$.

Maximization of the log-likelihood. To maximize $\tilde{\mathcal{L}}(\boldsymbol{\theta}; k) \approx \mathcal{L}(\boldsymbol{\theta})$ we use the Brent-Dekker method [7, 38]. It is implemented in the GNU Scientific library <https://www.gnu.org/software/gsl/>. The Brent-Dekker algorithm first uses the fast-converging secant method or inverse quadratic interpolation to maximize $\tilde{\mathcal{L}}(\boldsymbol{\theta}; \cdot)$. If those do not work, then it returns to the more robust bisection method. In the following we will call this optimization procedure \mathcal{H} -MLE. It iteratively computes parameter $\boldsymbol{\theta}$ where the maximum of $\tilde{\mathcal{L}}(\boldsymbol{\theta}; \cdot)$ is achieved.

Additionally to the \mathcal{H} -Cholesky factorisation $\mathbf{C}(\boldsymbol{\theta}) = \mathbf{L}(\boldsymbol{\theta})\mathbf{L}(\boldsymbol{\theta})^\top$, we implemented a more stable factorisation $\mathbf{L}(\boldsymbol{\theta})\mathbf{D}(\boldsymbol{\theta})\mathbf{L}^\top(\boldsymbol{\theta})$, which avoids extracting square roots of diagonal elements. Both factorizations are connected via $\mathbf{LDL}^\top = (\mathbf{LD}^{1/2})(\mathbf{LD}^{1/2})^\top$. Very small negative diagonal elements can appear due to, e.g., the rounding off error.

The computation of $\boldsymbol{\theta}$ depends on the number of iterations in the optimization algorithm and the used threshold (10^{-4} in our experiments). The maximal number of iterations we used was 400. We may need more depending on the initial guess and the threshold. The running times are listed in Table 4.

\mathcal{H} -matrix approximation error analysis. For multiple numerical tests we refer to our earlier works [32, 31, 26]. There the reader can find numerical errors for \mathbf{C} , the Cholesky factor \mathbf{L} , and the log-likelihood \mathcal{L} . The \mathcal{H} -matrix approximation accuracy of the Cholesky factor and the inverse depends on the condition number of \mathbf{C} . The prediction accuracy can be estimated as follows

$$\begin{aligned} \|\mathbf{Z}_2 - \tilde{\mathbf{Z}}_2\| &= \|\mathbf{C}_{21}\mathbf{C}_{11}^{-1}\mathbf{Z}_1 - \tilde{\mathbf{C}}_{21}\tilde{\mathbf{C}}_{11}^{-1}\mathbf{Z}_1\| \\ &= \|\mathbf{C}_{21}\mathbf{C}_{11}^{-1}\mathbf{Z}_1 - \tilde{\mathbf{C}}_{21}\mathbf{C}_{11}^{-1}\mathbf{Z}_1 + \tilde{\mathbf{C}}_{21}\mathbf{C}_{11}^{-1}\mathbf{Z}_1 - \tilde{\mathbf{C}}_{21}\tilde{\mathbf{C}}_{11}^{-1}\mathbf{Z}_1\| \\ &\leq \|\mathbf{C}_{21}\mathbf{C}_{11}^{-1}\mathbf{Z}_1 - \tilde{\mathbf{C}}_{21}\mathbf{C}_{11}^{-1}\mathbf{Z}_1\| + \|\tilde{\mathbf{C}}_{21}\mathbf{C}_{11}^{-1}\mathbf{Z}_1 - \tilde{\mathbf{C}}_{21}\tilde{\mathbf{C}}_{11}^{-1}\mathbf{Z}_1\| \\ &\leq \|\mathbf{C}_{21} - \tilde{\mathbf{C}}_{21}\| \cdot \|\mathbf{C}_{11}^{-1}\| \cdot \|\mathbf{Z}_1\| + \|\tilde{\mathbf{C}}_{21}\| \cdot \|\mathbf{C}_{11}^{-1} - \tilde{\mathbf{C}}_{11}^{-1}\| \cdot \|\mathbf{Z}_1\| \end{aligned}$$

Now we see that the quality of the prediction depends on the quality of the \mathcal{H} -matrix approxi-

mation of matrices \mathbf{C}_{21} and \mathbf{C}_{11}^{-1} , i.e. the norms $\|\mathbf{C}_{21} - \tilde{\mathbf{C}}_{21}\|$ and $\|\mathbf{C}_{11}^{-1} - \tilde{\mathbf{C}}_{11}^{-1}\|$. In our earlier works [32, 31, 26], we demonstrated the error decay for \mathbf{C} and \mathbf{C}^{-1} .

3 Prediction Errors

We used the Mean Loss Efficiency (MLOE), the Mean Misspecification of the Mean Square Error (MMOM), and the Root Mean Square Error (RMSE) as in the 2021 KAUST Competition on Spatial Statistics for Large Datasets [21]:

$$\text{MLOE} := \frac{1}{M} \sum_{j=1}^M \left(\frac{\mathbb{E}_t \left((\hat{Z}_a(\mathbf{s}_j) - Z(\mathbf{s}_j))^2 \right)}{\mathbb{E}_t \left((\hat{Z}_t(\mathbf{s}_j) - Z(\mathbf{s}_j))^2 \right)} - 1 \right), \quad (5)$$

$$\text{MMOM} := \frac{1}{M} \sum_{j=1}^M \left(\frac{\mathbb{E}_a \left((\hat{Z}_a(\mathbf{s}_j) - Z(\mathbf{s}_j))^2 \right)}{\mathbb{E}_t \left((\hat{Z}_a(\mathbf{s}_j) - Z(\mathbf{s}_j))^2 \right)} - 1 \right). \quad (6)$$

Here $(\mathbf{s}_1, \dots, \mathbf{s}_M) := \mathcal{J}$ is a fixed subset of $M < n$ randomly-chosen locations. For numerical purposes M was chosen to be equal 1000. $\hat{Z}_t(\mathbf{s}_j)$ and $\hat{Z}_a(\mathbf{s}_j)$ are respectively kriging prediction at \mathbf{s}_j using the true and approximated model (plugging in the true parameters and estimated parameters in the covariance function), and $\mathbb{E}_t(\cdot)$ and $\mathbb{E}_a(\cdot)$ are respectively the expectation using the true and approximated model. We refer to [21] for more details.

MLOE gives us an understanding of the average loss of prediction efficiency when the approximated model is used to predict instead of the true model. MMOM presents the average misspecification of the mean square error when calculated under the approximated model.

The RMSE error was used to evaluate the prediction accuracy

$$\text{RMSE} = \sqrt{\frac{1}{n_t} \sum_{i=1}^{n_t} \left(\hat{Z}(\mathbf{s}_i) - Z(\mathbf{s}_i) \right)^2}, \quad (7)$$

where $\hat{Z}(\mathbf{s}_i)$ and $Z(\mathbf{s}_i)$ are respectively the predicted and true realization values at the location \mathbf{s}_i in the testing dataset, and n_t is the total number of locations in the testing dataset.

4 Machine learning methods to make predictions

Machine learning is aimed at building a model of data automatically from the observations. The obtained predictions can be considered as a baseline for comparison with other forecasting methods in which some additional information on the studied process is used. In the following, we tried three methods:

k-nearest neighbours (kNN): This method belongs to classical non-parametric family of statistical machine learning methods and follows a simple idea: for each data point x for which one needs to predict its output \hat{y} , find its k nearest neighbors x_1, \dots, x_k with respect to some metrics, and set $\hat{y} = \frac{1}{k} \sum_{i=1}^k y_i$, where y_i is the observed value of the response for point x_i . The value of k should be determined in the best way, for example, by cross-validation procedure or using an independent test sample for error estimation.

Random Forest (RF): Random Forest (RF) is another popular machine learning method in which a large number of decision (or regression) trees are generated independently on random sub-samples of data. The final decision for x is calculated over the ensemble of trees by

averaging the predicted outcomes. The method is theoretically well-substantiated and gives state-of-the-art results in many practical tasks, especially in the presence of many irrelevant features describing the observed data.

Deep Neural Network (DNN): Methods of this broad class are based on the artificial neural network paradigm, which models the functioning of neurons in the brain. In our study, we use a fully connected neural network (FCNN) which includes several fully connected layers, i.e., connecting each neuron in a layer to every neuron in the next layer (see an example in Figure 1). Mathematically speaking, the input feature vector transformation performed with each layer can be presented as a matrix-vector multiplication, where the matrix elements are neuron connection weights. Each layer is followed by a non-linear activation unit. FCNN training procedure consists of finding neurons' connection weights for which the quality metric takes the best value (usually by gradient descent technique).

Each ML method needs a fine-tuning stage to optimize its hyperparameters or architecture.

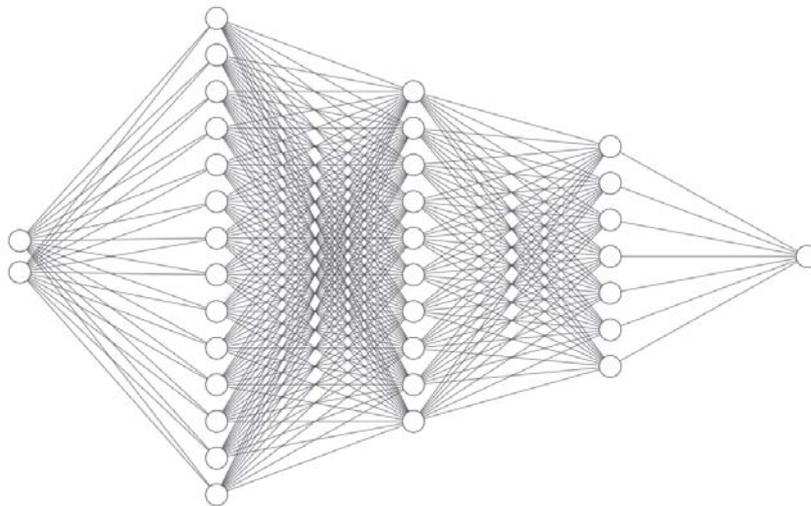


Figure 1: Example of FCNN architecture. The input layer consists of two neurons (input feature dimensionality), and the output layer consists of one neuron (predicted feature dimensionality). Three hidden layers with different numbers of neurons are presented.

The best value of k and the distance metric should be determined for kNN. For RF, one needs to set the number of trees in the ensemble, tree complexity, and splitting criterion. For FCNN, one needs to optimize the number of layers and neurons in each layer. For some other DNN (kNN and RF) parameters, we use default Matlab, Scikit-learn, or Tensorflow settings. For other hyperparameters (such as k for kNN, the number of trees for RF, or the number of layers and hidden units for FCNN), we minimized the root mean squared error (RMSE) metric using a validation sample repeatedly obtained by random sub-sampling of data in the proportion 1:9. We used a candidate set of k values in the interval $\{1, \dots, 20\}$, number of trees in the interval $\{100, \dots, 150\}$ and examined some variants of FCNN architecture for different number of hidden layers in the interval $[3, 10]$ (having 50-100 neurons in each hidden layer).

5 Numerical results, obtained by the \mathcal{H} -MLE method

Datasets: For all tests below we used datasets from the statistical competition [21]. Three hidden layers with different numbers of neurons The spatial domain is the unit square $\mathcal{D} := [0.0, 1.0] \times [0.0, 1.0]$. Each dataset includes 90% of training samples and 10% testing samples,

where prediction should be made.

In the following, we will perform the following numerical tests: 1a, 1b, 2a, and 2b.

Tests 1a, 1b, and 2a contain 100,000 locations, and Test 2b contains 1,000,000 locations. The training datasets and datasets for predictions are taken from [21].

Hardware: For the \mathcal{H} -MLE method we used a parallel cluster with two Intel Xeon Gold 6144 processors. Each processor has 8 cores (16 threads) with 3.5GHz and 384GB RAM in total.

Software: All \mathcal{H} -MLE numerical results are reproducible. We invite the reader to install HLIBPro-2.9 (from www.hlibpro.com), download our code from https://github.com/litvinen/large_random_fields.git and play with it.

Parameters identified in Test 1a are used for the prediction in Test 1b. Parameters identified in Tests 2a and 2b are used for prediction in Tests 2a and 2b, respectively.

After we identified all parameters and did all predictions, we uploaded all these data to the competition webpage[21] and the organisers of that competition computed for us the approximation errors. These errors are listed in Tables 1 and 5.

5.1 Tests 1a and 1b: Parameter identification and prediction, 8 datasets with $n = 90,000 + 10,000$

In the Test-1a, there are 16 given datasets from different zero-mean stationary isotropic Gaussian random fields with a Matérn covariance. The training dataset consists of 90,000 randomly distributed locations and associated observations at these locations. The task is to infer four unknown parameters of the Matérn covariance function shown in (2) for each dataset.

To avoid negative intermediate values for these parameters, in the following we assume that:

$$\sigma = \frac{2.0}{1.1^{\sigma_0}}, \quad \ell = \frac{1.0}{1.5^{\ell_0}}, \quad \nu = \frac{1.0}{1.2^{\nu_0}}, \quad \tau = \frac{1.0}{2.0^{\tau_0}},$$

where $\sigma_0, \ell_0, \nu_0, \tau_0$ the new parameters to be identified by the optimization algorithm. As the initial guess, we took $(\sigma_0, \ell_0, \nu_0, \tau_0) = (2, 2, 1, 15)$. If we saw that we were wrong with these values (too many iterations were needed), we rerun the optimization algorithm with some new values. The advantage of this “log”-representation is that the auxiliary values $\sigma_0, \ell_0, \nu_0, \tau_0$ are allowed to take negative values, whereas σ, ℓ, ν, τ not. Negative values may appear during iterations in the MLE optimization procedure.

Table 1 contains 8 solutions for 8 given datasets [21]. The 1st column contains the dataset index, columns 2,3,4 and 5 contain values of $(\sigma^2, \ell, \nu, \tau^2)$ respectively. The column 6,7,8, and 9 contain the true values $(\hat{\sigma}^2, \hat{\ell}, \hat{\nu}, \hat{\tau}^2)$ respectively. The columns 10 and 11 contain the MLOE and MMOM errors. The 12th columns contains the RMSE error (as defined in Eq. 7). One can see that in some rows the estimated parameter values are very close to the true values, but in some not. The reason is that the derivative of the log-likelihood function at the point $(\sigma, \ell, \nu, \tau)$ is almost zero (is equal to our threshold 10^{-4}), and our optimization algorithm indicates this point as the maximum. To improve the estimate, we should iterate longer. Later, in Test-1b, the estimated parameters are used for the prediction, and one can see that our predictions are reasonable.

One can see that for some datasets (e.g., 4,5,7), the MLOE and MMOM errors are large. We would not say that the \mathcal{H} -matrix method failed since the optimization algorithm’s accuracy to compute the MLE estimate was 10^{-4} . We think that the problem is ill-posed and contains multiple solutions, i.e., there are many points θ where the derivative of \mathcal{L} is almost zero. Here “almost” means smaller than 10^{-4} .

dataset	σ^2	ℓ	ν	τ^2	$\hat{\sigma}^2$	$\hat{\ell}$	$\hat{\nu}$	$\hat{\tau}^2$	MLOE	MMOM	RMSE
1	0.29	0.0106	2.471	2.5e-14	1.5	0.0175	2.3	0	2.2e-2	4.8e-1	4e-3
2	1.762	0.0223	1.501	1.1e-14	1.5	0.0211	1.5	0	1.8e-4	8.8e-2	2.4e-2
3	1.478	0.0305	0.600	1.0e-10	1.5	0.031	0.6	0	2.0e-6	8.0e-3	0.23
4	1.09	0.0176	1.522	7.0e-14	1.5	0.0526	2.3	0	1.8	3566	5.6e-4
5	0.95	0.0781	0.714	1.3e-13	1.5	0.0632	1.5	0	2.2e-1	100	5.4e-3
6	1.32	0.0826	0.601	1.22e-27	1.5	0.0928	0.6	0	5.4e-4	5.3e-2	0.12
7	2.38	0.4370	0.795	2.47e-8	1.5	0.1686	1.5	0	2.5e-1	164	2e-3
8	1.2	0.2043	0.601	4.1e-17	1.5	0.2475	0.6	0	2.8e-3	6.4e-2	6.6e-2

Table 1: \mathcal{H} -MLE method. Comparison of the identified (columns 2-5) and true parameters (columns 6-9).

Equation 4 was used to do prediction at the new 10,000 locations. Figure 2 (left and right) visualize datasets 4 and 7 (see the 4th and 7th rows in Table 1), where the estimated parameters are far away from the true values (see also the last three columns in Table 1). As we can see 90,000 given measurements (yellow points) and 10,000 predicted (blue points) for both datasets are very good aligned.

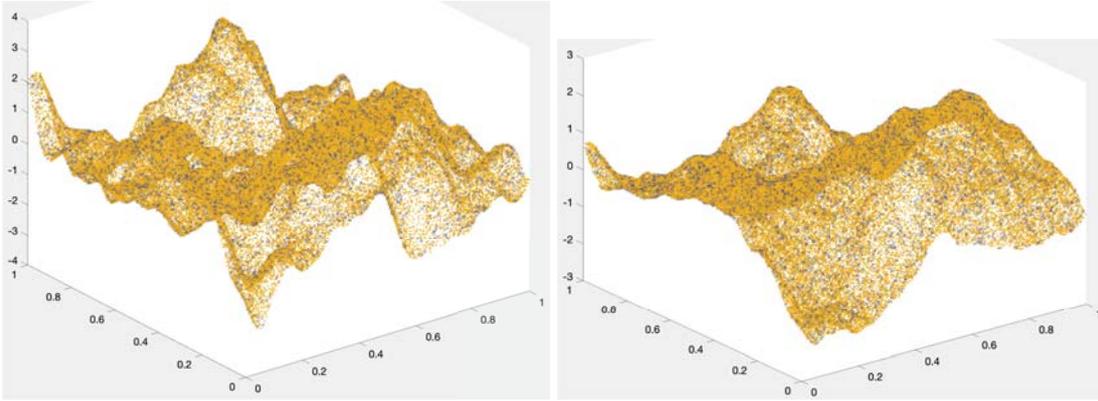


Figure 2: Test-1b, datasets 4 and 7: Prediction obtained by the \mathcal{H} -MLE method. The yellow points at 90,000 locations were used for training and the blue points were predicted in 10,000 locations. Although the identified parameters for these datasets were far away from the true values (see rows 4 and 7 in Table 1), one can still observe a very good alignment of yellow and blue points.

5.2 Test-2a: Parameter identification and prediction, $n = 90,000 + 10,000$

There are two datasets in this experiment. Each dataset contains 90,000 measurements to identify unknown parameters (training of the statistical model). Later this model is used to predict unknown values in new 10,000 locations. The identified parameters for both datasets are listed in Table 2. The columns 2,3,4,5 contain the values $(\sigma^2, \ell, \nu, \tau^2)$ respectively, the 6th column the value $\mathcal{L}(\sigma^2, \ell, \nu, \tau^2)$, and columns 7,8,9,10 contain the true values $(\hat{\sigma}^2, \hat{\ell}, \hat{\nu}, \hat{\tau}^2)$ respectively. These true values were obtained from organisers after the competition [21] finished.

Parameters ν and τ were identified well, but σ^2 and ℓ not. We see two possible reasons for this. The first reason is that the true model was not Gaussian. The second reason is the insufficient threshold 10^{-4} in the MLE optimization algorithm. Initially, the organizers did not provide any additional information about the utilized model. Here, we actually tested how the Gaussian model approximates the Tukey g-and-h random model [48]. Meaning, that both datasets in Task-2a were univariate non-Gaussian spatial datasets, which were generated by

the Tukey g -and- h random fields. These fields generalize Gaussian random fields $Z(\mathbf{s})$. The Tukey g -and- h random process $T(\mathbf{s})$ is defined by marginal transformation at each location \mathbf{s} as follows:

$$T(\mathbf{s}) := \xi + \omega \cdot \frac{\exp(g \cdot Z(\mathbf{s})) - 1}{g} \cdot \exp\left(\frac{hZ^2(\mathbf{s})}{2}\right), \quad (8)$$

where ξ and ω are the location and scale parameters. The parameter g defines the skewness and $h \geq 0$ the tail-heaviness. Two different pairs of (g, h) were chosen, which simulate medium and strong deviation from Gaussian random fields. These parameters ξ, ω, g, h used for generating both datasets are listed in Table 2.

data	σ^2	ℓ	ν	τ^2	\mathcal{L}	$\hat{\sigma}^2$	$\hat{\ell}$	$\hat{\nu}$	$\hat{\tau}^2$	ξ	ω	g	h
1	7.7	0.07	1.037	$1.6e - 15$	$9.6e + 4$	1	0.1	1	0	1	2	0.2	0.2
2	31	0.047	1.066	$4.0e - 14$	$0.5e + 4$	1	0.1	1	0	1	2	0.5	0.3

Table 2: \mathcal{H} -MLE method. Comparison of the obtained parameter values with the true values for Test-2a, $n = 90,000$.

Figures 3 (left and right) show predictions obtained by the \mathcal{H} -MLE method. The yellow points at 90,000 locations were used for training and the blue points were predicted in 10,000 new locations. One can see a very good alignment of yellow and blue points on both pictures.

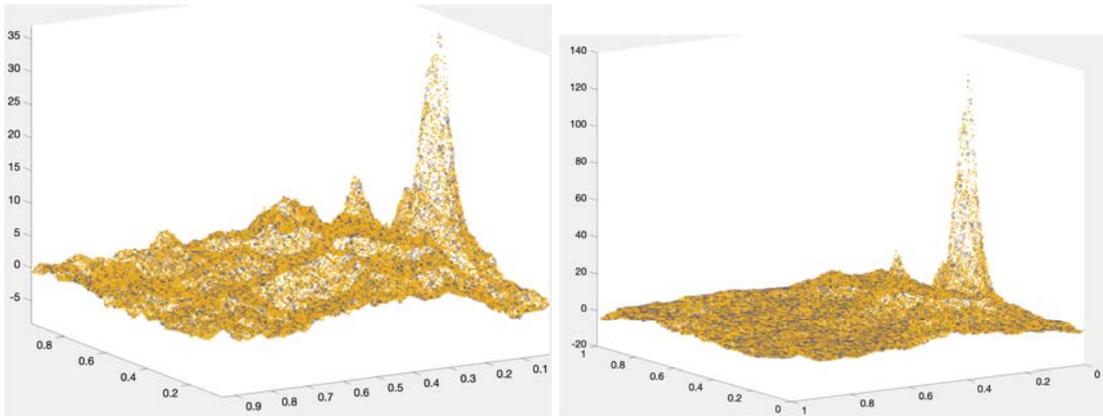


Figure 3: Test-2a, datasets 1 and 2: Prediction obtained by the \mathcal{H} -MLE method. The yellow points at 90,000 locations were used for training and the blue points were predicted in 10,000 locations. One can see a very good alignment of yellow and blue points on both figures.

5.3 Test-2b: Parameter identification and prediction, $n = 900,000 + 100,000$

There are two datasets in this experiment. Each dataset contains 900,000 measurements to identify unknown parameters (training of the statistical model). Later this model is used to predict unknown values in new 100,000 locations. The identified parameters are listed in Table 3 for two datasets. The columns 2,3,4,5 contain the values $(\sigma^2, \ell, \nu, \tau^2)$ respectively, the 6th column the value $\mathcal{L}(\sigma^2, \ell, \nu, \tau^2)$, and columns 7,8,9,10 contain the true values $(\hat{\sigma}^2, \hat{\ell}, \hat{\nu}, \hat{\tau}^2)$ respectively. These true values were obtained from organisers after the competition [21] finished.

Table 4 summarizes the computational times for the \mathcal{H} -MLE method. We note that this computing time varies a lot for the parameter identification task because it depends on the

data	σ^2	l	ν	τ^2	\mathcal{L}	$\hat{\sigma}^2$	\hat{l}	$\hat{\nu}$	$\hat{\tau}^2$
1	3.72	1.143830	0.94636	4.4e-3	3.6e+5	1.5	0.0632	1.5	0
2	0.92	0.012496	1.30867	8.5e-9	1.8e+6	1	0.1	1	0

Table 3: \mathcal{H} -MLE method. Comparison of the obtained parameter values with the true values for Test-2b, $n = 900,000$.

initial guess in the optimization algorithm. If the initial guess lies very close to the (unknown) true value of θ , then only a few iterations are needed. If not, then a few hundred iterations and the computing time may increase by a factor of 10.

Datasets/ Tasks	1a param. infer.	1b pred.	2a param. infer., pred.	2b param. infer., pred.
\mathcal{H} -MLE comp. time (sec.)	360-3600	60	180-3600, 120	3600-36000, 600

Table 4: Computing time for the parameter inference and for the prediction, \mathcal{H} -MLE method.

Figures 4 (left and right) show predictions obtained by the \mathcal{H} -MLE method. The yellow points at 900.000 locations were used for training and the blue points were predicted at 100.000 new locations. One can see a very good alignment of yellow and blue points (on the right) and slightly different values on the left.

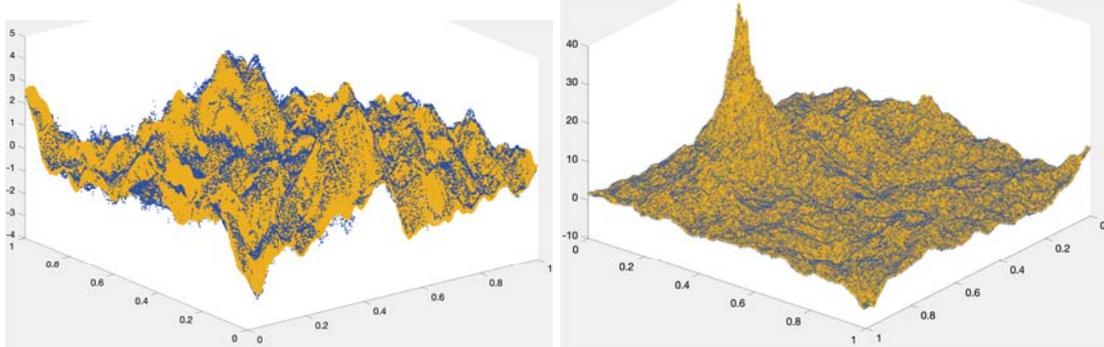


Figure 4: Test-2b, datasets 1 and 2: Prediction obtained by the \mathcal{H} -MLE method. The yellow points at 900.000 locations were used for training and the blue points were predicted in 100.000 locations. One can see a very good alignment of yellow and blue points (on the right) and slightly different values on the left.

6 Numerical results, obtained by the machine learning methods

To run the kNN method we used a usual notebook with Intel i5-9300 CPU, 2.40GHz and 8 GB RAM. We perform Monte-Carlo simulations, in which we repeatedly split the given dataset on training and testing subsamples and average the obtained prediction error estimates over all runs. In our experiments on both datasets in Test-2a, the kNN method has shown the best Monte-Carlo cross-validation results (over 100 runs) in comparison with other used ML methods. Predictions for datasets 1 and 2 from Test-2a are shown in Fig. 5.

Running the kNN method with different k , we found out that $k = 3$ is optimal for Test-2a, and $k = 7$ for Test 2b. Trying different numbers of trees in the random forest method, we defined that an ensemble of 120 regression trees is optimal. Further, we have designed the FCNN architecture with 7 hidden layers with 100 neurons in each layer. The number of training epochs is 500, and the batch size equals 10000. We use $\tanh(\cdot)$ activation function and Adam optimizer. The average calculation time is 0.07 sec. for kNN (k-d tree was used to speed up calculations), 12 sec. for RF and 173 sec. for FCNN. Because of its efficiency, we decided to run only the kNN method for the prediction in Test-2b.

Table 5 contains the RMSE errors for all methods (defined in Eq. 7). Note that the dataset2 from Test-2b is sampled from the same random field as the dataset1 from Test-2a. The dataset1 from Test-2b is sampled from the same random field as the dataset5 from Test-1a. Remarkable is that RMSE for the dataset5 (100.000 locations) in Test-1a (5th row in Table 1) is equal $5.4e - 3$, whereas RMSE for similar dataset1 (1.000.000 locations) from Test-2b is equal 0.25. We can explain this with 1) the fact that the MLE approach faces difficulties with large matrices, since the condition number of \mathbf{C} is increasing, and 2) the number of needed iterations in the MLE optimization procedure increases. We did not run RF and FCNN methods on Test-2b because the computing time is much larger than for the kNN time. The kNN time for Test-2b is 1.23 sec.

Predictions for datasets 1 and 2 from Test-2b are shown in Fig. 6. The training datasets are depicted by yellow points and kNN predictions by blue points. We can see that the kNN method provides very good results.

dataset	Test-2a		Test-2b	
	1	2	1	2
\mathcal{H} -MLE	0.057	0.14	0.25	0.021
kNN	0.129	0.357	0.007	0.04
RF	0.226	0.607	—	—
FCNN	0.243	0.74	—	—

Table 5: Comparison of RMSE errors for \mathcal{H} -MLE, kNN, RF, and FCNN methods.

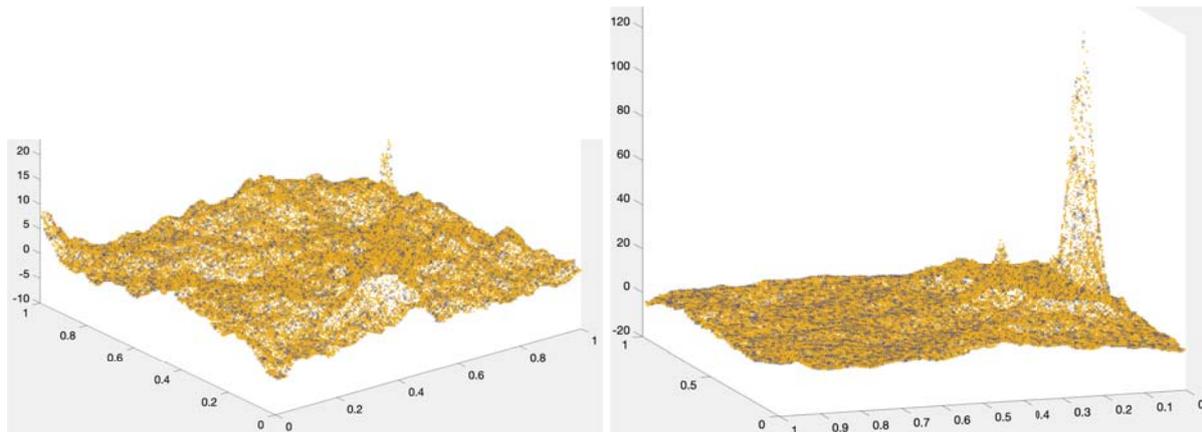


Figure 5: Test-2a, datasets 1 and 2: Prediction obtained by the kNN method. The yellow points at 90.000 locations were used for training and the blue points were predicted at 10.000 new locations. One can see a very good alignment of both.

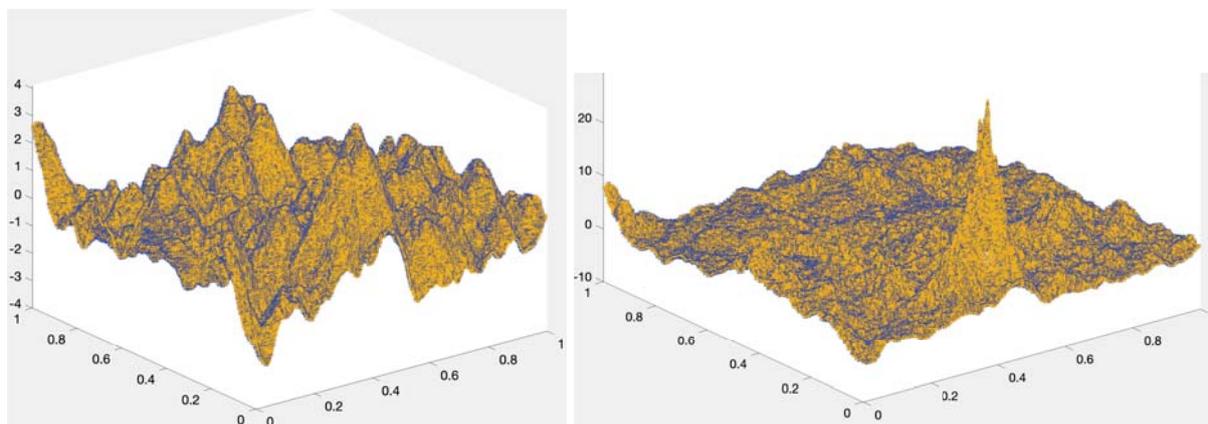


Figure 6: Test-2b, datasets 1(left) and 2(right): Prediction obtained by the kNN method. The yellow points at 900.000 locations were used for training and the blue points were predicted at 100.000 new locations. One can see a very good alignment of both.

7 Conclusion

We developed the \mathcal{H} -MLE procedure to estimate unknown parameters and to make statistical predictions. In order to make computations faster, we approximated the joint Gaussian log-likelihood function in the \mathcal{H} -matrix format. In the numerical section, we considered $8 + 2 + 2 = 12$ datasets: 8 in Tests 1a and 1b, 2 in Test-2a, and 2 in Test-2b. All datasets in Tests 1a, 1b and 2a contain 90,000 locations for training and 10,000 for testing (prediction). Both datasets in Test-2b contained 900,000 locations for training and 100,000 for prediction.

The \mathcal{H} -matrix technique drastically reduces the required memory and computing time, making it possible to work with larger sets of observations obtained on unstructured meshes. The main drawback of using the \mathcal{H} -matrix technique is that too many linear algebra operations are required to estimate just four scalar unknown parameters. For example, for some datasets with the unlikely chosen initial guess, we needed 400 iterations. On each iteration, we computed one \mathcal{H} -Cholesky factorization, one scalar product and solved a linear system. In total, it can take up to 8 hours on a modern parallel node for the dataset with 900,000 locations. A possible remedy is to precompute the initial guess. It will significantly reduce the required number of iterations. This could be done, for instance, on a smaller subset of observations. Another drawback is that

the \mathcal{H} -matrix approximation of \mathbf{C} and \mathbf{L} was recomputed entirely on every iteration for the new values of τ and σ . It would be a lot cheaper to add a new diagonal or scale the existing \mathbf{C} . It is indeed possible to modify the optimization algorithm, but the whole procedure will become more complicated. And the last drawback is that the total complexity depends on the matrix size and the number of parameters. For one to four parameters, the total computing time is acceptable, but it will be too large for five or more parameters. To tackle problems with large number of parameters we suggest to use low-rank tensor methods [30, 15, 25].

Among all implemented ML methods (kNN, random forest, deep neural network), the best results (for given datasets) were obtained by the kNN method with three or seven neighbors depending on the dataset. The results computed with the \mathcal{H} -MLE method were compared with the results obtained by the kNN method. For Test-2a, the \mathcal{H} -MLE method showed a smaller RMSE error than the kNN method, whereas, for Test-2b, the kNN method was better. To conclude, it is not surprising that our \mathcal{H} -MLE method worked fine on most datasets. We also understand that we can improve the \mathcal{H} -MLE results simply by taking a smaller threshold and more accurate \mathcal{H} -matrix arithmetics. What surprised us is that the well-known and straightforward kNN method performed very good and very fast. Since we did not make any theoretical comparison and compared \mathcal{H} -MLE and ML methods only numerically on given datasets, we can not in general conclude which method is better. We also remind that we used kNN only for the prediction.

Acknowledgment The study was carried out within the framework of the state contract of the Sobolev Institute of Mathematics (project no 0314-2019-0015). The work was partly supported by RFBR grant 19-29-01175. A. Litvinenko was supported by funding from the Alexander von Humboldt Foundation.

REFERENCES

- [1] S. Abdulah, H. Ltaief, Y. Sun, M. G. Genton, and D. E. Keyes. Exageostat: A high performance unified software for geostatistics on manycore systems. *IEEE Transactions on Parallel and Distributed Systems*, 29:2771–2784, 2018.
- [2] S. Ambikasaran, D. Foreman-Mackey, L. Greengard, D. W. Hogg, and M. O’Neil. Fast direct methods for gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):252–265, Feb 2016.
- [3] S. Ambikasaran, J. Y. Li, P. K. Kitanidis, and E. Darve. Large-scale stochastic linear inversion using hierarchical matrices. *Computational Geosciences*, 17(6):913–927, 2013.
- [4] J. Ballani and D. Kressner. Sparse inverse covariance estimation with hierarchical matrices. http://sma.epfl.ch/~anchpcommon/publications/quic_ballani_kressner_2014.pdf, 2015.
- [5] M. Bebendorf and W. Hackbusch. Existence of H-matrix approximants to the inverse FE-matrix of elliptic operators with L^∞ -coefficients. *Numerische Mathematik*, 95(1):1–28, 2003.
- [6] S. Börm and J. Garcke. Approximating Gaussian processes with H^2 -matrices. In Joost N. Kok, Jacek Koronacki, Ramon Lopez de Mantaras, Stan Matwin, Dunja Mladen, and Andrzej Skowron, editors, *Proceedings of 18th European Conference on Machine Learning, Warsaw, Poland, September 17-21, 2007. ECML 2007*, volume 4701, pages 42–53, 2007.

- [7] R. P. Brent. Chapter 4: An algorithm with guaranteed convergence for finding a zero of a function, algorithms for minimization without derivatives. *Englewood Cliffs, NJ: Prentice-Hall*, 1973.
- [8] N. Cressie and Chr. Wikle. *Statistics For Spatio-Temporal Data*, volume 465. 01 2011.
- [9] R. Dahlhaus and H. Künsch. Edge effects and efficient parameter estimation for stationary random fields. *Biometrika*, 74(4):877–882, 1987.
- [10] A. Datta, S. Banerjee, A. O. Finley, and A. E. Gelfand. Hierarchical nearest-neighbor Gaussian process models for large geostatistical datasets. *Journal of the American Statistical Association*, 111(514):800–812, 2016.
- [11] C.R. Dietrich and G. N. Newsam. Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix. 18(4):1088–107, 1997.
- [12] M. Fuentes. Approximate likelihood for large irregularly spaced spatial data. *Journal of the American Statistical Association*, 102:321–331, 2007.
- [13] G.-A. Fuglstad, D. Simpson, F. Lindgren, and H. Rue. Does non-stationary spatial data always require non-stationary random fields? *Spatial Statistics*, 14:505–531, 2015.
- [14] R. Furrer, M. G. Genton, and D. Nychka. Covariance tapering for interpolation of large spatial datasets. *Journal of Computational and Graphical Statistics*, 15(3):502–523, 2006.
- [15] L. Grasedyck, D. Kressner, and Chr. Tobler. A literature survey of low-rank tensor approximation techniques. *GAMM-Mitteilungen*, 36(1):53–78, 2013.
- [16] J. Guinness and M. Fuentes. Circulant embedding of approximate covariances for inference from gaussian data on large lattices. *Journal of Computational and Graphical Statistics*, 26(1):88–97, 2017.
- [17] P. Guttorp and T. Gneiting. Studies in the history of probability and statistics XLIX: On the Matérn correlation family. *Biometrika*, 93:989–995, 2006.
- [18] W. Hackbusch. A sparse matrix arithmetic based on \mathcal{H} -matrices. I. Introduction to \mathcal{H} -matrices. *Computing*, 62(2):89–108, 1999.
- [19] W. Hackbusch. *Hierarchical matrices: Algorithms and Analysis*, volume 49 of *Springer Series in Comp. Math.* Springer, 2015.
- [20] H. Harbrecht, M. Peters, and M. Siebenmorgen. Efficient approximation of random fields for numerical applications. *Numerical Linear Algebra with Applications*, 22(4):596–617, 2015.
- [21] H. Huang, S. Abdulah, M. G. Genton, Y. Sun, H. Ltaief, and D. E. Keyes. 2021 KAUST Competition on Spatial Statistics for Large Datasets, 2020. <https://cemse.kaust.edu.sa/stsds/2021-kaust-competition-spatial-statistics-large-datasets>.
- [22] H. Huang and Y. Sun. Hierarchical low rank approximation of likelihoods for large spatial datasets. *Journal of Computational and Graphical Statistics*, 27(1):110–118, 2018.

- [23] M. Katzfuss and J. Guinness. A general framework for Vecchia approximations of Gaussian processes. *ArXiv e-prints*, August 2017.
- [24] C. G. Kaufman, M. J. Schervish, and D. W. Nychka. Covariance tapering for likelihood-based estimation in large spatial datasets. *Journal of the American Statistical Association*, 103(484):1545–1555, 2008.
- [25] B. Khoromskij. *Tensor Numerical Methods in Scientific Computing*. De Gruyter, 2018.
- [26] B. N Khoromskij, A. Litvinenko, and H. G. Matthies. Application of hierarchical matrices for computing the Karhunen–Loève expansion. *Computing*, 84(1-2):49–67, 2009.
- [27] William Kleiber and Douglas W Nychka. Equivalent kriging. *Spatial Statistics*, 12:31–49, 2015.
- [28] J. Y. Li, S. Ambikasaran, E. F. Darve, and P. K. Kitanidis. A Kalman filter powered by H2 matrices for quasi-continuous data assimilation problems. *Water Resources Research*, 50(5):3734–3749, 2014.
- [29] A. Litvinenko. Application of hierarchical matrices for solving multiscale problems, 2006. PhD Dissertation, Leipzig University, <https://publications.rwth-aachen.de/record/754296/files/754296.pdf>.
- [30] A. Litvinenko, D. Keyes, V. Khoromskaia, B. N. Khoromskij, and H. G. Matthies. Tucker Tensor analysis of Matern functions in spatial statistics. *Computational Methods in Applied Mathematics*, November 2018.
- [31] A. Litvinenko, R. Kriemann, M. G. Genton, Y. Sun, and D. E. Keyes. Hlibcov: Parallel hierarchical matrix approximation of large covariance matrices and likelihoods with applications in parameter identification. *MethodsX*, 7:100600, 2020.
- [32] A. Litvinenko, Y. Sun, M. G. Genton, and D. E. Keyes. Likelihood approximation with hierarchical matrices for large spatial datasets. *Computational Statistics & Data Analysis*, 137:115–132, 2019.
- [33] Bertil Matérn. *Spatial Variation*, volume 36 of *Lecture Notes in Statistics*. Springer-Verlag, Berlin; New York, second edition edition, 1986.
- [34] H. G. Matthies, E. Zander, O Pajonk, B. V. Rosić, and A. Litvinenko. Inverse problems in a Bayesian setting. In *Computational Methods for Solids and Fluids Multiscale Analysis, Probability Aspects and Model Reduction Editors: Ibrahimbegovic, Adnan (Ed.), ISSN: 1871-3033*, pages 245–286. Springer, 2016.
- [35] H. G. Matthies, E. Zander, B. V. Rosić, and A. Litvinenko. Parameter estimation via conditional expectation: a bayesian inversion. *Advanced Modeling and Simulation in Engineering Sciences*, 3(1):24, 2016.
- [36] W Nowak and A Litvinenko. Kriging and spatial design accelerated by orders of magnitude: combining low-rank covariance approximations with FFT-techniques. *Mathematical Geosciences*, 45(4):411–435, 2013.

- [37] D. Nychka, S. Bandyopadhyay, D. Hammerling, F. Lindgren, and S. Sain. A multiresolution Gaussian process model for the analysis of large spatial datasets. *Journal of Computational and Graphical Statistics*, 24(2):579–599, 2015.
- [38] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Section 9.3. Van Wijngaarden-Dekker-Brent Method. Numerical Recipes: The Art of Scientific Computing*, volume 3rd ed. New York: Cambridge University Press., 2007.
- [39] A. Saibaba and P. Kitanidis. Efficient methods for large-scale linear inversion using a geostatistical approach. *Water Resources Research*, 48(5).
- [40] A. Saibaba and P. Kitanidis. Fast computation of uncertainty quantification measures in the geostatistical approach to solve inverse problems. *Advances in Water Resources*, 82:124 – 138, 2015.
- [41] A.K. Saibaba, S. Ambikasaran, J Yue Li, P. K. Kitanidis, and E.F. Darve. Application of hierarchical matrices to linear inverse problems in geostatistics. *Oil & Gas Science and Technology—Rev. IFP Energies Nouvelles*, 67(5):857–875, 2012.
- [42] H. Sang and J. Z. Huang. A full scale approximation of covariance functions for large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(1):111–132, 2012.
- [43] M. L. Stein. Statistical properties of covariance tapers. *Journal of Computational and Graphical Statistics*, 22(4):866–885, 2013.
- [44] M. L. Stein, J. Chen, and M. Anitescu. Stochastic approximation of score functions for gaussian processes. *Ann. Appl. Stat.*, 7(2):1162–1191, 06 2013.
- [45] J. R. Stroud, M. L. Stein, and S. Lysen. Bayesian and maximum likelihood estimation for gaussian processes on an incomplete lattice. *Journal of Computational and Graphical Statistics*, 26(1):108–120, 2017.
- [46] A. V. Vecchia. Estimation and model identification for continuous spatial processes. *Journal of the Royal Statistical Society. Series B (Methodological)*, 50(2):297–312, 1988.
- [47] P. Whittle. On stationary processes in the plane. *Biometrika*, 41(3-4):434–449, 1954.
- [48] G. Xu and M. G. Genton. Tukey g-and-h random fields. *Journal of the American Statistical Association*, 112(519):1236–1249, 2017.

OPTIMAL SELECTION OF BAYESIAN VIRTUAL SENSORS FOR DAMAGE DETECTION UNDER VARIABLE ENVIRONMENTAL CONDITIONS

Jyrki Kullaa

Department of Automotive and Mechanical Engineering
Metropolia University of Applied Sciences
P.O. Box 4071, 00079 Metropolia, Finland
e-mail: jyrki.kullaa@metropolia.fi

Abstract

Measuring structural vibrations with a large sensor network results in lots of data in structural health monitoring applications. A large number of sensors is advantageous for damage detection and localization. By storing only a few selected Bayesian virtual sensors it is possible to decrease the amount of data and reconstruct the discarded sensor data even with higher accuracy than the original measurements. A method is proposed, in which the stored and reconstructed data are used for damage detection and localization in the time domain. A numerical experiment was performed with a structure having a large number of sensors. The excitation and environmental conditions were variable and unknown. An optimal sensor placement algorithm was applied individually to each measurement to select the appropriate virtual sensors for storage. Less than ten percent of the data were stored, and the signals of all the reconstructed sensors were still more accurate than the actual measurements. The stored and reconstructed data outperformed the actual measurement data in damage detection and localization. Surprisingly, damage detection was also more successful with the stored and reconstructed data than with the full set of virtual sensors.

Keywords: Data Compression, Virtual Sensing, Damage Detection, Optimal Sensor Placement, Environmental Effects, Sensor Network.

1 INTRODUCTION

Structural health monitoring (SHM) produces lots of data. Vibrations of structures are measured frequently using a sensor network with tens or hundreds of sensors. There is an increasing interest to develop dense sensor networks for SHM applications, for example sensing skins [1]. With an increasing number of sensors, damage detection and localization become more reliable, but at the expense of higher data management costs. Historical data must be stored for unsupervised learning to train the model of the undamaged structure under different environmental or operational conditions. Storing such a large amount of data during years of monitoring may be difficult and costly. Therefore, data reduction would be necessary. Data can be reduced using multivariate statistical techniques for dimensionality reduction. The most common dimensionality reduction method is principal component analysis (PCA) [2], which is a linear method that maximizes the variance in the data by projecting the data onto directions, principal components (PC) that account for the largest variability. If only a few PCs are retained, some loss of data results. PCA has been applied e.g. to image compression [3].

Another method for data reduction is to extract selected features from the time records and store these features only. Features are dynamic characteristics of the structure, which are expected to be sensitive to damage. Such features are for example natural frequencies and mode shapes, which can be extracted from the measurement data using system identification techniques [4]. Significant data compression occurs, because a single feature vector (one data point) only results from each measurement. The dimensionality of the feature vector is, however, often higher than the number of sensors. This may result in the curse of dimensionality in statistical data analysis. In addition, the time histories will be lost and cannot be recovered in case new potential features are later considered. Therefore, it may be necessary to save everything resulting in terabytes of data every day [5].

If only a limited number of sensor signals are permanently stored, the sensors must be selected according to some criterion. This selection is associated with optimal sensor placement (OSP), which has been studied for different applications including vibration control, experimental modal analysis, model updating, fault detection, and impact identification [6]. Although the objective in these applications is to place a limited amount of physical sensors in optimal positions, the same approach can be used to select a subset of measurements for storage. The selection criterion is related to the accuracy of the reconstructed signals.

Some review papers and comparisons of different optimal sensor placement algorithms exist [7–10]. They present the most commonly applied algorithms and criteria. Sensor placement is a discrete optimization problem, for which genetic algorithms have been proposed [11–13]. Alternatively, a computationally efficient and widely used algorithm is to start with a large set of candidate sensor locations and remove one sensor in each round based on the selected cost function until the required criterion is violated. This backward sequential sensor placement (BSSP) algorithm has been used in many studies [14–16]. Another iterative method is to add one sensor in turn to the sensor network until the stopping criterion is met. The algorithm is called forward sequential sensor placement (FSSP) algorithm [11, 16]. BSSP is used in this study.

If the number of sensors in the network is larger than the number of active modes, the sensor network is redundant. The redundancy can be utilized to estimate the quantity of interest using virtual sensing techniques. Virtual sensing (VS) can be either model-based (analytical) or data-driven (empirical) [17]. In analytical virtual sensing, in addition to measurement data, a numerical model of the structure is needed, for example a finite element model. Empirical virtual sensing is based on training data from a redundant full sensor network. It can be used,

for example, to replace a temporarily installed or failed sensor [18]. Empirical virtual sensing has also been used in structural dynamics for damage detection [19] or sensor fault detection [20].

In this paper, virtual sensors are developed for data compression and accurate reconstruction in a large sensor network. The objective is to detect and localize damage using the stored and reconstructed virtual sensor data.

With empirical Bayesian virtual sensing, the resulted accuracy of the virtual sensors is higher than that of the physical hardware [21]. A limited number of virtual sensors are stored, from which the discarded signals can be reconstructed. Optimal sensor placement is studied for the most accurate reconstruction of the excluded data. The cost function in the sensor placement optimization is related to the reconstruction error, which must be minimized.

Damage detection is based on changes in the dynamic characteristics of the structure. Records of structural motion, for example acceleration, are measured simultaneously at selected degrees-of-freedom. First, a training data set is acquired from the undamaged structure under different environmental or operational conditions. These data are used to build a statistical data model of the undamaged structure. Next, the structure is being monitored with repeated measurements in order to have an early warning of structural failure. The new test data are compared to the training data using novelty detection techniques, and a statistically significant change in the dynamic characteristics is an indication of damage. Particular attention is needed to take variable environmental or operational conditions into account, because they can have a considerable influence on the very same dynamic characteristics. Several techniques have been proposed to eliminate the environmental or operational influences on the data, even without measuring the underlying quantities, see e.g. [19, 22] and the references therein. Damage localization can also be attempted if the changes in the data can be assigned to a particular sensor.

In this paper, damage detection and localization are performed in the time domain. The data are the stored and reconstructed virtual sensors. Statistically significant differences between the training and test data are assumed to reveal damage. The largest difference is assumed to localize damage close to the corresponding sensor.

The paper is organized as follows. Virtual sensing using Bayesian estimation is outlined in Section 2. Optimal sensor placement for virtual sensing is also discussed. An algorithm for damage detection and localization follows in Section 3. In Section 4, the proposed method is studied with numerical simulations of ambient vibration measurements. Concluding remarks are given in Section 5.

2 VIRTUAL SENSING AND OPTIMAL SENSOR PLACEMENT

The objective is to store only a small percentage of the dense sensor network data so that the full data can be accurately reconstructed for damage detection. One possible data compression technique is Bayesian virtual sensing, which is applied to the whole sensor network, and a selected set of the resulting virtual sensors are only stored. The data from the discarded sensors can be reconstructed using the stored signals.

2.1 Empirical Bayesian virtual sensing

Empirical virtual sensing is based on available current or historical measurements. Consider a sensor network measuring p simultaneously sampled response variables $\mathbf{y} = \mathbf{y}(t)$ at time t . Each measurement \mathbf{y} includes measurement error $\mathbf{w} = \mathbf{w}(t)$:

$$\mathbf{y} = \mathbf{x} + \mathbf{w} \quad (1)$$

where $\mathbf{x} = \mathbf{x}(t)$ are the exact values of the measured degrees of freedom. Equation 1 can be written in the following form at time t [23].

$$\begin{Bmatrix} \mathbf{x} \\ \mathbf{y} \end{Bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{I} \end{bmatrix} \begin{Bmatrix} \mathbf{x} \\ \mathbf{w} \end{Bmatrix} \quad (2)$$

For simplicity but without loss of generality, assume zero-mean variables \mathbf{x} and \mathbf{y} . The partitioned covariance matrix is

$$\begin{aligned} \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} &= E\left(\begin{Bmatrix} \mathbf{x} \\ \mathbf{y} \end{Bmatrix} \begin{bmatrix} \mathbf{x}^T & \mathbf{y}^T \end{bmatrix}\right) = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{I} \end{bmatrix} E\left(\begin{Bmatrix} \mathbf{x} \\ \mathbf{w} \end{Bmatrix} \begin{bmatrix} \mathbf{x}^T & \mathbf{w}^T \end{bmatrix}\right) \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{I} \end{bmatrix}^T \\ &= \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \Sigma_{xx} & \mathbf{0} \\ \mathbf{0} & \Sigma_{ww} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} = \begin{bmatrix} \Sigma_{xx} & \Sigma_{xx} \\ \Sigma_{xx} & \Sigma_{xx} + \Sigma_{ww} \end{bmatrix} \end{aligned} \quad (3)$$

where $E(\cdot)$ denotes the expectation operator and the measurement error \mathbf{w} is assumed to be zero mean Gaussian, independent of \mathbf{x} , with a covariance matrix Σ_{ww} .

A linear minimum mean square error (MMSE) estimate for $\mathbf{x} | \mathbf{y}$ (\mathbf{x} given \mathbf{y}) is obtained by minimizing the mean-square error (MSE) [20, 23]. The expected value, or the conditional mean, of the predicted variable is:

$$\hat{\mathbf{x}} = E(\mathbf{x} | \mathbf{y}) = \Sigma_{xx} (\Sigma_{xx} + \Sigma_{ww})^{-1} \mathbf{y} = \Sigma_{xx} \Sigma_{yy}^{-1} \mathbf{y} \quad (4)$$

and the estimation error is

$$\Sigma_{\text{post}} = \text{cov}(\mathbf{x} | \mathbf{y}) = \Sigma_{xx} - \Sigma_{xx} (\Sigma_{xx} + \Sigma_{ww})^{-1} \Sigma_{xx} = \Sigma_{xx} - \Sigma_{xx} \Sigma_{yy}^{-1} \Sigma_{xx} \quad (5)$$

The covariance matrix Σ_{xx} is not known, but Σ_{yy} can be estimated from the measurement data. If the noise covariance matrix can be approximated, then an estimate for $\Sigma_{xx} = \Sigma_{yy} - \Sigma_{ww}$ can be computed. In this study, measurement errors are assumed uncorrelated between sensors resulting in a diagonal noise covariance matrix. In addition, because Σ_{xx} must be positive definite, an upper bound of the noise level in each sensor can be obtained [21].

2.2 Data compression

After a single measurement, the data from all sensors are available. If only a subset of the signals is stored, a lot of disc storage space can be saved. Let us assume that only channels v are stored. The stored signals can be either the actual measurements \mathbf{y}_v or virtual sensors $\hat{\mathbf{x}}_v$. It was proved [24] that the stored Bayesian virtual sensors $\hat{\mathbf{x}}_v$ outperform the corresponding raw measurements \mathbf{y}_v resulting in a smaller reconstruction error and should be preferred. Therefore, the stored data in this paper are from the virtual sensors.

Let us assume that channels v of the virtual sensors are stored, while the remaining channels u must be reconstructed. Storing the virtual sensors $\hat{\mathbf{x}}_v$, the conditional mean $E(\mathbf{x}_u | \hat{\mathbf{x}}_v)$ and covariance matrix $\text{cov}(\mathbf{x}_u | \hat{\mathbf{x}}_v)$ must be derived. The Bayesian virtual sensors are not exact, but follow the error model

$$\mathbf{x} = \hat{\mathbf{x}} + \mathbf{e} \quad (6)$$

where \mathbf{e} is the posterior error (5) having a zero mean. Thus,

$$E(\mathbf{x}_u | \hat{\mathbf{x}}_v) = E(\hat{\mathbf{x}}_u | \hat{\mathbf{x}}_v) = \Sigma_{\hat{\mathbf{x}}_u \hat{\mathbf{x}}_v} \Sigma_{\hat{\mathbf{x}}_v \hat{\mathbf{x}}_v}^{-1} \hat{\mathbf{x}}_v = \mathbf{A} \hat{\mathbf{x}}_v \quad (7)$$

where $\mathbf{A} = \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_u, \hat{\mathbf{x}}_v} \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_v, \hat{\mathbf{x}}_v}^{-1}$ is the coefficient matrix that has to be stored together with the stored signals $\hat{\mathbf{x}}_v$. According to MMSE, the two terms in the right hand side of (6) are orthogonal [25]. Therefore, the covariances are related as

$$\begin{aligned} \text{cov}(\mathbf{x}_u | \hat{\mathbf{x}}_v) &= \text{cov}(\hat{\mathbf{x}}_u | \hat{\mathbf{x}}_v) + \boldsymbol{\Sigma}_{\text{post},uu} \\ &= \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_u,uu} - \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_u,uv} \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_v,uv}^{-1} \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_v,vu} + \boldsymbol{\Sigma}_{\text{post},uu} \end{aligned} \quad (8)$$

which means that the error of the reconstructed sensor is larger than that in the Bayesian virtual sensor (5). This difference in accuracy is studied in Section 4.1.

2.3 Optimal sensor placement

The proposed optimal sensor placement algorithm is an iterative procedure starting with an initial large sensor network including all measured degrees-of-freedom (DOF). Each sensor in turn is removed with replacement, and the error variances of all reconstructed signals are computed, which are the diagonal terms of the error covariance matrix (8). The cost function for these reduced sensor networks is evaluated. The minimum cost is found and the reduced sensor network corresponding to this minimum becomes the new candidate set for the next round. In other words, the removed sensor corresponding to this minimum cost is permanently discarded. The process is repeated until the desired number of sensors or the allowed error limit is reached. Finally, the data from the remaining sensors are stored together with matrix \mathbf{A} in (7) for reconstruction of the discarded sensors.

3 DAMAGE DETECTION AND LOCALIZATION

Damage detection is applied to the reconstructed data in the time domain. First, the mean vector and the covariance matrix are estimated using training data from the undamaged structure under different environmental or operational conditions. Whitening transformation is applied to the training data [26]. This transformation is then fixed and applied to the test data. The residual errors between the model and actual data are computed and subjected to principal component analysis (PCA). Retaining the first principal component scores of the residuals, the data dimensionality is decreased to one. An extreme value statistics control chart is then designed for the first PC scores of the residuals with appropriate control limits and subgroup size [19, 27, 28]. In this paper, the probability of false alarms equal to 0.001 has been used.

Damage location is assumed to correspond to the direction of the first principal component of the residuals. The largest projection of the first PC on the sensor coordinates reveals the sensor closest to damage.

It is essential to model the data of each measurement independently for compression and reconstruction so that the environmental or operational or damage effects are retained during this first phase. Elimination of the environmental or operational influences is performed only in the second phase in which several measurements are pooled to build a data model of the undamaged structure under different environmental or operational conditions. Novelty detection is then applied to the test data using the data model of the second phase.

4 NUMERICAL EXPERIMENT

A numerical experiment was performed with a finite element (FE) model of a steel frame having a height of 4.0 m and a width of 3.0 m (Figure 1). The columns were fixed at the bottom. The frame was also supported with a horizontal spring at the elevation of 2.75 m with a spring constant of $k = 2.0$ MN/m. The frame was modelled with simple beam elements having

a hollow square cross section of $100 \text{ mm} \times 100 \text{ mm} \times 5 \text{ mm}$. The FE model consisted of 176 beam elements 62.5 mm in length and a single spring element.

Three horizontal random excitations were applied to the right column at elevations of 4 m, 3 m, and 2 m, respectively (Figure 1). The loads were mutually independent having random standard deviations between 100 N and 900 N. Periodic pseudorandom excitations in the frequency range between 0 and 53.33 Hz with random amplitudes and phases were generated [3]. All analyses had different loading functions. The first seven modes were used in the simulation. Modal damping was assumed with damping ratios of $\zeta_{1-2} = 0.01$, $\zeta_3 = 0.015$, and $\zeta_{4-7} = 0.02$.

Steady state analysis was performed in the frequency domain using modal superposition. Lateral accelerations at 59 points (every third node) were recorded. The sampling frequency was 250 Hz and the measurement period was 32.77 s. Each sensor thus produced 8192 samples. Mutually independent Gaussian random noise with equal standard deviations was added to each sensor. The average SNR was 30 dB. For validation and comparison, exact transverse accelerations were also recorded. The standard deviation of the noise was assumed to be known.

A relatively complex but also quite realistic environmental model was applied. The temperature of the left upper corner, T_{65} varied randomly between -25°C and $+40^\circ\text{C}$. The subscript 65 indicates the node number. The temperature of the other end points varied randomly: $T_{113} = T_{65} \pm 5^\circ\text{C}$ (upper right corner); $T_1 = T_{65} \pm 3^\circ\text{C}$ (bottom left support); and $T_{177} = T_{113} \pm 3^\circ\text{C}$ (bottom right support). Temperature variation between the aforementioned points was linear except that Gaussian random error with a standard deviation of 0.2°C was added to each element. The relationship between temperature and the Young's modulus E was stepwise linear as shown in Fig. 2a. Sample distributions of the Young's modulus in the elements are plotted in Fig. 2b. Within each short measurement, the distribution did not change.

Due to the temperature effect, the natural frequencies varied between measurements. Fig. 3 shows the seven lowest natural frequencies of the structure in all measurements. The data points on the right hand side of the vertical line are from the damaged structure. It is difficult to detect damage visually from the frequency changes due to the strong environmental effect.

Damage was removal of material inside a beam element due to corrosion. The damaged element was located at the bottom of the left leg (element 1). Five different damage levels were considered with the wall thicknesses of 4.5, 4.0, 3.5, 3.0, and 2.5 mm. Notice that as the material was removed, both the stiffness and mass were decreased.

The first 100 measurements were taken from the undamaged structure and each damage level was monitored with six measurements under different and unknown environmental conditions. Training data were the first 70 measurements. The extreme value statistics (EVS) control charts were designed using the same training data.

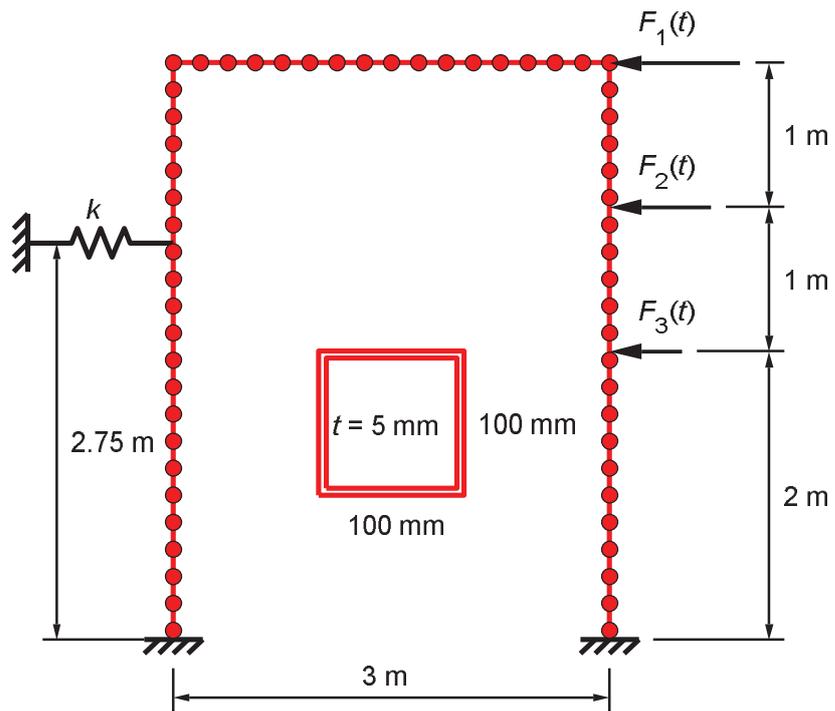


Figure 1: Frame structure with 59 accelerometers.

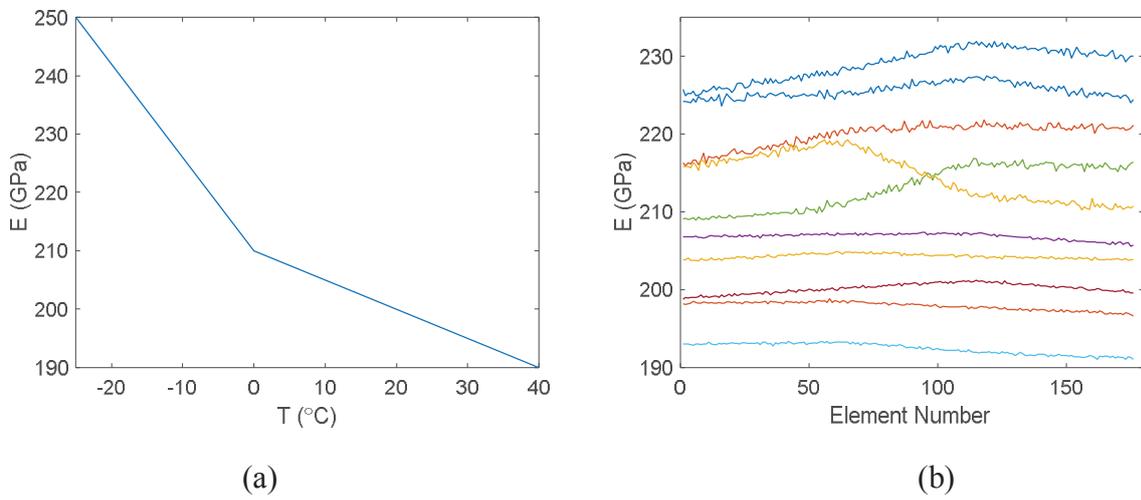


Figure 2: (a) Young's modulus versus temperature. (b) Sample distributions of the Young's modulus.

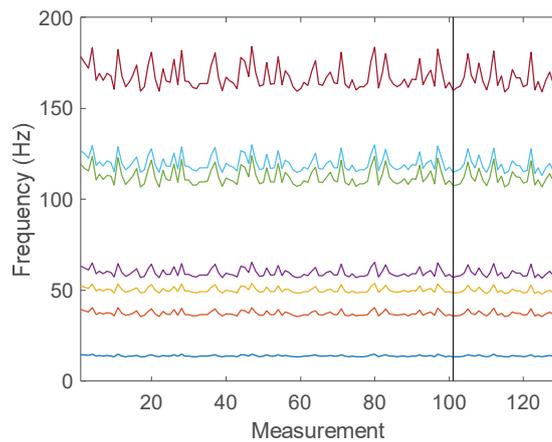


Figure 3: Variation of the seven lowest natural frequencies due to temperature and damage. Frequencies on the right of the vertical line are from the damaged structure.

4.1 Bayesian virtual sensing and sensor selection

Bayesian virtual sensing was applied individually to each measurement resulting in noise reduction. A detail of the measured and estimated accelerations of sensor 2 in measurement 1 is plotted in Figure 4. Also the exact values are shown. It can be seen that Bayesian virtual sensor was more accurate than the physical sensor.

Next, a subset of the Bayesian virtual sensors was selected separately for each measurement by applying the backward sequential sensor placement (BSSP) algorithm. The requirement was that noise had to be decreased at least 50% in all sensors. In other words, the standard deviation of the error in each reconstructed virtual sensor had to be less or equal to half of that of the corresponding measurement error. The cost function was the maximum difference between the current and allowed reconstruction errors in any sensor in the network. The reduced network having the minimum cost was selected for the next round. In other words, the aim was to maximize the minimum distance from the error limit. Sensor removal continued until the accuracy requirement was violated. The required number of virtual sensors was five for most measurements.

Once a single sensor was permanently removed, the mean error of the whole sensor network (stored and reconstructed virtual sensors) was evaluated. The mean error as a function of the number of stored sensors is plotted in Figure 5 for measurement 1. It can be seen that storing only five virtual sensors instead of all 59 virtual sensors did not significantly increase the average noise level. If the number of stored sensors were further decreased below five, the reconstruction error would have considerably increased.

The standard deviations of the errors (measurement error, Bayesian virtual sensor error, and reconstruction error) in all sensors are plotted in Figure 6 for measurement 1. It can be seen that the reconstruction error was only slightly larger than that in the Bayesian virtual sensors. Sensors, for which the two errors were equal, corresponded to the stored signals, which were not reconstructed. The measurement error is also shown. The reconstruction errors were clearly smaller than requested.

The reconstruction errors in all measurements are plotted in Figure 7 for each sensor. The variability between measurements was quite small satisfying the accuracy requirement.

A histogram of the selected sensors for storage in all measurements is shown in Figure 8 left. The most often selected sensors were located in six different regions of the structure. The placement of the stored sensors in measurement 1 is plotted in Figure 8 right. Notice that no sensors were selected close to damage location (sensor 1).

The data compression ratio was computed as follows. If all data were stored, the number of floating point numbers in each measurement was $59 \times 8192 = 483,328$ numbers. Storing five virtual sensor signals and the coefficient matrix \mathbf{A} (7) of size 54×5 resulted in 41,230 numbers. Consequently, only 8.5% of the total data had to be stored.

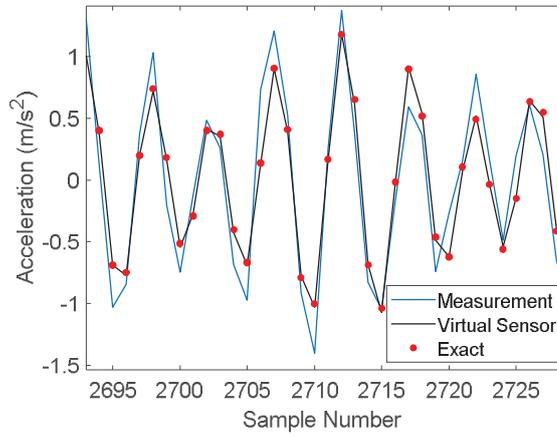


Figure 4: Detail of time history of accelerometer 2 in measurement 1.

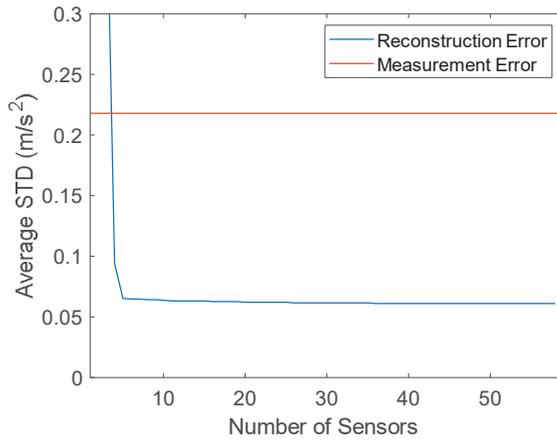


Figure 5: Mean reconstruction error as a function of the number of stored signals in measurement 1. The red horizontal line is the measurement error.

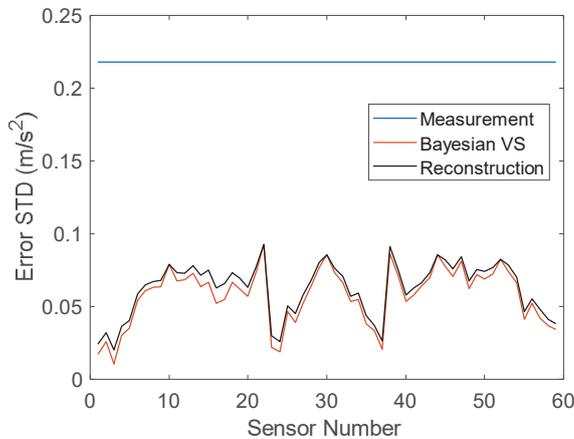


Figure 6: Measurement error, virtual sensor error and the reconstruction error in all sensors in measurement 1.

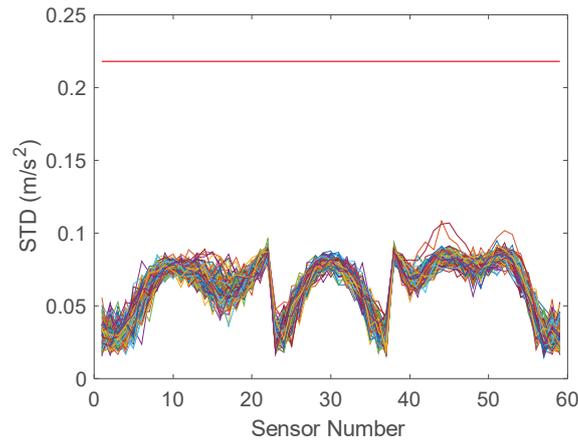


Figure 7: Measurement error (red horizontal line) and the reconstruction error in all sensors in each measurement.

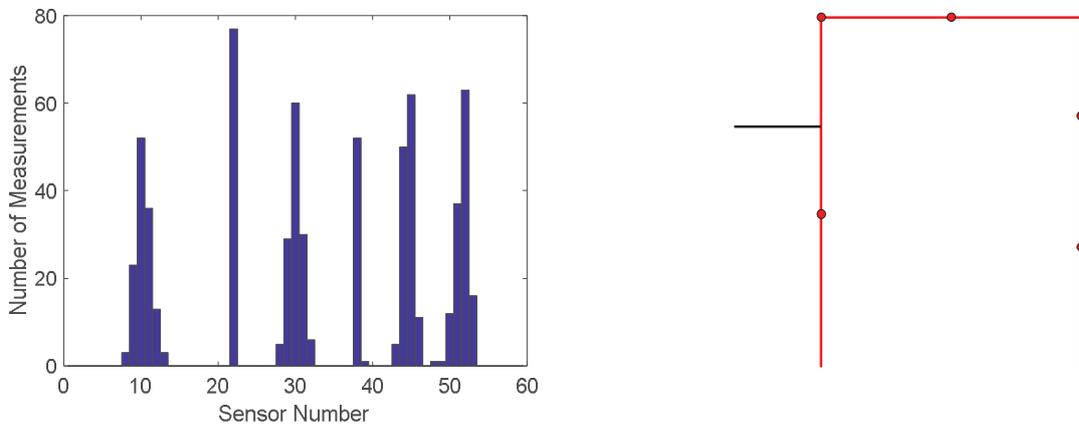


Figure 8: Left: Histogram of selected virtual sensors for storage in all measurements. Right: Selected sensors for storage in measurement 1: sensors 10, 22, 30, 44, and 52.

4.2 Damage detection and localization

Damage detection was studied using four different data: actual measurements, all virtual sensors, stored and reconstructed signals, and stored signals only. EVS control charts were designed with a subgroup size of 1000 and are plotted in Figure 9. The data points to the left of the blue vertical line correspond to the training data, while the black vertical lines indicate the five damage levels. Only the two largest damage levels were detected using the actual measurement data (Figure 9a). All damage cases were detected using the stored and reconstructed data (Figure 9b) or all virtual sensors (Figure 9c). There is a slight difference between the two control charts showing that the detection performance increased due to compression. This was quite a surprise, because the noise level in the reconstructed data was slightly larger than in the Bayesian virtual sensor data. The reason for this behavior is not known and it is questionable if this result can be generalized.

It may be argued that due to redundancy, only the selected virtual sensors would be enough for damage detection. This argument was tested by selecting the same five virtual sensors from each measurement and designing an EVS control chart for these data (Figure 9d). No damage was detected. Due to different environmental conditions between measurements, more than five signals would have been needed to remove the environmental influences.

Damage localization was done by plotting the squared projection of the first principal component on each sensor (Figure 10). Using the actual measurement data, damage was localized to sensor 6, and using the stored and reconstructed virtual sensors, damage was localized to sensor 4. Notice that sensor 4 was not included in the stored sensors but was reconstructed. The correct position was closest to sensor 1. Neither analysis pointed to the correct sensor, but in either case, the suggested damage location was in the vicinity of the actual damage. The localization accuracy was slightly higher when the virtual sensors were used. The SNR in sensor 1 was very small, which probably resulted in the inaccuracy in damage localization. In many structures, damage may be located close to the fixed support, where the stresses are large but the vibration amplitude is very small resulting in a small SNR. Therefore, strain measurements at these locations could be considered.

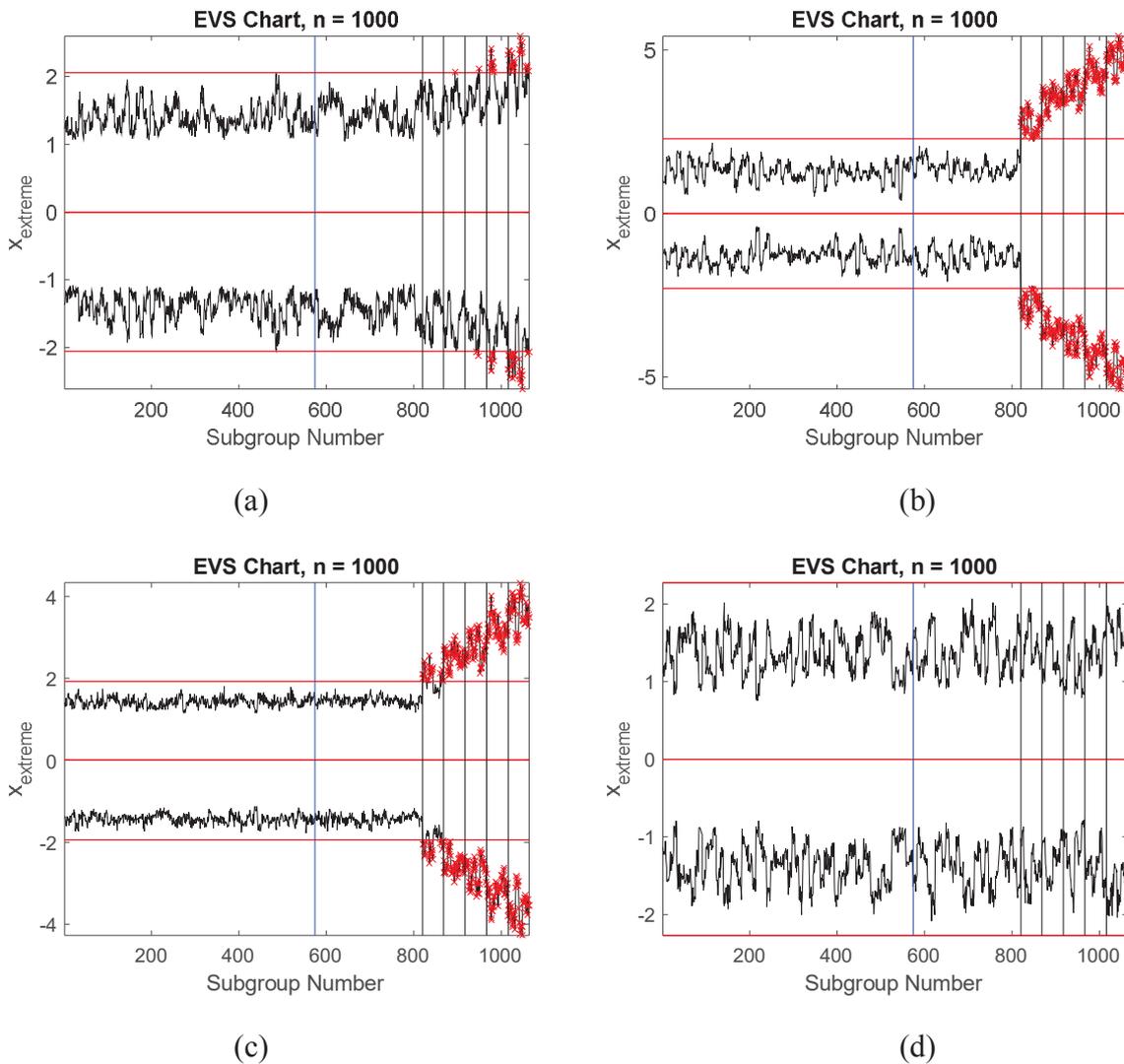


Figure 9: Damage detection. (a) All physical sensors, (b) stored and reconstructed sensors, (c) all Bayesian virtual sensors, and (d) stored 5 virtual sensors. The vertical lines correspond to the end of training data (blue) and the five damage levels (black).

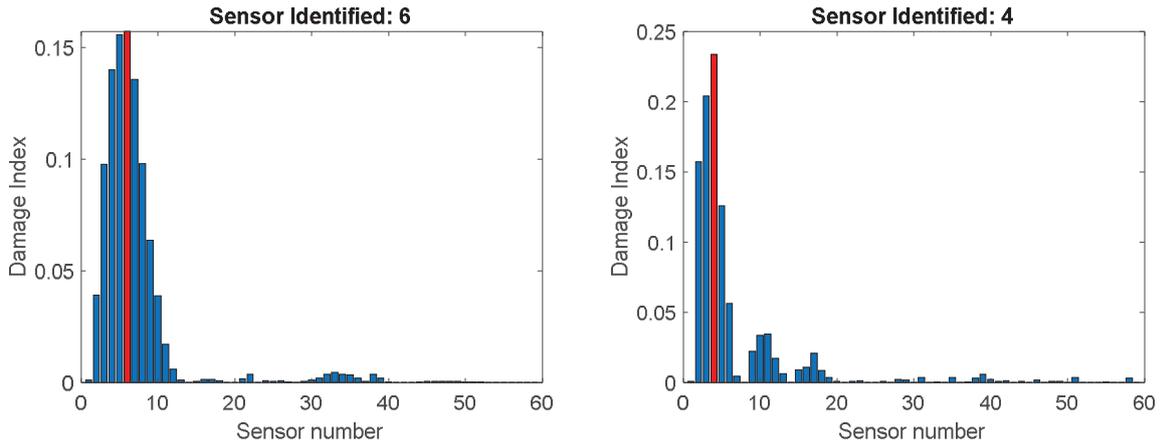


Figure 10: Damage localization. (a) All physical sensors, and (b) stored and reconstructed virtual sensors. The correct damage position was closest to sensor 1.

5 CONCLUSION

A data compression technique for storing and reconstructing simultaneously measured vibration signals in a dense sensor network was proposed. The stored and reconstructed data were used to detect and localize damage.

The first step was to reduce measurement error by applying Bayesian virtual sensing. The virtual sensors, being more accurate than the physical sensors, replaced the actual measurements in the subsequent steps.

Data compression and reconstruction was made individually for each measurement, because the dynamic characteristics of the structure could vary between measurements due to environmental or operational variability or damage. On the other hand, a full set of training data from several measurements under different environmental or operational conditions was used to build a covariance model of the undamaged structure. This model was applied to novelty detection using whitening transformation and principal component analysis. The largest discrepancy between the model and actual data was assumed to reveal the sensor closest to the damage location.

The main results are: (1) Only around 8.5% of the total amount of data had to be stored in the studied example. (2) The stored and reconstructed virtual sensor data were more accurate than the actual measurements. (3) Damage detection and localization were more reliable with the stored and reconstructed virtual sensors than with the actual measurements. (4) Damage was not localized exactly to the correct sensor but in the close neighborhood. (5) The accuracy of the reconstructed virtual sensors was only slightly smaller than that of the Bayesian virtual sensors. (6) The reconstruction errors were not the same even if the measurement errors were equal. (7) Damage localization to a reconstructed virtual sensor was possible. (8) Damage detection performance was slightly higher using the stored and reconstructed data than all virtual sensors, but generalization of this result remained questionable and needs further investigation. Experimental results are also needed to validate the proposed technique.

ACKNOWLEDGEMENTS

This research was supported by Metropolia University of Applied Sciences.

REFERENCES

- [1] M. Sadoughi, A. Downey, J. Yan, C. Hu, S. Laflamme, Reconstruction of unidirectional strain maps via iterative signal fusion for mesoscale structures monitored by a sensing skin. *Mechanical Systems and Signal Processing*, **112**, 401-416, 2018.
- [2] S. Sharma, *Applied multivariate techniques*. Wiley, 1996.
- [3] A. Brandt, *Noise and vibration analysis: Signal analysis and experimental procedures*. Wiley, 2011.
- [4] R. Brincker, C. Ventura, *Introduction to operational modal analysis*. Wiley, 2015.
- [5] P. Cawley, Structural health monitoring: closing the gap between research and industrial deployment. *Structural Health Monitoring*, **17**, 1225–1244, 2018.
- [6] V. Mallardo, M. Aliabadi, Optimal sensor placement for structural, damage and impact identification: A review, *Structural Durability & Health Monitoring*, **9**, 287–323, 2013.
- [7] T-H. Yi, H-N. Li, Methodology developments in sensor placement for health monitoring of civil infrastructures, *International Journal of Distributed Sensor Networks*, 1550–1329, 2012.
- [8] A. Krause, C. Guestrin, A. Gupta, J. Kleinberg, Near-optimal sensor placements: maximizing information while minimizing communication cost, *Proceedings of the 5th International Conference on Information Processing in Sensor Networks (IPSN '06)*, New York, NY, USA, 2006, 2–10.
- [9] M. Meo, G. Zumpano, On the optimal sensor placement techniques for a bridge structure, *Engineering Structures*, **27**, 1488–1497, 2005.
- [10] C. Leyder, E. Chatzi, A. Frangi, G. Lombaert, Comparison of optimal sensor placement algorithms via implementation on an innovative timber structure. J. Bakker, D.M. Frangopol, K. van Breugel eds. *Life-Cycle of Engineering Systems: Emphasis on Sustainable Civil Infrastructure. Proceedings of the Fifth International Symposium on Life-Cycle Civil Engineering (IALCCE 2016)*, Delft, The Netherlands, October 16–19, 2016, 260–267.
- [11] C. Papadimitriou, Optimal sensor placement methodology for parametric identification of structural systems, *Journal of Sound and Vibration*, **278**, 923–947, 2004.
- [12] J-H. Han, I. Lee, Optimal placement of piezoelectric sensors and actuators for vibration control of a composite plate using genetic algorithms, *Smart Materials and Structures*, **8**, 257–267, 1999.
- [13] K. Worden, A.P. Burrows, Optimal sensor placement for fault detection, *Engineering Structures*, **23**, 885–901, 2001.
- [14] 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.
- [15] D.C. Kammer, Effects of noise on sensor placement for on-orbit modal identification of large space structures, *Journal of dynamic systems, measurements and control—Transactions of the ASCE*, **114**, 436–443, 1992.
- [16] C. Papadimitriou, G. Lombaert, The effect of prediction error correlation on optimal sensor placement in structural dynamics, *Mechanical Systems and Signal Processing*, **28**, 105–127, 2012.

- [17] B. Lin, B. Recke, J.K.H. Knudsen, S.B. Jørgensen, A systematic approach for soft sensor development, *Computers and Chemical Engineering*, **31**, 419–425, 2007.
- [18] L. Liu, S.M. Kuo, M. Zhou, Virtual sensing techniques and their applications, *Proceedings of the 2009 IEEE International Conference on Networking, Sensing and Control*, Okayama, Japan, March 26–29, 2009, 31–36.
- [19] J. Kullaa, Robust damage detection in the time domain using Bayesian virtual sensing with noise reduction and environmental effect elimination capabilities, *Journal of Sound and Vibration*, **473**, 115232, 2020.
- [20] J. Kullaa, Sensor validation using minimum mean square error estimation, *Mechanical Systems and Signal Processing*, **24**, 1444–1457, 2010.
- [21] J. Kullaa, Bayesian virtual sensing in structural dynamics, *Mechanical Systems and Signal Processing*, **115**, 497–513, 2018.
- [22] H. Sohn, Effects of environmental and operational variability on structural health monitoring, *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **365**, 539–560, 2007.
- [23] L.L. Scharf, *Statistical signal processing: detection, estimation, and time series analysis*. Addison-Wesley, 1991.
- [24] J. Kullaa, Optimal sensor placement of Bayesian virtual sensors. W. Desmet, B. Pluymers, D. Moens, S. Vandemaele eds. *Proceedings of ISMA2020, International Conference on Noise and Vibration Engineering*, Leuven, Belgium, September 7–9, 2020, 973–985.
- [25] H. Stark, J.W. Woods, *Probability and random processes with applications to signal processing*. 3rd edition. Prentice-Hall, 2002.
- [26] J. Kullaa, Whitening transformation in damage detection. A.E. Del Grosso, P. Basso eds. *Smart structures: Proceedings of the 5th European Conference on Structural Control — EACS 2012*, Genoa, Italy, June 18–20, 2012.
- [27] K. Worden, D. Allen, H. Sohn, C.R. Farrar, Damage detection in mechanical structures using extreme value statistics, *SPIE Proceedings, Vol. 4693, 9th Annual International Symposium on Smart Structures and Materials*, San Diego, CA, 2002, 289–299.
- [28] D.C. Montgomery, *Introduction to statistical quality control, 3rd edition*, Wiley, 1997.

SENSOR FAULT LABEL IDENTIFICATION FOR ROBUST STRUCTURAL HEALTH MONITORING

Andreea-Maria Oncescu¹, Alice Cicirello²

¹Department of Engineering Science, University of Oxford
Parks Road, Oxford OX1 3PJ, UK
e-mail: andreea-maria.oncescu@sjc.ox.ac.uk

² Department of Engineering Structures, Delft University of Technology
Stevinweg 1, Delft 2628 CD, Netherlands
e-mail: a.cicirello@tudelft.nl

Keywords: Monitoring device failure, Failure reports, Natural Language Processing, Wearable device failure, Automatic failure label extraction.

Abstract. *Health Monitoring strategies rely on tracking the health status of critical engineering structures (Structural Health Monitoring) and of people (monitoring of medical conditions) to detect anomalies in the measurements and make inferences on the health condition for supporting decisions on preventive actions to be implemented to restore normal conditions. In these applications, the health monitoring devices are subjected daily to various events that can damage internal electrical components and sensors. As a result, the quality of the data collected can be compromised and therefore lead to a wrong health assessment. Therefore, robust health monitoring strategies need to be capable of automatically detecting sensors failures. Having the sensors' data is often not enough to gain insights into a monitoring system failure since the data variation can be related to changes in operating and environmental conditions. Alternatively, a supervised machine learning approach can be used. However, this requires an engineer to label the data in real-time, which rarely happens. Nonetheless, the common practice when a system fails is to write failure reports from which information about the failure can be extracted. Manually extracting comprehensive labels from the failure reports can be time-consuming. A strategy for automatically extracting failure labels from a set of failure reports written to describe failures of different types of sensors of a monitoring device is presented. This strategy consists in transforming the reports in their word vector form, processing each failure report to reduce the list of important words and identifying clusters of reports. The feasibility of the proposed approach is shown through its application to the failure reports compiled to describe seven types of failure of a low-cost wearable device based on an Arduino programmable board. Comparisons between manually extracted labels, and labels extracted with the proposed strategy when considering semi-supervised and unsupervised clustering strategies are presented. It is shown that the proposed strategy is capable of identify the failure label of a cluster of reports with a good accuracy. Therefore, enabling the development of a self-supervised classification algorithm for sensor fault identification for robust Structural Health Monitoring.*

1 INTRODUCTION

In engineering and healthcare applications, effective monitoring strategies are being developed tracking the health status of critical engineering structures (Structural Health Monitoring, SHM) [1, 2] and of people [3, 4] to make inferences on the health condition and support decisions, such as preventive actions to restore normal conditions. Therefore, the measurements obtained with the monitoring system must be informative, reliable and accurate. However, a monitoring device can fail during operating conditions because of poorly manufactured sensors and/or electronics, problems with cable harnesses, ageing effects, improper handling, electromagnetic interference, and environmental factors [5]. Unnoticed failures of the monitoring device undermine the quality of the measurements and consequently compromise inferences and decisions making. In SHM applications, a faulty monitoring device could lead to a wrong assessment of the remaining useful life of a structure [5]. This is one of the key bottlenecks undermining the reliable deployment of SHM technologies. A faulty wearable health monitoring devices can cause fatal conditions to be missed, over-treatment and it might produce health anxiety or fatigue [3, 6].

Failures of the monitoring device may not be detected during inspections [5]. The implementation of an additional monitoring system can be costly and prone to the same problems. Several investigations have been carried out for automatically detecting a faulty monitoring device for chemical process monitoring [7], in aircraft control applications [8, 9], in wearable health monitoring devices [10] and in SHM applications [1, 2, 5, 11, 12]. Broadly speaking, the approaches for sensor validation [5, 13] can be grouped into model-based approaches, knowledge-based approaches and data-driven approaches. Currently the monitoring device health status cannot be reliably identified and/or distinguished from structural failures and/or operating and environmental conditions by using only measurements [14, 15, 16, 17, 18, 19]. Alternatively, a supervised machine learning approach can be used where discriminative features in the measurements are paired with failure labels. However, this would require an engineer to label the data in real-time, which rarely happens and might be inaccurate [1, 2, 20], and to assess the discriminative features. Nonetheless, the common practice when a system fails is to write failure reports [21, 22] from which information about the failure of the device can be extracted. Manually extracting comprehensive labels from the failure reports can be time-consuming. Therefore, this work focusses on automatically extracting failure labels from a set of failure reports written to describe failures of different types of sensors of a monitoring device. This strategy consists in transforming the reports in their word vector form, processing each failure report to reduce the list of important words and identifying clusters of reports for each failure type. The feasibility of the proposed approach is shown through its application to the failure reports compiled to describe the sensor failures of a low-cost wearable device based on an Arduino programmable board. The chosen application displays similar challenges encountered in SHM applications, such as: (i) the sensors employed record various quantities at different rates; (ii) the measurements are influenced by operational and environmental conditions; (iii) similar failure types can occur for the same sensor; (iv) only a limited dataset of recorded failures is available; and (v) the number of elements in the training dataset for each failure type is imbalanced. Comparisons between manually extracted labels, and automatic extraction based on semi-supervised and unsupervised clustering strategies are presented. Finally, the implications of using these labels to train a self-supervised classification algorithm for sensor fault identification are discussed.

2 BRIEF DESCRIPTION OF THE MONITORING DEVICE AND FAILURES CONSIDERED

A low-cost wearable device that includes typical sensors used in wearable applications is chosen for investigating several failure types while keeping the costs low. This monitoring device is composed of a programmable Printed Circuit Board (Adafruit Metro Mini 328), a temperature sensor (digital Dallas Temperature Sensor), a humidity sensor (digital Grove - Temperature & Humidity Sensor Pro), an accelerometer (analog Triple Axis Accelerometer BMA220(Tiny)) and a Galvanic Skin Response (GSR) sensor. Seven types of failure are manually induced for a total of 117 failure instances. Specifically, three failure types are considered for the GSR sensor and two for the accelerometer, a failure type for the temperature sensor and another for the humidity sensor. Moreover, different number of failure instances are considered for each failure type.

Wearable devices, and in general small electronic devices, experience predominantly failures related to the solder joints and to the the sensor connectors [23]. These failures can be caused by improper soldering, ageing, improper handling of the wearable device or cracks in the solder at the connection point caused by a bent Printed Circuit Board (PCB). Within the current setup these failures can be easily reproduced by disconnecting wires at the interface with the PCB. Depending on the sensor and which pin was disconnected, the effects on the recorded signal varied. Moreover, another common failure type is related to burnt resistors. This failure type is induced by adding a resistor to the analog and power pins of the GSR sensor. The induced failures are summarised in Table 1.

Failure Type	Effects on measurements	Occurrences
(GSR, analog, pin)	jumps to 521	24
(GSR, ground, pin)	jumps above 1000	24
(GSR, burnt, resistor)	signal distorted	16
(accelerometer, ground, pin)	jumps to higher values	11
(accelerometer, power, pin)	jumps to lower values or zeros	11
(humidity, power, pin)	jumps to different values or -300%	18
(temperature, ground, pin)	jumps to different values or -127 ° C	13

Table 1: Induced failures and effects on recorded data

Data was recorded during controlled and operating conditions, and a failure report was written each time a failure occurred, for a total of 117 failure instances. Data and reports are stored within a Structured Query Language database for easy retrieval of information.

3 FAILURE INVESTIGATIONS AND FAILURE REPORTS

Failure investigations of a structure/system are carried out by an expert to identify the root-cause of failure and suggest remedial actions [21, 22]. Each failure investigation includes the measurements collected in operation, an analysis of the patterns observed in the measurements before and after the failure occurred, the lab experiments and steps required to identify the root-cause of the failure, and a failure report. Currently, the information collected during failure investigations is used for quality assessment, to support decisions about design changes and schedule maintenance [22]. The information collected during these investigations can be also used to improve SHM technologies.

Failure reports are documents with a standard outline [21, 22] and with sections written as free text. The first section focuses on the description of the failure effects observed during operating conditions, and it includes images and plots, and a brief description of the patterns observed in the measurements. Other sections focus on describing the steps taken to identify the root-cause of failure and to reproduce it; the remedial actions implemented; and how to manage similar failures in the future. One example of a failure report for the low-cost monitoring device under investigation is provided in Figure 1.

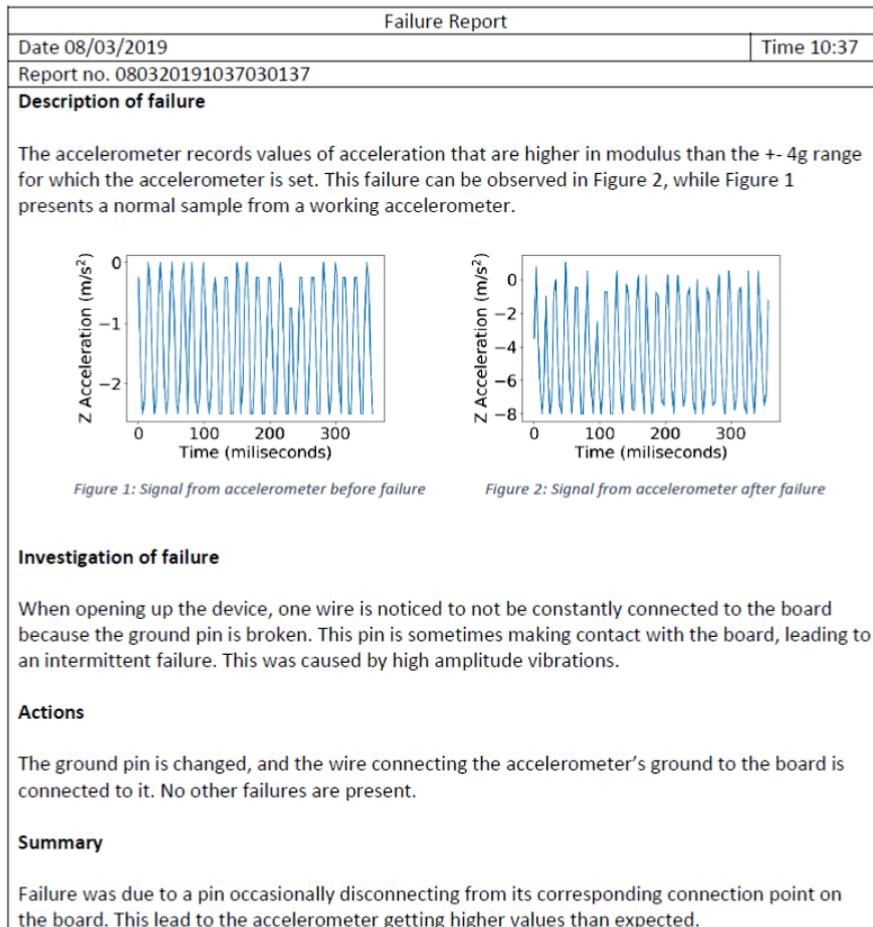


Figure 1: Example of a failure report.

Manual extraction of the information from reports can be time consuming and costly. Therefore a strategy for automatically grouping and extracting the failure labels from failure reports is presented.

4 APPROACH FOR REPORTS CLUSTERING AND LABEL EXTRACTION

A strategy is proposed for automatically grouping the reports according to the failure type described and extract the failure labels by pre-processing the failure reports and applying Natural Language Processing (NLP) techniques. Each document is represented as a vector in a multi-dimensional space, the so-called document embedding [24]. Initially text is extracted from the failure reports (word documents) by using the *docxpy* python package. The number

of representative words of each failure report is reduced to the most relevant ones by applying pre-processing techniques [24] such as: tokenization; reducing list of tokens; part of speech tagging; and lemmatization. Then, each failure report is represented as a vector in a word-space model. In particular, the Term Frequency-Inverse Document Frequency (TF-IDF) [24] is used in combination with Bag of Words (BoW) [24] to refine the list of words. BoW considers the raw frequency of that specific word within the report [24] and therefore selects the most frequent words. However, some words, such as *figure*, *failure* or *sensor*, are not helpful for distinguishing the group of reports. These non-informative words are then eliminated by using the TF-IDF approach [24] which considers how many times each word appears in one document and also how many times that word appears in all the documents of the corpus. Words that appear in all the documents being processed will be given a zero TF-IDF score.

These weights can be calculated by first finding the term frequency (tf) [24]:

$$\text{tf}(\text{word}) = \frac{\text{Number of times the word appears in document}}{\text{Total number of words in document}} \quad (1)$$

Next, the inverse document frequency term is needed (idf) [24]:

$$\text{idf}(\text{word}) = \log \left(\frac{\text{Total number of documents in corpus}}{\text{Number of documents containing the required word}} \right) \quad (2)$$

Finally, the TF-IDF score (which takes values in the interval [0,1]), is calculated as [24]:

$$\text{TF-IDF} = \text{tf}(\text{word}) \times \text{idf}(\text{word}) \quad (3)$$

The words with TF-IDF scores above a certain threshold are then used to represent each document as a vector. Once this vector representation is obtained, groups of reports belonging to different failure types can be then obtained by applying semi-supervised and unsupervised K-means clustering [25]. While in the unsupervised clustering the initial cluster centres are randomly allocated, in the semi-supervised clustering the initial cluster centres are assigned by selecting one report for each failure type. Once the K-means algorithm has been run to determine each cluster centre and the reports belonging to that cluster, the label of each cluster is manually extracted by selecting a single report within that cluster that is close to the identified cluster centre.

5 APPLICATION OF THE PROPOSED APPROACH

A TF-IDF score threshold of 0.0019 was set. The K-means implementation from the sklearn package [26] was used where the cluster number was set to 7. A brute-force algorithm was implemented to quantify the performance of K-means clustering. This performance was assessed in terms of “accuracy”, that is the ratio of correct failure type predictions to total predictions made. Since multiple classes are considered and each class has an unequal number of observations, the confusion matrix is also considered. These matrices display the count values of the correct and incorrect failure labels predictions, and they are defined such that rows display the expected class, while the columns represent the predicted class obtained with the clustering algorithm. The goal is to maximise the count values obtained on the main diagonal since they correspond to the total number of failures for that class.

For the unsupervised K-means clustering with 100 starting points, a maximum accuracy of 83.7% was observed, and a lowest of 70.1%. The clustering with the lowest accuracy is shown in Table 2.

Labels	L1	L2	L3	L4	L5	L6	L7
L1= (GSR, analog, pin)	12	0	0	0	12	0	0
L2= (GSR, ground, pin)	12	12	0	8	0	0	0
L3= (GSR, burnt, resistor)	0	0	16	0	0	0	0
L4= (accelerometer, ground, pin)	0	0	0	11	0	0	0
L5 = (accelerometer, power, pin)	0	0	0	11	0	0	0
L6= (humidity, power, pin)	0	0	0	0	0	18	0
L7= (temperature, ground, pin)	0	0	0	0	0	0	13

Table 2: Unsupervised clustering, confusion matrix with accuracy of 70.1%.

It can be observed that the failure types (GSR, analog, pin), (GSR, ground, pin), (accelerometer, ground, pin), and (accelerometer, power, pin) can be miss-clustered due to the similarity of the failure reports and to the reduced number of reports to learn from. In Table 3 it is shown that even when an accuracy of 83.7% is obtained, the failure type (accelerometer, power, pin) can still be entirely miss-clustered.

Labels	L1	L2	L3	L4	L5	L6	L7
L1= (GSR, analog, pin)	24	0	0	0	0	0	0
L2= (GSR, ground, pin)	0	24	0	0	0	0	0
L3= (GSR, burnt, resistor)	0	0	16	0	0	0	0
L4= (accelerometer, ground, pin)	0	0	0	11	0	0	0
L5 = (accelerometer, power, pin)	0	0	0	11	0	0	0
L6= (humidity, power, pin)	0	0	0	0	6	12	0
L7= (temperature, ground, pin)	0	2	0	0	0	0	11

Table 3: Unsupervised clustering, confusion matrix with accuracy of 83.7%.

These results could be improved by considering pre-knowledge on the labels and/or relationships between words at the TF-IDF stage, before running the clustering algorithms, or by increasing the number of available documents.

For example, when the initial cluster centre was set manually by assigning one failure report to each failure type, the accuracy was improved as shown in Table 4 and the (accelerometer, power, pin) was correctly clustered.

Labels	L1	L2	L3	L4	L5	L6	L7
L1= (GSR, analog, pin)	20	4	0	0	0	0	0
L2= (GSR, ground, pin)	0	24	0	0	0	0	0
L3= (GSR, burnt, resistor)	0	3	13	0	0	0	0
L4= (accelerometer, ground, pin)	0	0	0	2	9	0	0
L5 = (accelerometer, power, pin)	0	0	0	0	11	0	0
L6= (humidity, power, pin)	0	0	0	0	0	18	0
L7= (temperature, ground, pin)	0	2	0	0	0	0	11

Table 4: Semi-supervised clustering, confusion matrix obtained when initial cluster centres are assigned by specifying one report belonging to each cluster.

Therefore, when the labels are extracted automatically, some reports may be incorrectly labelled. As a result, this would reduce the performance of a self-supervised classification algorithm. Nonetheless, the overall performance of the proposed approach can still reach a certain target accuracy while at the same time reducing the setting up time by making the labelling process fully automatic or semi-automatic for the user.

6 CONCLUSIONS

An approach for extracting information obtained during failure investigations is proposed with the aim to facilitate the implementation of a self-supervised machine learning strategy that enables to detect if a monitoring device failed, and if so, to classify which sensor failed and the type of sensor failure.

Within the proposed approach, the process of extracting labels from failure reports, and therefore assigning labels to the corresponding measurements, is sped up by pre-processing the failure reports and applying Natural Language Processing (NLP) techniques to create a vector representation of each failure report in the word-space. The failure report vector representations are clustered together using K-means clustering, and a failure label is assigned to each cluster.

The applicability of the proposed approach was shown by analysing the reports collected when performing failure investigations of a low-cost health monitoring device. This application displays similar challenges encountered in SHM applications, such as: (i) the sensors employed record various quantities at different rates; (ii) the measurements are influenced by operational and environmental conditions; (iii) similar failure types can occur for the same sensor; (iv) only a limited dataset of recorded failures is available; and (v) the number of elements in the training dataset for each failure type is imbalanced. Seven types of failures were manually induced, and measurements with different sensors were recorded during operating and testing conditions. Failure reports for each failure investigated were written, and paired with the recorded data. A small dataset of 117 failures was produced. This limited dataset was characterised by four different faulty sensors, two of which displayed multiple failure types and an imbalanced number failure were considered for for each failure type.

It was shown that the proposed label extraction procedure when using unsupervised clustering can miss-cluster entirely one of the failure types even if yielding an overall high accuracy. As a result, this would reduce the performance of the self-supervised classification algorithms. Nonetheless, the overall performance of the proposed approach can still reach a certain target accuracy while at the same time reducing the setting up time by making the labelling process fully automatic or semi-automatic for the user. It was concluded that when dealing with small failure datasets, with unbalanced classes and similar failure types that the semi-supervised clustering procedure should be preferred.

Indeed, depending on the complexity of the failure reports, the extraction of the failure type labels using NLP strategies can lead to wrong labels assignment, with the risk of not including a particular failure type in the training dataset. Moreover, a failure type can potentially be wrongly identified in the failure report itself, and in fact it might not be supported by the features observed in the data. In turn, this will affect the capability of the proposed approach to detect and isolate the correct failure type for new, unseen data. This is of particular importance for SHM applications. The assessment of the quality of the features-label pairs for improving the training of the classification algorithm is the subject of current research investigations.

REFERENCES

- [1] C.R. Farrar, K. Worden, *Structural Health Monitoring: a Machine Learning Perspective*. Wiley, 2013.
- [2] R.J. Barthorpe, K. Worden, Emerging Trends in Optimal Structural Health Monitoring System Design: From Sensor Placement to System Evaluation. *Journal of Sensor and Actuator Networks*, **9(3)**, 1–31, 2003.
- [3] S. Patel, H. Park, P. Bonato, L. Chan and M. Rodgers, A review of wearable sensors and systems with application in rehabilitation. *J Neuroeng Rehabil.*, **9**, 1–21, 2012.
- [4] D. Dias, J. Paulo Silva Cunha, Wearable Health Devices - Vital Sign Monitoring, Systems and Technologies. *Sensors*, **8**, 1–28, 2018.
- [5] T.H. Yi, H.B. Huang, H.N. Li, Development of sensor validation methodologies for structural health monitoring: A comprehensive review. *Measurement*, **109**, 200–214, 2017.
- [6] Academy of Medical Royal Colleges, *Artificial Intelligence in Healthcare*. Academy of Medical Royal Colleges, 2019.
- [7] R. Dunia, J.S. Qin, E.F. Thomas, T.J. McAvoy, Identification of faulty sensors using principal component analysis. *AIChE Journal*, **42**, 2797–2812, 1996.
- [8] L. Van Eykeren, Q.P. Chu, Sensor fault detection and isolation for aircraft control systems by kinematic relations. *Control Engineering Practice*, **31**, 200–210, 2014.
- [9] B.M. de Silva, J. Callaham, J. Jonker, N. Goebel, J. Klemisch, D. McDonald, N. Hicks, J. Nathan Kutz, S.L. Brunton, A.Y. Aravkin, Physics-informed machine learning for sensor fault detection with flight test data. *arXiv*, 2020.
- [10] H. Zhang and J. Liu and N. Kato, Threshold Tuning-Based Wearable Sensor Fault Detection for Reliable Medical Monitoring Using Bayesian Network Model. *IEEE Systems Journal*, **12**, 1886–1896, 2018.
- [11] M.I. Friswell and D.J. Inman, Sensor Validation for Smart Structures. *Journal of Intelligent Material Systems and Structures*, **10**, 973–982, 1999.
- [12] G. Kerschen, P. De Boe, J. Golinval, K. Worden, Sensor validation using principal component analysis. *Smart Materials and Structures*, **14**, 36–42, 2004.
- [13] Da. Li, Y. Wang, J. Wang, C. Wang, Y. Duan, Recent advances in sensor fault diagnosis: A review. *Sensors and Actuators A: Physical*, **309**, 2020.
- [14] H. Sohn, Effects of environmental and operational variability on structural health monitoring. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **365**, 539–560, 2007.
- [15] J. Kullaa, Eliminating Environmental or Operational Influences in Structural Health Monitoring using the Missing Data Analysis. *Journal of Intelligent Material Systems and Structures*, **20**, 1381–1390, 2009.

- [16] J. Kullaa, Distinguishing between sensor fault, structural damage, and environmental or operational effects in structural health monitoring. *Mechanical Systems and Signal Processing*, **25**, 2976–2989, 2011.
- [17] J. Kullaa, Robust damage detection in the time domain using Bayesian virtual sensing with noise reduction and environmental effect elimination capabilities. *Journal of Sound and Vibration*, **473**, 2020.
- [18] E.J. Cross, K. Worden, Q. Chen, Cointegration: a novel approach for the removal of environmental trends in structural health monitoring data. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **467**, 2712–2732, 2011.
- [19] L.D. Avendano Valencia, E.N. Chatzi, D. Tcherniak, Gaussian process models for mitigation of operational variability in the structural health monitoring of wind turbines. *Mechanical Systems and Signal Processing*, **142**, 2020.
- [20] L.A. Bull, T.J. Rogers, C. Wickramarachchi, E.J. Cross, K. Worden, N. Dervilis, Probabilistic active learning: An online framework for structural health monitoring. *Mechanical Systems and Signal Processing*, **134**, 2019.
- [21] M.L. Perry, *Electronic Failure Analysis Handbook: Techniques and Applications for Electronic and Electrical Packages, Components, and Assemblies*. McGRAW-HILL, 1999.
- [22] J.S. Otegui, *Failure Analysis: Fundamentals and Applications in Mechanical Components*. Springer, 2014.
- [23] N. Jiang, L. Zhang, Z.Q. Liu, L. Sun, W.M. Long, P. He, M.Y. Xiong, M. Zhao, Reliability issues of lead-free solder joints in electronic devices. *Science and technology of advanced materials*, **20(1)**, 876–901, 2019.
- [24] D. Jurafsky, J.H. Martin, *Speech and language processing - an introduction to natural language processing, computational linguistics, and speech recognition*. Prentice Hall series in artificial intelligence, 2000.
- [25] M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, 2006.
- [26] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830, 2011.

ON THE INVESTIGATION OF UTILITY FUNCTIONS ON OPTIMAL SENSOR LOCATIONS

Felipe Igea¹, Manolis N. Chatzis¹, and Alice Cicirello²

¹ University of Oxford
Department of Engineering Science, Oxford, UK
{DPhil Candidate, Associate Professor}
e-mail: {felipe.igea, manolis.chatzis}@eng.ox.ac.uk

² Delft University of Technology
Faculty of Civil Engineering and Geosciences, Delft, Netherlands
Associate Professor
a.cicirello@tudelft.nl

Abstract

Structural Health Monitoring uses data collected from sensors placed on structures to determine their operating condition and whether maintenance is required. Often, optimal sensor placement strategies are used to find the optimal locations for the identification of their modal properties, structural parameters and/or abnormal behaviours under the influence of model and measurement uncertainty. An approach that has been frequently used to solve the problem of sensor placement is the Bayesian experimental design. This approach chooses the locations using the data measured by the sensors to reduce the prior uncertainty of the parameters that are being inferred. The Bayesian experimental design minimizes the uncertainty of the parameters to be inferred through the use of metrics called utility functions. Most of these metrics are based on functions of the posterior distribution. In this paper, the use of three utility functions (Bayesian D-posterior precision, Bayesian A-posterior precision, and Expected Information Gain) is investigated for the problem of sensor placement.

The case study chosen consists of a beam with translational and rotational springs connected to the ground subject to an impulsive load. The goal of the analysis is to select the most informative position of a sensor in order to update the distribution of two uncertain physical parameters of the beam based on natural frequencies extracted using the Eigensystem Realization Algorithm. It is shown that for the case investigated, the three utility functions yield the same optimal sensor location.

Keywords: Optimal Sensor Placement, Uncertainty Quantification, Structural Health Monitoring.

1 INTRODUCTION

Structural Health Monitoring (SHM) often focuses on non-intrusive structure damage detection [1]. It can be used to provide early warnings on the health status of engineering systems. The equipment required for implementing SHM, includes sensors and data acquisition systems. The real time information obtained from the sensors has to be post-processed and statistical procedures are implemented to detect anomalies and suggest preventive actions [1,2]. Technological advances in sensor monitoring allow the development of optimal sensor strategies that have made SHM cost effective and easier to implement.

Structural parameters are usually inferred from the sensorial data (such as velocity or acceleration measurements), especially the modal parameters (natural frequency, damping ratio and mode shape). The inferred parameters are then used to assess the acceptability of the models of structures and to evaluate the structures' condition. For some cases, local forms of damage may be identified by a shift in the modal properties [3]. The position of the sensor can strongly influence the inference on the structural parameters. This has led to the widespread development of optimal sensor strategy techniques [4]. Broadly speaking, the sensor placement framework methods can be split into methods based on information theory or non-information methods [4]. The non-information-based methods are not discussed in this paper, however, more information on these techniques can be found in [4]. Work based on information theory heavily relies on the application and development of the general Bayesian framework [5]. This framework was proposed for system identification in [5,6] and it has been consequently extended to the problem of sensor placement [7–10]. The main objective is the selection of the location and number of sensors that maximises the information needed to estimate the uncertain parameters [8]. The research challenges linked to these approaches are the definition of the metric to be used to assess the different configurations of sensors (number and location of the sensors) and the choice of the most adequate optimisation technique [11,12].

In this paper a Bayesian experimental design framework [13] is used to solve the sensor placement problem. Within this framework, the number and locations of the sensors are chosen by using the data obtained from the sensors to reduce the prior uncertainty on the parameters to be inferred. Therefore, the framework's focus is the minimization of the uncertainty of certain physical parameters of interest to the practitioner by comparing different metrics, the so-called utility functions. Two physical parameters of a beam attached to ground using translational and rotational springs subject to an impulsive load have to be inferred by using a single sensor. The beam is investigated by building a Finite Element model. Numerical simulations of the dynamic response (velocity signal) at different locations are used to obtain numerical 'measurements' of possible sensor locations. An intermediate step requires the post-processing of the numerical 'measurements' to obtain the modal parameters that are subsequently used as the data used to reduce the model parametric uncertainty via Bayesian model updating. This is achieved by using the Eigensystem Realization Algorithm. Model updating is then used to obtain the posterior probability density function of the parameters to be inferred, having assigned a uniform prior distribution and applying Monte Carlo sampling-based strategies. The obtained posterior is used to evaluate a utility function that is then used to select the optimal sensor location. Three utility functions are investigated.

2 BAYESIAN OPTIMAL DESIGN FRAMEWORK

Bayesian optimal design [13] allows the designation of resources required to obtain information for reduction of systematic error, inference of unknown parameters (i.e., reduction of prior uncertainty), obtaining future predictions and the comparison of models chosen to represent a

system [13]. The framework's objective is the maximisation of the information obtained from a set of measurements for the inference of the unknown parameters of the model used to describe the physical system [13]. The choice of the optimal design improves parameter inference and reduces the experimental costs.

Lindley proposed a unifying theory of Bayesian optimal design in [14]. The definition of the best possible design given a set of objectives and restrictions is described by a utility function. The maximization of this function is used to choose the possible design that measures how well the set of objectives and restrictions are obeyed [15].

One of the major challenges in Bayesian optimal design methods is the reduction of their high computational cost incurred in the calculation of their utility functions [13]. This is because the utility functions require the knowledge of the posterior distribution of the parameters to be inferred. These distributions are dependent on the set of measurements available and therefore are different when different designs are considered.

2.1 Bayesian Framework

Probability density functions are used to model the uncertain model parameters in the Bayesian inference framework [5,16]. The prior knowledge on the uncertain parameters before any measurements or data is obtained, is described by the prior density function $P(\boldsymbol{\theta})$. The likelihood function $p(\mathbf{y} | \boldsymbol{\theta})$, is normally assumed to follow a specific distribution (e.g., Gaussian). The $p(\mathbf{y} | \boldsymbol{\theta})$ measures the degree of suitability of the model to justify the obtained measurements. The denominator $p(\mathbf{y})$ of eq.(1) below is the evidence pdf and normalizes the pdf of the posterior. If the above described pdfs are known, the eq.(1) can be used to calculate the so-called posterior distribution $p(\boldsymbol{\theta} | \mathbf{y})$:

$$p(\boldsymbol{\theta} | \mathbf{y}) = \frac{p(\boldsymbol{\theta})p(\mathbf{y} | \boldsymbol{\theta})}{p(\mathbf{y})} \quad (1)$$

The posterior distribution obtained can then be used to determine the utility function. Hence, it is important to obtain accurate estimations of both location (median or mean) and scale (interquartile range or standard deviation) of the posterior [13]. Most frequently, it is not possible to express the posterior distributions with a closed form, so computational methodologies are used to obtain samples from the posterior or to approximate it [16–21]. In this work, the sampling-based model updating techniques are used. Specifically, the Sequential Monte Carlo (SMC) [16] sampling and the Transitional Markov Chain Monte Carlo (TMCMC) [20] are chosen to infer the two physical parameters of the case study investigated.

2.2 Bayesian Utility Functions

Many different utility functions have been developed for inferring parameters of a model [13]. Metrics that quantify the performance of experiments are obtained by using a set of utility functions that are maximized (or minimized) with the objective of identifying the optimal experiment [13]. Three utility functions are reviewed in what follows.

A well-known utility function is expressed as the inverse of the determinant of the posterior covariance matrix [13]. This utility function also known as the Bayesian D-posterior precision maximises the posterior precision of the model parameters to be inferred and it is given by [13]:

$$U_D(\mathbf{d}, \mathbf{y}) = \frac{1}{\det(\text{cov}(\boldsymbol{\theta}|\mathbf{d}, \mathbf{y}))} \quad (2)$$

Where \mathbf{d} is the vector that represents the experimental design to be optimized (e.g. the sensor positions for a given number of sensors).

Another useful utility function similar to the Bayesian D-posterior precision is given by the inverse of the trace of the posterior covariance matrix [15]. This utility function also known as the ‘Bayesian A-posterior precision’ maximises the marginal posterior precision of the model parameters to be inferred and it is given by [15]:

$$U_A(\mathbf{d}, \mathbf{y}) = \frac{1}{\text{trace}(\text{cov}(\boldsymbol{\theta}|\mathbf{d}, \mathbf{y}))} \quad (3)$$

Alternatively, the utility function may be expressed as the expected Kullback–Leibler (KL) divergence from the posterior distribution to the prior distribution [14]. The expected KL divergence utility function is also known as Expected Information Gain (EIG) over the parameters to be inferred [22], and it is expressed as:

$$U_{EIG}(\mathbf{d}) = E_{\mathbf{y}} \left[D_{KL} \left(p_{\boldsymbol{\theta}|\mathbf{y}} \| p_{\boldsymbol{\theta}} \right) \right] = \int_{\mathcal{Y}} \int_{\mathcal{Q}} p(\boldsymbol{\theta}|\mathbf{y}) \log \left(\frac{p(\boldsymbol{\theta}|\mathbf{y})}{p(\boldsymbol{\theta})} \right) p(\mathbf{y}) d\mathbf{y} d\boldsymbol{\theta} \quad (4)$$

Where $E_{\mathbf{y}}$ is the expectation with respect to the measurements \mathbf{y} , $D_{KL} \left(p_{\boldsymbol{\theta}|\mathbf{y}} \| p_{\boldsymbol{\theta}} \right)$ is the KL-divergence from the posterior distribution to the prior distribution, \mathcal{Y} and \mathcal{Q} are the support of the measurements \mathbf{y} and the parameters to be inferred $\boldsymbol{\theta}$ respectively.

The EIG can be interpreted as a non-linear generalization of the Bayesian D-optimal utility function [23]. It has been found [24] that this metric can be approximated using a Monte Carlo approach:

$$\tilde{U}(\mathbf{d}) = \frac{1}{N_{out}} \sum_{i=1}^{N_{out}} \left\{ \ln \left[p(\mathbf{y}^i | \boldsymbol{\theta}^i, \mathbf{d}) \right] - \ln \left[p(\mathbf{y}^i | \mathbf{d}) \right] \right\} \quad (5)$$

$$p(\mathbf{y}^i | \mathbf{d}) \approx \frac{1}{N_{in}} \sum_{j=1}^{N_{in}} p(\mathbf{y}^i | \boldsymbol{\theta}^j, \mathbf{d}) \quad (6)$$

Where N_{out} is the number of samples used in the outer loop and N_{in} is the number of samples used in the inner loop of the Monte Carlo approximations. The samples are obtained from the prior distribution and the likelihood is evaluated for these samples.

The number of likelihood function evaluations can be reduced if $N = N_{in} = N_{out}$, so that [24,25]:

$$\hat{U}(\mathbf{d}) = \frac{1}{N} \sum_{i=1}^N \left\{ \ln \left[p(\mathbf{y}^i | \boldsymbol{\theta}^i, \mathbf{d}) \right] - \ln \left(\frac{1}{N} \sum_{j=1}^N p(\mathbf{y}^i | \boldsymbol{\theta}^j, \mathbf{d}) \right) \right\} \quad (7)$$

However, this result is a biased estimate of the EIG [24,25]. A large number of samples may be required if the prior assumed has a large support at regions of low probability density, as this results in arithmetic underflow [26].

Another way to calculate the EIG is by calculating the difference between the differential entropy of the prior $h(\boldsymbol{\theta})$ and the differential entropy of the posterior $h(\boldsymbol{\theta} | \mathbf{y}, \mathbf{d})$ [15]:

$$U_{EIG,2}(\mathbf{d}) = I(\boldsymbol{\theta}, \mathbf{y}) = h(\boldsymbol{\theta}) - h(\boldsymbol{\theta} | \mathbf{y}, \mathbf{d}) \quad (8)$$

The calculation of the differential entropy using samples from the posterior distribution can be approximated using the recursive copula splitting approach given in [27].

3 NUMERICAL RESULTS

A beam connected to the ground via two sets of translational and rotational spring positioned along the length of the beam, as shown on fig.1, is investigated in this paper. This simple case study has been chosen as it can represent a variety of practical situations where a component is attached to some fixtures, but there are uncertainties that may due to its assembly, boundary conditions and/or manufacture. In particular, in this case, the location of the first set of springs and the magnitude of the rotational spring are investigated. A prior distribution is assigned to each of these two parameters. The goal of the analysis is to select the most informative position of a sensor in order to update the distribution of these two uncertain parameters. The utility functions defined in section 2 are used to assess the optimal position of the sensor.

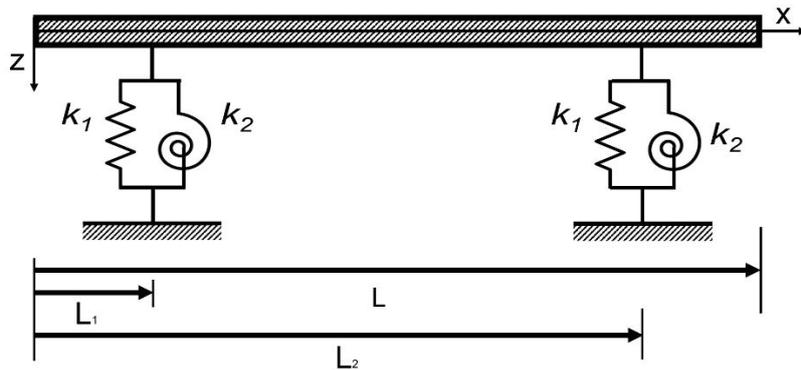


Figure 1: Beam attached to ground by translational and rotational springs.

The following geometric and material properties were used: L (length)=0.6m; b (base)=0.04m; h (height)=0.003m; ρ (density) =8000kg/m³; E (Young's modulus) =100GPa; k_1 (translational spring stiffness) = 1kN/m; k_2 (rotational spring stiffness) = 101.7Nm/rad; L_1 (length to springs) = 0.181m; L_2 (length to springs) = 0.4m. Modal damping was introduced into the system ($\eta=0.01$ for all modes). A force F (triangular pulse of length 10ms and maximum amplitude of 50N) applied at length L is used to excite the beam. The parameters to be inferred are the stiffness k_2 of both rotational springs and the location L_1 of the first rotational spring.

A Finite Element (FE) model is used to calculate the transversal velocity signals of the beam at several locations, to investigate the effects of the position of a single sensor on the utility functions. In particular, a 2-dimensional Euler-Bernoulli beam model is considered. This is discretized uniformly using 200 Euler-Bernoulli beam FEs with 2 degrees of freedom per node. Moreover, to simulate experimental conditions, for each transversal velocity signal measure at each node point, ten different realizations are created contaminating each signal using a white Gaussian noise with a noise to signal ratio (rms) of 5%.

The numerically contaminated velocity signals obtained at each possible sensor location (where the locations available are the ones at each node of the FE system) are post-processed using the Eigensystem Realization Algorithm (ERA) [28] to calculate the modal properties. Therefore, it was required to apply ERA 200 times to cover all the possible sensor locations in the system. These modal properties are then used as the data observed in the likelihood function. The likelihood function is then approximated by using the kernel smoothing function (`ksdensity` function of MATLAB [29]) on the set of modal properties obtained from ERA using the 10 different realizations of the contaminated velocity signals for each possible sensor location. Uniform priors were used for both the stiffness (100Nm/rad to 103 Nm/rad) of the rotational springs and location (0.17m to 0.19m) of the rotational spring. The joint posterior distribution of k_2 and L_1 is calculated using two Bayesian model updating techniques [16]: Sequential Monte Carlo (SMC) sampling and the Transitional Markov Chain Monte Carlo (TMCMC). In the SMC sampling approach [16] the samples obtained from the prior were re-used in all possible sensor locations to reduce the amount of forward simulations needed and to investigate how the bias resulting from this approach could affect the calculation of the utility functions. The results obtained with this implementation of SMC were compared with the result obtained using the unbiased TMCMC [20] that required new simulations each time a possible sensor location was considered. While SMC required 20,000 forward simulations to obtain acceptable estimations of the posterior distribution, the TMCMC required only 6,000 simulations but each time a new sensor position was considered the forward simulations could not be reused.

Figures 2, 3 show the precision values obtained for the Bayesian D-posterior precision and Bayesian A-posterior precision utility functions as a function of a sensor location along the length of the beam when using SMC and TMCMC.

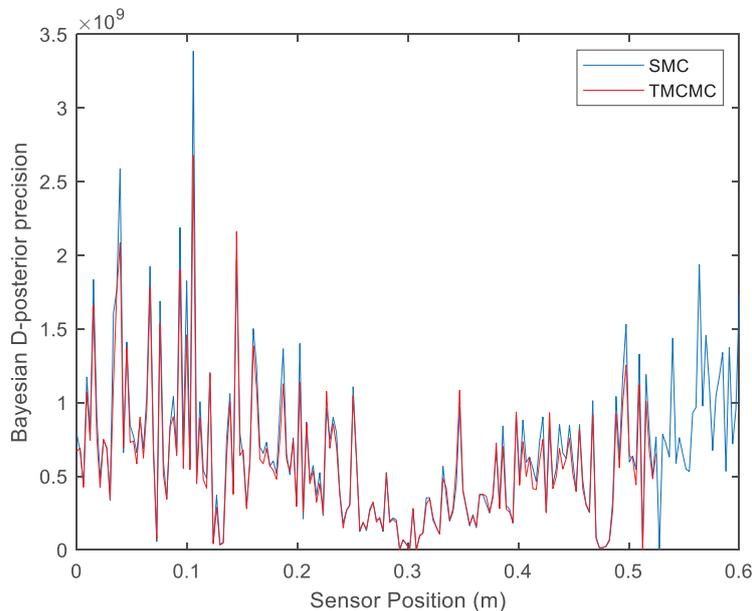


Figure 2: Bayesian D-posterior precision values vs sensor location.

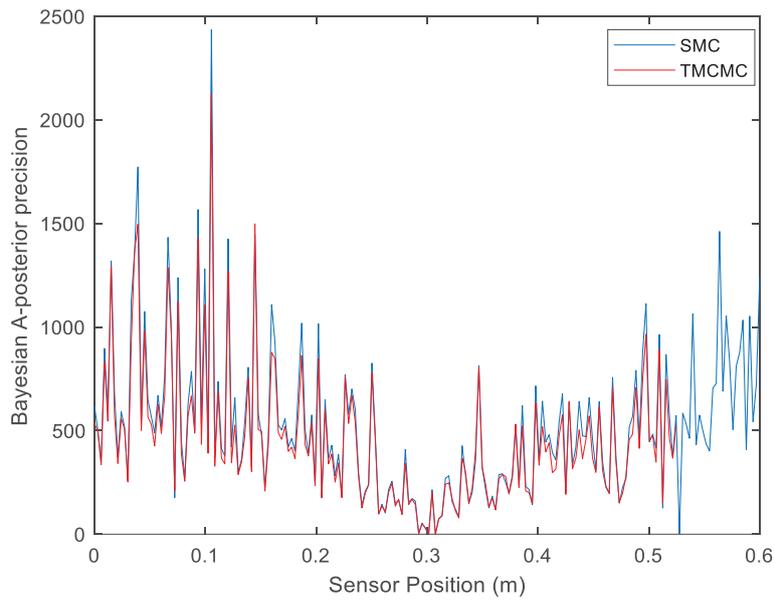


Figure 3: Bayesian A-posterior precision values vs sensor location.

The results obtained with the EIG utility function are shown in figure 4. These results were obtained by using the recursive copula splitting approach from [27] as using the Monte Carlo approximation shown in eq.(7) resulted in evaluating likelihoods at supports of low probability density which lead to arithmetic underflow.

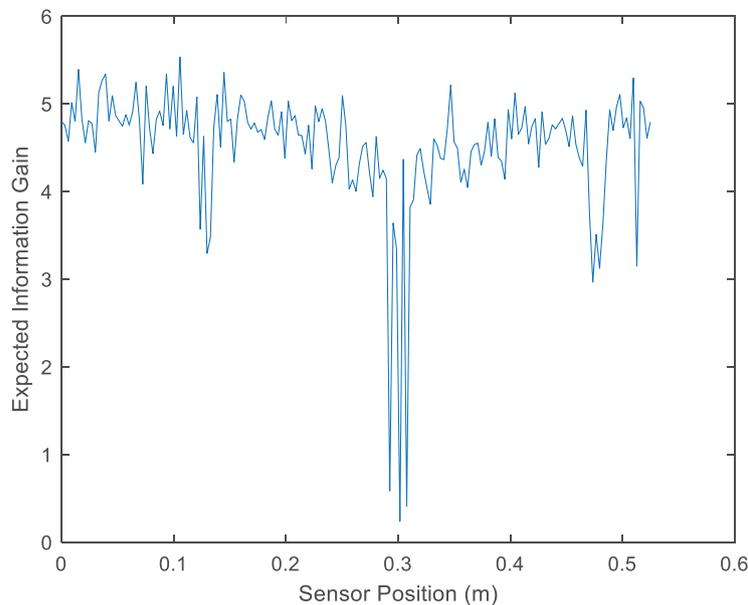


Figure 4: Expected information gain vs sensor location.

It can be observed that the three utility functions used have identified the same best sensor location - that is 0.106m. Locations where the utility values were low were close to nodal points and hence the modal properties resulting from ERA were less accurate. For this beam, the results obtained using the three different utility functions investigated, have been found to be

similar. However, it is expected that if a large number of physical parameters would have been inferred, the utility functions may have shown dissimilar results as the sensor location that maximises the joint posterior precision may have not been the same as the sensor location that maximises the marginal posterior precision.

It was also shown that for this case study the SMC and TMCMC provide similar results. The discrepancies found in the values of the utility function are largely due to the bias introduced by reusing samples from the prior and the choice of using a sequential importance sampling algorithm instead of a sequential importance resampling algorithm. If a sequential importance resampling algorithm had been chosen a lower bias would have been introduced in exchange for a higher computational cost.

4 CONCLUSIONS

The optimal sensor placement for the identification of two physical parameters of a beam attached to ground by translational and rotational springs has been investigated by considering three utility functions: Bayesian D-posterior precision and Bayesian A-posterior precision, and Expected Information Gain. It was shown that these utility functions led to the same best sensor location. As expected, poor values of the utility function were found for locations close to nodal points, as the modal properties estimated by ERA were less accurate. This result is expected as the measurements obtained at nodal points would not have as much information as other points along the beam system.

The utility functions chosen require the calculation of the posterior distribution. Therefore, the computational cost is reliant on the Bayesian inference technique being used. However, the choice of the inference technique usually shows a trade-off between computational cost and accuracy. Reusing samples, as in SMC techniques, may limit the amount of likelihood evaluations but this is at the risk of not evaluating samples close to regions of high probability densities. However, it was found that SMC and TMCMC lead to the same results for the case under investigation. The current challenge for the Bayesian optimal design approach would be the development of a fast inference technique that estimates the posterior at a limited computational cost.

REFERENCES

- [1] F. Magalhães, Á. Cunha, E. Caetano, Online automatic identification of the modal parameters of a long span arch bridge, *Mechanical Systems and Signal Processing*. 23 (2009) 316–329. <https://doi.org/10.1016/j.ymssp.2008.05.003>.
- [2] C.R. Farrar, K. Worden, *Structural health monitoring : a machine learning perspective*, Wiley, Chichester, 2013.
- [3] J.J. Moughty, J.R. Casas, A State of the Art Review of Modal-Based Damage Detection in Bridges: Development, Challenges, and Solutions, *Applied Sciences*. 7 (2017) 510. <https://doi.org/10.3390/app7050510>.
- [4] D. Li, Herausgeber: Claus-Peter Fritzen *Sensor Placement Methods and Evaluation Criteria in Structural Health Monitoring*, 2011.
- [5] J.L. Beck, L.S. Katafygiotis, Updating Models and Their Uncertainties. I: Bayesian Statistical Framework, *Journal of Engineering Mechanics*. 124 (1998) 455–461. [https://doi.org/10.1061/\(ASCE\)0733-9399\(1998\)124:4\(455\)](https://doi.org/10.1061/(ASCE)0733-9399(1998)124:4(455)).

- [6] L.S. Katafygiotis, J.L. Beck, Updating Models and Their Uncertainties. II: Model Identifiability, *Journal of Engineering Mechanics*. 124 (1998) 463–467. [https://doi.org/10.1061/\(ASCE\)0733-9399\(1998\)124:4\(463\)](https://doi.org/10.1061/(ASCE)0733-9399(1998)124:4(463)).
- [7] C. Papadimitriou, Optimal sensor placement methodology for parametric identification of structural systems, *Journal of Sound and Vibration*. 278 (2004) 923–947. <https://doi.org/10.1016/j.jsv.2003.10.063>.
- [8] C. Papadimitriou, G. Lombaert, The effect of prediction error correlation on optimal sensor placement in structural dynamics, *Mechanical Systems and Signal Processing*. 28 (2012) 105–127. <https://doi.org/10.1016/j.ymsp.2011.05.019>.
- [9] C. Argyris, C. Papadimitriou, G. Lombaert, Optimal sensor placement for response predictions using local and global methods, in: *Conference Proceedings of the Society for Experimental Mechanics Series*, Springer New York LLC, 2020: pp. 229–236. https://doi.org/10.1007/978-3-030-12075-7_26.
- [10] C. Argyris, C. Papadimitriou, G. Samaey, G. Lombaert, A unified sampling-based framework for optimal sensor placement considering parameter and prediction inference, *Mechanical Systems and Signal Processing*. 161 (2021) 107950. <https://doi.org/10.1016/j.ymsp.2021.107950>.
- [11] W. Ostachowicz, R. Soman, P. Malinowski, Optimization of sensor placement for structural health monitoring: a review, *Structural Health Monitoring*. 18 (2019) 963–988. <https://doi.org/10.1177/1475921719825601>.
- [12] Y. Tan, L. Zhang, Computational methodologies for optimal sensor placement in structural health monitoring: A review, *Structural Health Monitoring*. (2019) 1–22. <https://doi.org/10.1177/1475921719877579>.
- [13] E.G. Ryan, C.C. Drovandi, J.M. McGree, A.N. Pettitt, A Review of Modern Computational Algorithms for Bayesian Optimal Design, *International Statistical Review*. 84 (2016) 128–154. <https://doi.org/10.1111/insr.12107>.
- [14] D. v. Lindley, *Bayesian Statistics*, Society for Industrial and Applied Mathematics, 1972. <https://doi.org/10.1137/1.9781611970654>.
- [15] C. ben Issaid, *Bayesian Optimal Experimental Design Using Multilevel Monte Carlo*, 2015. <https://repository.kaust.edu.sa/handle/10754/552705> (accessed June 8, 2021).
- [16] A. Lye, A. Cicirello, E. Patelli, Sampling methods for solving Bayesian model updating problems: A tutorial, *Mechanical Systems and Signal Processing*. 159 (2021) 107760. <https://doi.org/10.1016/j.ymsp.2021.107760>.
- [17] E. Simoen, G. de Roeck, G. Lombaert, Dealing with uncertainty in model updating for damage assessment: A review, *Mechanical Systems and Signal Processing*. 56 (2015) 123–149. <https://doi.org/10.1016/j.ymsp.2014.11.001>.
- [18] J.L. Beck, S.-K. Au, Bayesian Updating of Structural Models and Reliability using Markov Chain Monte Carlo Simulation, (n.d.). <https://doi.org/10.1061/ASCE0733-93992002128:4380>.
- [19] H.-F. Lam, J.-H. Yang, S.-K. Au, Markov chain Monte Carlo-based Bayesian method for structural model updating and damage detection, *Structural Control and Health Monitoring*. 25 (2018) e2140. <https://doi.org/10.1002/stc.2140>.
- [20] J. Ching, Y.-C. Chen, Transitional Markov chain Monte Carlo method for Bayesian model updating, model class selection, and model averaging., *Journal of Engineering Mechanics*. 133 (2007) 816. [https://doi.org/10.1061/\(ASCE\)0733-9399\(2007\)133:7\(816\)](https://doi.org/10.1061/(ASCE)0733-9399(2007)133:7(816)).
- [21] D.M. Blei, A. Kucukelbir, J.D. McAuliffe, Variational Inference: A Review for Statisticians, *Journal of the American Statistical Association*. 112 (2017) 859–877. <https://doi.org/10.1080/01621459.2017.1285773>.

- [22] Z. Xu, Q. Liao, Gaussian process based expected information gain computation for bayesian optimal design, *Entropy*. 22 (2020) 258. <https://doi.org/10.3390/e22020258>.
- [23] K. Chaloner, I. Verdinelli, Bayesian experimental design: A review, *Statistical Science*. 10 (1995) 273–304. <https://doi.org/10.1214/ss/1177009939>.
- [24] X. Huan, Y.M. Marzouk, Simulation-based optimal Bayesian experimental design for nonlinear systems, *Journal of Computational Physics*. 232 (2013) 288–317. <https://doi.org/10.1016/j.jcp.2012.08.013>.
- [25] K.J. Ryan, Estimating Expected Information Gains for Experimental Designs with Application to the Random Fatigue-Limit Model, *Journal of Computational and Graphical Statistics*. 12 (2003) 585–603. <https://doi.org/10.1198/1061860032012>.
- [26] J. Beck, B.M. Dia, L.F.R. Espath, Q. Long, R. Tempone, Fast Bayesian experimental design: Laplace-based importance sampling for the expected information gain, *Computer Methods in Applied Mechanics and Engineering*. 334 (2018) 523–553. <https://doi.org/10.1016/j.cma.2018.01.053>.
- [27] G. Ariel, Y. Louzoun, Estimating differential entropy using recursive copula splitting, *Entropy*. 22 (2020) 236. <https://doi.org/10.3390/e22020236>.
- [28] J.N. Juang, R.S. Pappa, An eigensystem realization algorithm for modal parameter identification and model reduction, *Journal of Guidance, Control, and Dynamics*. 8 (1985) 620–627. <https://doi.org/10.2514/3.20031>.
- [29] MATLAB, (2020).

OPTIMAL SENSOR PLACEMENT IN DISTRICT HEATING NETWORKS FOR BAYESIAN INFERENCE OF UNCERTAIN DEMANDS

A. Matei¹, A. Bott², L. Rehlich¹, F. Steinke² and S. Ulbrich¹

¹ Technical University of Darmstadt, Department of Mathematics, Research Group Optimization
Dolivostraße 15, 64293 Darmstadt, Germany
e-mail for correspondence: matei@mathematik.tu-darmstadt.de

² Technical University of Darmstadt, Department of Electrical Engineering and Information
Technology, Energy Information Networks and Systems Lab
Landgraf-Georg Straße 4, 64289, Darmstadt, Germany
e-mail for correspondence: andreas.bott@eins.tu-darmstadt.de

Keywords: heating networks, sensor placement, Bayesian inference, demand estimation

Abstract. *District heating networks have traditionally been designed and operated as distribution networks supplied by few central heat production units. In order to reduce emissions in the heating sector feed-in from smaller decentralized units and industrial waste heat is becoming ever more important. This transition requires a more detailed monitoring of the network state, which can be achieved either by installing a huge number of additional sensors in the grid or by a state estimation based on a few additional measurements. However, the uncertain heat consumption generated by consumers presents a major challenge in this endeavor. In this paper we propose a model-based approach for optimal sensor placement in district heating networks in order to minimize the uncertainty in the demand values which are estimated by solving a Bayesian inverse problem. The optimization scheme is designed to yield a fair compromise between the desired information gain and the costs for installing the chosen sensors. A steady-state model is employed to estimate temperatures, mass flows and pressures of network components given the mean demand and the initial pressure generated by the heating plant. Our approach is applied to a real-sized district heating network using actual consumption distributions as given prior in order to validate the model and to prove scalability.*

1 INTRODUCTION

District heating networks are closed looping systems in the sense that water is pumped from heating plants towards consumers through a pipe system and flows back through parallel laid pipes. Energy is transmitted by heating up the water at the heating plants and cooling it down at the consumer's place. Within the grid the energy flow through each element is not predetermined by the network operator but is a result of the energy extracted by the consumers. In order to ensure security of supply and minimize energy losses in the grid, temperature and pressure at the heating plant have to be adjusted accordingly.

Future district heating networks will be characterized by lower temperature levels and additional decentralized feed-in [16] making it considerably more difficult to ensure, that no grid element is overloaded. Therefore additional information about the network state and consequently the consumer's actual consumption have to be obtained. However, constantly measuring the demand at each consumer's place is not practically feasible. The large number of additional sensors would not only mean high investment cost but also a high on-site electricity demand. Moreover measurements directly at the consumers may be misleading, due to the way they are connected to the grid. Usually the main pipes are laid under the roads and smaller pipes connect these with the heat-exchange-stations inside the buildings. If the consumption changes drastically or is close to zero the network state inside this connection pipes is not representative for the main pipes in the grid.

Alternatively, the heat consumption can be predicted or estimated based on external parameters [6]. These estimations naturally inherit some kind of uncertainty affecting the model prediction [18]. Hence, we say that uncertainty propagates from the consumption parameters to the network's state via the parameter-to-observable map. In this paper we propose a model-based approach to place a small number of sensors at optimal positions in the district heating network in order to minimize the variance of the estimated demand values. We claim that a better knowledge about the consumer's consumption eventually leads to more precise state predictions in the whole network. In order to quantify the uncertainty of both the demand estimation and the model's prediction we use the linearized parameter-to-observable map and a Bayesian viewpoint.

Optimal sensor placement is a broad field of research. It often appears in the context of optimal experimental design in the literature [2, 17, 10, 12, 21]. Probabilistic sensitivity-based approaches [3, 14, 19] and Bayesian inference-based perspectives [1, 2, 11] are mainly used as a tool to maximize the information gain obtained from optimally positioned sensors at low cost. A topic that is closely related to our question is leakage detection. A review paper on leakage detection methods in district heating networks is given by [24]. In [5] a distributed demand response approach based on augmented Lagrangian methods to optimize the heating demand with minimal private information exchange is developed. To the best of our knowledge, the sensor placement problem for variance-minimal demand estimation has not been applied to district heating networks so far.

The paper is structured as follows. In Section 2 we introduce a heating model which maps the consumer's demand onto the network's state. A Bayesian inference approach for model-based optimal experimental design is applied to our setting in Section 3. Numerical results for a real-sized district heating network are presented in Section 4. We end the paper with a short discussion of the results and a conclusion.

2 HEAT MODEL EQUATIONS

For practical purposes we are interested in the pressures and temperatures at given points in the network as well as the amount of water flowing through the pipes. Therefore, an operator $(\mathbf{y}, \boldsymbol{\theta}, \boldsymbol{\eta}) \mapsto \mathbf{e}(\mathbf{y}, \boldsymbol{\theta}, \boldsymbol{\eta})$ is constructed which couples the consumer demands $\boldsymbol{\theta}$, the network state \mathbf{y} consisting of temperatures \mathbf{T} , pressures \mathbf{p} and mass flows $\dot{\mathbf{m}}$, and set-point values $\boldsymbol{\eta}$. In the following we describe the different components of this operator \mathbf{e} . The structure of district heating networks can be described and implemented as a graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ with nodes \mathbf{V} and directed edges $\mathbf{E} \subseteq \mathbf{V} \times \mathbf{V}$. In this setting, the network state is defined according to the edges and nodes of the graph:

$$\mathbf{y} := (p_i, T_i, T_{kl}^{\text{end}}, T_{kl}^{\text{start}}, \dot{m}_{kl}), \quad \text{for all } i \in \mathbf{V} \text{ and } (k, l) \in \mathbf{E}, \quad (1)$$

where p_i, T_i are the pressure and the temperature in node i , and $T_{kl}^{\text{end}}, T_{kl}^{\text{start}}, \dot{m}_{kl}$ denote the water temperature at the end respectively the start of edge (k, l) and the mass flow on that edge. These kinds of models are commonly used to analyze different aspects of district heating networks [9, 15, 23, 4, 7]. In this paper, we assume steady state conditions and neglect time delays in the network. Since \mathbf{G} is directed, we can assign a nominal flow direction to each edge. For an edge $(i, j) \in \mathbf{E}$ the nominal flow direction is from node i to node j , meaning that $\dot{m}_{ij} \geq 0$ if water flows from node i towards node j . The superscripts *start* and *end* should be understood in the sense of this nominal flow direction. Per definition it follows that $\dot{m}_{ij} = -\dot{m}_{ji}$ and $T_{ij}^{\text{end}} = T_{ji}^{\text{start}}$. We only investigate treelike networks with one heating plant, in which the mass flow directions cannot change. Therefore, we deliberately choose the nominal flow direction in such a way, that only positive values for the mass flow occur, i.e., if $(i, j) \in \mathbf{E}$ then $\dot{m}_{ij} \geq 0$.

Let $\mathbf{N}_i := \{j \in \mathbf{V} \mid (i, j) \in \mathbf{E} \text{ or } (j, i) \in \mathbf{E}\}$ be the set of nodes in the neighborhood of i which are connected to node i by an edge $(i, j) \in \mathbf{E}$ or $(j, i) \in \mathbf{E}$. Furthermore, let

$$\mathbf{E}_i^+ := \{(j, i) \in \mathbf{E} \mid j \in \mathbf{N}_i \text{ and } \dot{m}_{ji} > 0\} \quad \text{and} \quad \mathbf{E}_i^- := \{(i, j) \in \mathbf{E} \mid j \in \mathbf{N}_i \text{ and } \dot{m}_{ij} \geq 0\}$$

denote the set of edges through which water flows into respectively out of node i . Kirchhoff's law can be applied to the network in the sense that the total mass of water which flows into a node, matches the total mass of water flowing out of a node:

$$\sum_{(j,i) \in \mathbf{E}_i^+} \dot{m}_{ji} = \sum_{(i,j) \in \mathbf{E}_i^-} \dot{m}_{ij}, \quad \text{for all } i \in \mathbf{V}. \quad (2)$$

The temperature in each node is determined by the mixing laws of thermodynamics [23]:

$$T_i = \left(\sum_{(j,i) \in \mathbf{E}_i^+} \dot{m}_{ji} T_{ji}^{\text{end}} \right) / \left(\sum_{(j,i) \in \mathbf{E}_i^+} \dot{m}_{ji} \right), \quad \text{for all } i \in \mathbf{V}. \quad (3)$$

Similarly, the temperature at the origin of an edge is given by the temperature of the node if the edge is an outflow of that node:

$$T_{ij}^{\text{start}} = T_i \quad \text{if } (i, j) \in \mathbf{E}_i^-. \quad (4)$$

In our model, the four edge types \mathbf{E}^{load} , \mathbf{E}^{pipe} , $\mathbf{E}^{\text{heating}}$ and \mathbf{E}^{pump} are distinguished. A single edge might represent multiple components in the physical network:

- The edges \mathbf{E}^{load} represent consumers in the heating grid. In the physical world a heat exchanger is used to transfer energy from the district heating grid into the household heating system. A valve controls the mass flow through the heat exchanger in order to keep the water flowing back into the district heating network at a constant temperature. Additional equipment might be installed in order to measure the heat consumption or restrict the maximal mass flow [8]. This behavior is modeled as

$$T_{ij}^{\text{end}} = T_{ij}^{\text{set}}, \quad \text{for all } (i, j) \in \mathbf{E}^{\text{load}} \quad (5)$$

$$\dot{m}_{ij} = \frac{\dot{Q}_{ij}}{c_p (T_{ij}^{\text{start}} - T_{ij}^{\text{end}})}, \quad \text{for all } (i, j) \in \mathbf{E}^{\text{load}} \quad (6)$$

$$p_i - p_j \geq \Delta p_{\min}, \quad \text{for all } (i, j) \in \mathbf{E}^{\text{load}} \quad (7)$$

where $c_p = 4.182 \text{ kJ kg}^{-1} \text{ K}^{-1}$ is the specific heat capacity of water and T_{ij}^{set} and \dot{Q}_{ij} are the consumer's set-points for the return temperature and the transferred heat energy, respectively. The model parameters θ are exactly these heat energies \dot{Q}_{ij} on a demand edge $(i, j) \in \mathbf{E}^{\text{load}}$. A minimum pressure difference Δp_{\min} has to be applied by the grid in order to enable the flow through all components, cf. [8].

- Pipes \mathbf{E}^{pipe} are passive elements in the grid, meaning that the change in pressure and temperature are not actively controlled but resulting from the mean mass flow through the pipe and the soil temperature T_a :

$$T_{ij}^{\text{end}} = (T_{ij}^{\text{start}} - T_a) \exp\left(-\frac{l_{ij} \lambda_{ij}}{c_p \dot{m}_{ij}}\right) + T_a, \quad \text{for all } (i, j) \in \mathbf{E}^{\text{pipe}}, \quad (8)$$

$$p_i - p_j = \kappa_{ij} f_{D,ij} \frac{8l_{ij}}{\pi^2 \rho d_{ij}^5} \dot{m}_{ij}^2 + \rho g(z_j - z_i), \quad \text{for all } (i, j) \in \mathbf{E}^{\text{pipe}}, \quad (9)$$

compare [9, 23]. The coefficient λ_{ij} denotes the heat transferred through the isolation per pipe length and temperature difference between water and soil in $\text{W m}^{-1} \text{ K}^{-1}$. Furthermore, $f_{D,ij}$ is the Darcy friction factor which can be calculated by the Colebrook-White equation

$$\frac{1}{\sqrt{f_{D,ij}}} = -2 \log_{10} \left(\frac{\epsilon_{ij}}{3.7 d_{ij}} + \frac{2.51}{\text{Re}_{ij} \sqrt{f_{D,ij}}} \right) \quad (10)$$

depending on the inner roughness ϵ_{ij} and the diameter d_{ij} of the pipes, as well as the Reynolds number Re_{ij} . The correction factor κ_{ij} is introduced in eq. (9) to account for the pressure loss due to bends. The constants κ_{ij} , d_{ij} and ϵ_{ij} are grid parameters that are assumed to be well known. Evidently, $\rho = 997 \text{ kg m}^{-3}$ is the density of water and $z_j - z_i$ is the difference of altitude between the nodes j and i .

- The heating edges $\mathbf{E}^{\text{heating}}$ are introduced to serve as slack edges to fulfill the law of energy conservation in the grid. Therefore the equations

$$T_{ij}^{\text{end}} = T_{ij}^{\text{set}} \quad \text{for all } (i, j) \in \mathbf{E}^{\text{heating}} \quad (11)$$

$$p_j = p_i \quad \text{for all } (i, j) \in \mathbf{E}^{\text{heating}}. \quad (12)$$

restrict only the temperature at the end of an edge to be at a fixed set-point T_{ij}^{set} while the pressure does not change.

- Likewise, the pump edges E^{pump} are introduced as slack edges where the temperature

$$T_{ij}^{\text{end}} = T_{ij}^{\text{start}}, \quad \text{for all } (i, j) \in E_{ij}^{\text{pump}}, \quad (13)$$

does not change. Typically, the model has only one heating and one pump edge which are directly connected in series and represent the largest heating plant in the network. The pressure difference between the supply side and the return side is at its highest at the heating plant and decreases with increasing distance. The demand edge $(v, w) \in E^{\text{load}}$ with the lowest pressure difference is the so called worst point of the network. In real life networks this point is well known. The typical control scheme consists of measuring the pressure difference at this point and adjusting the pump pressure in such a way that the requirement in eq. (7) is just met for the worst point. Additionally an overall pressure level has to be maintained to prevent evaporation. These restrictions are mimicked in our model by fixing the pressures

$$p_v = p_0, \quad p_w = p_0 - \Delta p_{\min}, \quad (14)$$

with a suitable setpoint value p_0 measured in bar.

The functional relations (2)–(14) form a system of nonlinear equations which are summarized by the operator $e(\mathbf{y}, \boldsymbol{\theta}, \boldsymbol{\eta})$ in a state equation

$$e(\mathbf{y}, \boldsymbol{\theta}, \boldsymbol{\eta}) = 0, \quad (15)$$

which has a unique solution $\mathbf{y}(\boldsymbol{\theta}, \boldsymbol{\eta})$ for given consumer demands $\boldsymbol{\theta}$ and given set-point values $\boldsymbol{\eta} := (p_0, \Delta p_{\min}, p_{ij}^{\text{start}}, T_{ij}^{\text{set}})$ as introduced before. The proof of the uniqueness and existence of the solution can be done analogous to [9]. We solve eq. (15) by a Newton-method with projected gradients where the starting point is determined after a fixed point iteration according to [9, 23].

3 BAYESIAN INFERENCE AND OPTIMAL EXPERIMENTAL DESIGN

Model-based optimal design of experiments has the task to find a setup of experimental conditions, like sensor positions and control mechanisms, such that the model parameters can be estimated with minimal variance. In our setting we want to find optimal sensor positions in district heating networks such that the uncertainty in the estimated demand values $\boldsymbol{\theta}$ is minimized.

The model equation (15) brings the state \mathbf{y} , the demands $\boldsymbol{\theta}$ and the set-point values $\boldsymbol{\eta}$ into a functional relation. Our aim is to employ temperature-, pressure- and flow-sensors that measure the components of the solution $\mathbf{y}(\boldsymbol{\theta}, \boldsymbol{\eta})$. However, not all state variables in the network are of equal interest. We want to exclude the unreasonable sensor positions at the outset to reduce the dimension of the resulting optimization problem. Therefore, we select only a subset of the vector \mathbf{y} as possible output channels that can be measured by sensors. Let Ξ be such a selection matrix. Thus, we introduce the overall parameter-to-observable map

$$\boldsymbol{\theta} \mapsto \mathbf{h}(\boldsymbol{\theta}) := \Xi \cdot \mathbf{y}(\boldsymbol{\theta}, \boldsymbol{\eta}) \in \mathbb{R}^{n_s} \quad (16)$$

that maps the model parameters to n_s quantities of interest that can be directly measured by sensors. This mapping $\mathbf{h}(\boldsymbol{\theta})$ serves as our computer model that is commonly enhanced by a probabilistic point of view [22] where the collected data \mathbf{z} is assumed to be subject to observational noise which is modeled as a random variable $\boldsymbol{\varepsilon}$:

$$\mathbf{z} = \mathbf{h}(\boldsymbol{\theta}) + \boldsymbol{\varepsilon}. \quad (17)$$

Within a Bayesian framework, the posterior probability distribution of the estimated parameters is determined by the Bayes formula. Let $\gamma > 0$ and $\pi_0 \in \mathcal{N}(\boldsymbol{\theta}_0, \boldsymbol{\Gamma}_0/\gamma)$ be a Gaussian prior, capturing the a priori knowledge we have about $\boldsymbol{\theta}$. Furthermore, let $\boldsymbol{\varepsilon} \in \mathcal{N}(0, \boldsymbol{\Sigma})$ be a Gaussian random variable with density ρ and noise covariance matrix $\boldsymbol{\Sigma}$. We assume that the measurements obtained from different sensors are independently distributed and thus $\boldsymbol{\Sigma}$ is a diagonal matrix. Considering eq. (17), the data likelihood $\pi(\mathbf{z} | \boldsymbol{\theta})$ has thus the density $\rho(\mathbf{z} - \mathbf{h}(\boldsymbol{\theta}))$. Then the posterior $\pi(\boldsymbol{\theta} | \mathbf{z})$ is given by

$$\begin{aligned} \pi(\boldsymbol{\theta} | \mathbf{z}) &\propto \pi(\mathbf{z} | \boldsymbol{\theta})\pi_0 \\ &= \exp\left(-\frac{1}{2} \|\mathbf{z} - \mathbf{h}(\boldsymbol{\theta})\|_{\boldsymbol{\Sigma}^{-1}}^2 - \frac{\gamma}{2} \|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|_{\boldsymbol{\Gamma}_0^{-1}}^2\right), \end{aligned} \quad (18)$$

where $\|\mathbf{x}\|_A := \sqrt{\mathbf{x}^\top \mathbf{A} \mathbf{x}}$ is the weighted norm of a vector \mathbf{x} with a matrix \mathbf{A} . The maximum a posteriori estimator (MAP) is a point $\bar{\boldsymbol{\theta}}$ that maximizes this posterior probability distribution function:

$$\bar{\boldsymbol{\theta}}(\mathbf{z}) := \operatorname{argmin}_{\boldsymbol{\theta}} \left(\frac{1}{2} \|\mathbf{z} - \mathbf{h}(\boldsymbol{\theta})\|_{\boldsymbol{\Sigma}^{-1}}^2 + \frac{\gamma}{2} \|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|_{\boldsymbol{\Gamma}_0^{-1}}^2 \right), \quad (19)$$

compare [22].

In our case, the parameter-to-observable map is nonlinear and thus one cannot expect to obtain a posterior probability distribution that yields confidence regions which are analytically tractable. Therefore, we linearize the mapping $\boldsymbol{\theta} \mapsto \mathbf{h}(\boldsymbol{\theta})$ at the MAP point $\bar{\boldsymbol{\theta}}$ to obtain a Gaussian posterior whose covariance matrix is given by

$$\mathbf{C}_{\text{post}}(\bar{\boldsymbol{\theta}}) = \left(\mathbf{J}^\top \boldsymbol{\Sigma}^{-1} \mathbf{J} + \gamma \boldsymbol{\Gamma}_0^{-1} \right)^{-1}, \quad (20)$$

where \mathbf{J} is the sensitivity matrix which is computed by the implicit function theorem:

$$\mathbf{J} := \left. \frac{\partial \mathbf{h}}{\partial \boldsymbol{\theta}} \right|_{\boldsymbol{\theta}=\bar{\boldsymbol{\theta}}} = -\boldsymbol{\Xi} \left[\frac{\partial \mathbf{e}(\mathbf{y}(\bar{\boldsymbol{\theta}}, \boldsymbol{\eta}), \bar{\boldsymbol{\theta}}, \boldsymbol{\eta})}{\partial \mathbf{y}} \right]^{-1} \frac{\partial \mathbf{e}(\mathbf{y}(\bar{\boldsymbol{\theta}}, \boldsymbol{\eta}), \bar{\boldsymbol{\theta}}, \boldsymbol{\eta})}{\partial \boldsymbol{\theta}} \quad (21)$$

The confidence region $G(\bar{\boldsymbol{\theta}}, \mathbf{C}_{\text{post}}, \alpha)$ around the MAP point $\bar{\boldsymbol{\theta}}$ to a level $1 - \alpha$, where $\alpha \in (0, 1)$, has then the analytical expression

$$G(\bar{\boldsymbol{\theta}}, \mathbf{C}_{\text{post}}, \alpha) := \left\{ \boldsymbol{\theta} \in \mathbb{R}^{n_p} : (\boldsymbol{\theta} - \bar{\boldsymbol{\theta}})^\top \mathbf{C}_{\text{post}}^{-1} (\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}) \leq \chi_{n_p}^2(1 - \alpha) \right\}, \quad (22)$$

where $\chi_{n_p}^2(1 - \alpha)$ is the quantile of the χ^2 distribution with n_p degrees of freedom, see [10].

We now introduce weights $\omega_k \in \{0, 1\}$ for each predefined sensor position $k = 1, \dots, n_s$ such that $\omega_k = 1$ if, and only if, sensor k is used. Set $\boldsymbol{\Omega} := \operatorname{diag}(\omega_1, \dots, \omega_{n_s})$ as the weight matrix containing the vector $\boldsymbol{\omega}$ on its diagonal. The knowledge received from the used sensors is added to the noise model $\boldsymbol{\varepsilon} \in \mathcal{N}(0, \boldsymbol{\Omega}^{-1} \boldsymbol{\Sigma})$, whereby a division by zero is set to infinity. This has the meaningful interpretation that an unused sensor yields an infinitely large covariance in the corresponding output channel, i.e., we know nothing about that quantity of interest.

These weights $\boldsymbol{\omega}$ directly influence the MAP point

$$\bar{\boldsymbol{\theta}}(\mathbf{z}; \boldsymbol{\omega}) := \operatorname{argmin}_{\boldsymbol{\theta}} \left(\frac{1}{2} \|\mathbf{z} - \mathbf{h}(\boldsymbol{\theta})\|_{\boldsymbol{\Omega} \boldsymbol{\Sigma}^{-1}}^2 + \frac{\gamma}{2} \|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|_{\boldsymbol{\Gamma}_0^{-1}}^2 \right), \quad (23)$$

the sensitivity \mathbf{J} in eq. (21), the posterior covariance matrix of the parameters

$$\mathbf{C}_{\text{post}}(\bar{\boldsymbol{\theta}}(\mathbf{z}; \boldsymbol{\omega}), \boldsymbol{\omega}) = \left(\mathbf{J}^\top \boldsymbol{\Omega} \boldsymbol{\Sigma}^{-1} \mathbf{J} + \gamma \boldsymbol{\Gamma}_0^{-1} \right)^{-1}, \quad (24)$$

as well as the confidence region in eq. (22), compare [2, 14].

In an optimally designed experiment, a fair compromise between the cost $\mathbf{c} \in \mathbb{R}_+^{n_s}$ of the used sensors and a measure Ψ of \mathbf{C}_{post} , representing the information gain by these sensors, is obtained. In order to reduce the computational complexity of the following problem and since \mathbf{z} is difficult to obtain or not available beforehand, we set $\bar{\boldsymbol{\theta}}(\mathbf{z}; \boldsymbol{\omega}) = \bar{\boldsymbol{\theta}}_0$ to an initial MAP estimation and keep it constant through the optimization. Let $\kappa > 0$ be a penalty factor and let $\|\cdot\|_0$ be the ℓ_0 -"norm". We consider the optimal experimental design problem

$$\min_{\boldsymbol{\omega} \in \{0,1\}^{n_s}} \Psi[\mathbf{C}_{\text{post}}(\bar{\boldsymbol{\theta}}_0, \boldsymbol{\omega})] + \kappa \sum_{k=1}^{n_s} c_k \|\omega_k\|_0 \quad (25)$$

If n_s is large, problem (25) is very difficult to solve due to combinatorial explosion. Therefore, we perform a relaxation on the domain of definition of $\boldsymbol{\omega}$ and replace the discontinuous penalty by a smooth function P_δ , where $\delta \in (0, 1]$, which converges to the ℓ_0 -"norm" for $\delta \rightarrow 0$. For $\delta = 1$ this function has the form $P_{\delta=1}(\boldsymbol{\omega}, \mathbf{c}) := \mathbf{c}^\top \boldsymbol{\omega}$, otherwise

$$P_\delta(\boldsymbol{\omega}, \mathbf{c}) := \sum_{k=1}^{n_s} c_k f_\delta(\omega_k), \quad \text{for } \delta \in (0, \frac{1}{2}), \quad (26)$$

where $f_\delta(x) : [0, 1] \mapsto [0, 1]$ is continuously differentiable and approximates the ℓ_0 -"norm", see [1] for more details. The relaxed optimization problem

$$\min_{\boldsymbol{\omega} \in [0,1]^{n_s}} \Psi[\mathbf{C}_{\text{post}}(\bar{\boldsymbol{\theta}}_0, \boldsymbol{\omega})] + \kappa P_\delta(\boldsymbol{\omega}, \mathbf{c}) \quad (27)$$

is first solved for $\delta = 1$ and then by a reiteration scheme for diminishing δ the optimal sensor weights $\boldsymbol{\omega}_{\text{opt}}$ tend to become sparse and $\{0, 1\}$ -valued for a suitable choice of $\kappa > 0$, cf. [1, 2]. We solve problem (27) by standard BFGS-SQP methods [20].

According to [12], the most prominent design criteria Ψ measuring the size of a matrix \mathbf{C} are the following:

$$\Psi_A = \text{trace}(\mathbf{C}), \quad \Psi_D = \det(\mathbf{C}), \quad \Psi_E = \lambda_{\max}(\mathbf{C}). \quad (28)$$

It is known that problem (27) with $\delta = 1$ is convex for $\Psi = \Psi_A$ and $\Psi = \Psi_D$, see [17]. However, using Ψ_E requires non-smooth methods [13]. In this paper, we choose to compute the trace of the posterior covariance matrix.

4 NUMERICAL RESULTS FOR A REAL-SIZED HEATING NETWORK

We demonstrate our method for the heating grid in the district Darmstadt-Nord of the German city Darmstadt. The network is operated by the ENTEGA AG, the heat is provided by a central heating plant and distributed to 67 consumers through a pipe network of approximately 14.2 km length. The graph representation of the grid consists of 372 edges and 306 nodes.

Over the course of a day significant changes of overall demands as well as the relative distribution between the consumers is observed in practice. In the following we analyze the experiment design for each hour of the day individually and compare the results. Final investment decisions would have to be taken such that they yield good performance in all hours of the day. The hourly index is omitted in this section.

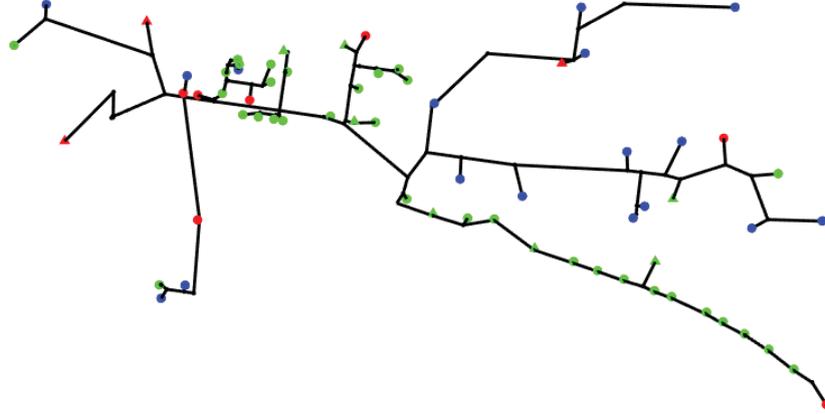


Figure 1: Network structure of the considered heating grid. Consumers with hourly measured demands are marked as circles, unmeasured as triangles. Green markers represent residential buildings, red ones commercial buildings and blue represent buildings in the group others.

4.1 Prior demand estimation

The prior load distribution $\mathcal{N}(\boldsymbol{\theta}_0, \boldsymbol{\Gamma}_0/\gamma)$ is estimated based on available consumption data. For 56 consumers the load is directly measured with hourly resolution. Let \mathbf{K}_M be the index set of all measured consumers and let \mathbf{K}_U be the index set of all unmeasured. The mean parameter $\theta_{0,i}$ for the measured demand \dot{Q}_i^d of consumer i is given by the mean consumption over all observed days d

$$\theta_{0,i} = \frac{1}{D} \sum_{d=1}^D \dot{Q}_i^d, \quad \text{for all } i \in \mathbf{K}_M. \quad (29)$$

Unmeasured consumers $j \in \mathbf{K}_U$ are paired with similar measured consumers $j^* \in \mathbf{K}_M$. The expected demand is given by scaling the expectation value of the paired consumer by the associated total consumption that was also available for us, \dot{Q}_j^{2019} , respectively, $\dot{Q}_{j^*}^{2019}$:

$$\theta_{0,j} = \frac{\dot{Q}_j^{2019}}{\dot{Q}_{j^*}^{2019}} \theta_{0,j^*}, \quad \text{for all } j \in \mathbf{K}_U. \quad (30)$$

The return temperatures T^{set} for each consumer are estimated by the same procedure. Different heat consumption profiles can be explained by the varying outdoor temperatures and by different consumer behavior [6]. Grouping consumers for whom similar behavior is expected leads to three groups. The first group \mathbf{G}^{res} , being residential buildings, consist of 40 multi-family houses and one single-family house. The second group \mathbf{G}^{TC} is made up by 10 commercial buildings. The remaining 17 buildings form the group $\mathbf{G}^{\text{other}}$ and are not expected to show strong similarities. Fig. 1 shows the distribution of demand classes in the network.

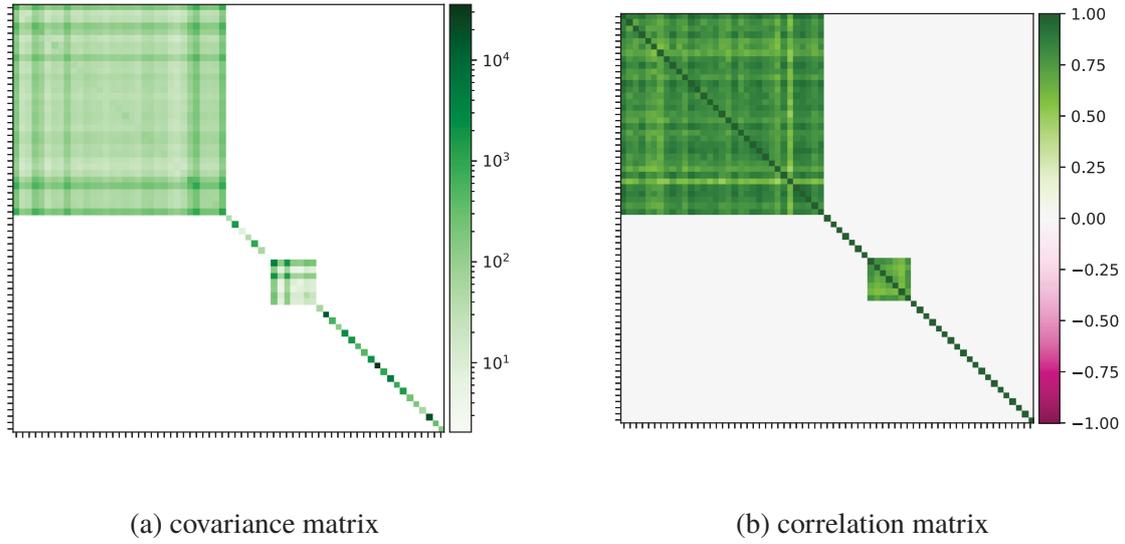


Figure 2: In 2a the covariance matrix Γ_0 for the heat demand \dot{Q} hour 3 to 4 pm is shown. The first block represents the group of measured residential buildings, followed by the unmeasured residential buildings for which all correlation coefficients are set to zero. The second block represents measured and unmeasured commercial buildings, the third the buildings of the "other" group. The variances can differ by several orders of magnitude due to the largely different size of the buildings. 2b shows the corresponding correlation matrix normalized by the standard deviations.

The standard deviation σ_i for the measured demands i are estimated by

$$\sigma_i = \sqrt{\frac{1}{D} \sum_{d=1}^D (\dot{Q}_i^d - \theta_{0,i})^2}, \quad \text{for all } i \in \mathbf{K}_M. \quad (31)$$

Within the first two groups the normalized standard deviations $\sigma_i/\theta_{0,i}$ are close-by for all hourly measured consumers. The standard deviation of the unmeasured demands

$$\sigma_j = \frac{\theta_{0,j}}{|\mathbf{K}_M|} \sum_{i \in \mathbf{K}_M} \frac{\sigma_i}{\theta_{0,i}}, \quad \text{for all } j \in \mathbf{K}_U, \quad (32)$$

are therefore estimated by scaling the mean normalized variations by the estimated demand expectations. For the group $\mathbf{G}^{\text{other}}$ all standard deviations can be gathered directly from the data. The entries $\Gamma_{0,ij}$ for the prior covariance matrix Γ_0 can now be gathered by

$$\Gamma_{0,ij} = \begin{cases} \sigma_i^2, & \text{if } i = j, \\ \frac{1}{D-1} \sum_{d=1}^D (\dot{Q}_i^d - \theta_{0,i})(\dot{Q}_j^d - \theta_{0,j}), & \text{if } i, j \in \mathbf{G}^{\text{res}} \cap \mathbf{K}_M \text{ or } i, j \in \mathbf{G}^{\text{TC}} \cap \mathbf{K}_M, \\ 0, & \text{else.} \end{cases} \quad (33)$$

The covariance coefficients between two groups as well as for consumers in group $\mathbf{G}^{\text{other}}$ are set to 0, because no similarity in the demand patterns are expected here. When ordered according to the groups, the block structure of the covariance matrix becomes clearly visible as it is shown in Fig. 2a for the hour 3 to 4 pm.

The factor $\gamma > 0$ must be large enough to achieve regularity of the covariance matrix, see eq. (24), and to make the solver of the optimization problem (27) numerically stable. This factor also brings the prior into proper scaling when added to the data misfit part of the covariance formula in eq. (24). In our case, taking $\gamma = 5 \times 10^3$ was sufficient.

4.2 Sensors

Two different kinds of sensors are considered, namely pressure sensors and power flow sensors measuring three values simultaneously. There are two parallel pipes in the grid, one delivering hot water and one returning the cold water to the heating plant. Let $(i, j) \in \mathbf{E}^{\text{pipe}}$ be a pipe on the return side (cold water) and $(k, l) \in \mathbf{E}^{\text{pipe}}$ be the corresponding parallel pipe on supply side (hot water). A potential power flow sensor would then measure the mass flow \dot{m}_{ij} , which is equal for both pipes, as well as the temperatures T_{ij}^{end} and T_{kl}^{start} . The power transmitted along the pipe-pair is given by the temperature difference between the two pipes and the mass flow through them:

$$P_{ij}^{kl} = c_p \dot{m}_{ij} (T_{ij}^{\text{end}} - T_{kl}^{\text{start}}). \quad (34)$$

There are 150 plausible positions for such heating power sensors and as many pressure sensor locations as nodes in the network are available, altogether we obtain $n_s = 456$ candidate sensor positions.

Each sensor has a different accuracy when measuring the quantities of interest. The pressure sensors operate with a 0.04 % precision of the measured pressure value: $\Delta p_i = 0.04 \% \cdot p_i$, for all $i \in \mathbf{V}$. The mass flow \dot{m}_{ij} is determined by measuring the flow speed with a fixed accuracy. The accuracy of the mass flow therefore depends on the pipe diameter d_{ij} and the density of water ρ :

$$\Delta \dot{m}_{ij} = \rho \frac{d_{ij}^2}{4} \cdot 0.012 \text{ m s}^{-1}, \quad \text{for all } (i, j) \in \mathbf{E}. \quad (35)$$

For the temperature measurement, we have $\Delta T_i = \Delta T_{kl}^{\text{end}} = \Delta T_{kl}^{\text{start}} = 0.6 \text{ }^\circ\text{C}$, for all $i \in \mathbf{V}$ and $(k, l) \in \mathbf{E}$. These values form the diagonal entries of the covariance matrix Σ of the noise model ε , see eq. (17). The non-diagonal entries in Σ are set to zero since we assume the sensor readings to be statistically independent.

4.3 Computational results

We solved the state equation (15) by a Newton-method with projected gradients after the starting point had been computed by a truncated fixed-point iteration. The usage of projected gradients enhanced the convergence properties, since the network was designed to yield only positive mass flows $\dot{\mathbf{m}}$. After computing the sensitivity matrix \mathbf{J} , see eq. (21), we solved problem (27) with a standard SQP-method where the Hessian is constructed by BFGS-updates. The reiteration scheme with the penalty term was performed starting with $\delta_1 = 1$ and then updating $\delta_{k+1} := \delta_k/2$ for $k = 1, \dots, 6$. Thus, in a few reiterations, the optimal sensor weights became sparse and almost $\{0, 1\}$ -valued. We additionally want to point out that the solution was found after approximately 120 function and 110 gradient evaluations in total for each hour, respectively.

We pick the hour of the day with the highest heat demand, which was 9 to 10 am, and compare the results for different values of κ in Tab. 1. We also generated 500 random vectors

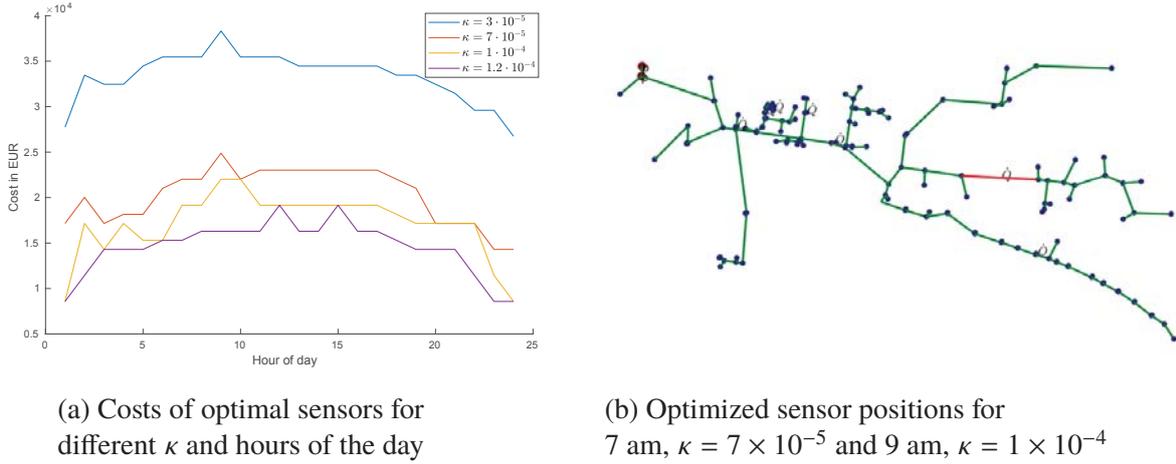


Figure 3: 3a The optimized set of sensors depends on the chosen value for κ as well as the hour of the day due to different prior assumptions for the demands. Sets resulting in equal costs are usually very similar or equal as in 3b. This indicates that the proposed approach leads to sensor positions that perform well for all hours.

ω_{rnd} and computed the average of their respective Ψ_A, Ψ_D and Ψ_E criteria, to compare with an optimal selection of ω and with no sensors at all. We observe that the smaller the value of κ the more sensors are used and the smaller is the design criterion. However, the greater the number of installed sensors the higher the costs. The improvement of prior information about the demands which is achieved by using sensors at the optimized positions ω_{opt} , is between 56 % and 64 % when evaluating Ψ_A , clearly over 99 % when using Ψ_D and between 71 % and 77 % when evaluating Ψ_E . A random selection of an equal number of sensors produces a much smaller difference while the computational efforts are much higher.

As seen in Fig. 3a, the optimal trade-off between the cost of sensors and the design criterion varies for the different hours of the day for a constant κ resulting in a different number of sensors and therefore different costs at the optimal solution. This behavior can be attributed to the different prior assumptions over the demands for each hour of the day. Fig. 4 shows the design criteria Ψ_A over the cost for the sensors for optimized sensor networks for each hour individually. By changing the value for κ the optimum can be altered along these estimated optimality curves. It can be seen from this representation that the curve steeply decreases for low numbers of sensors and flattens for higher numbers. This is similar to a pareto-front known from multi-objective optimization. Additionally, it seems as if the optimality curves tend to

Table 1: Comparison of the solution of problem (27) for different κ from hour 9 to 10 am. We additionally computed the design criteria Ψ_D and Ψ_E .

#	$\ \omega\ _0$	$c^\top \omega$	κ	Ψ_A	Ψ_D	Ψ_E
\emptyset	0	-	-	23.46	1.65×10^{-127}	7.91
ω_{rnd}	7	-	-	22.60	1.20×10^{-127}	7.62
ω_{opt}	7	1.630×10^4	1.2×10^{-4}	10.27	3.48×10^{-131}	2.28
ω_{rnd}	9	-	-	22.58	1.14×10^{-127}	7.65
ω_{opt}	9	2.202×10^4	0.7×10^{-4}	9.28	3.45×10^{-132}	2.00
ω_{rnd}	15	-	-	21.97	8.98×10^{-128}	7.47
ω_{opt}	15	3.546×10^4	0.3×10^{-4}	8.55	5.55×10^{-133}	1.80

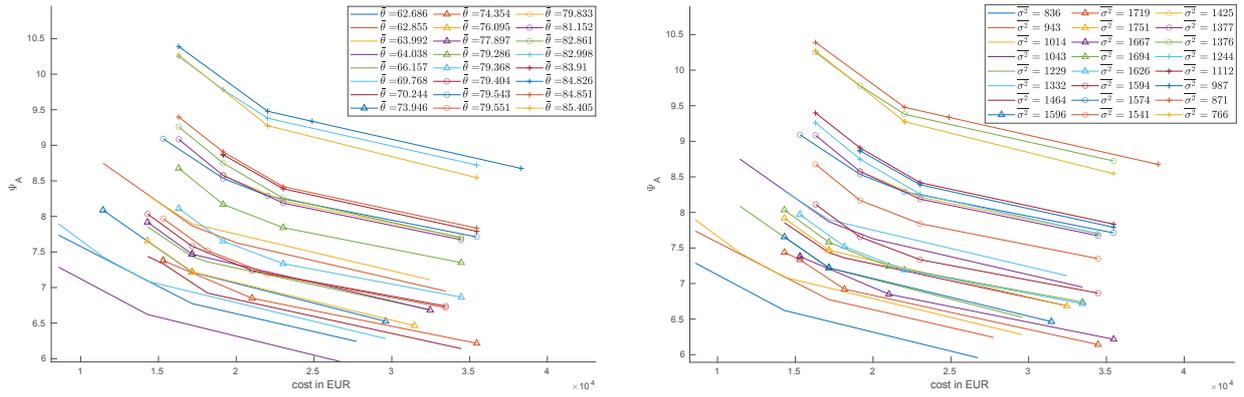


Figure 4: By choosing κ , a trade-off can be made between a better information score Ψ_A and lower costs. The lines along the weighting moves tend to shift towards higher prices and worse scores for increasing θ_0 and σ^2 . The values given in these figures are the mean values for all consumers. It can be seen that there are also some exceptions to this trend.

shift towards higher costs as well as higher Ψ_A -values with overall higher demands and higher variances for the demands. Even though these trends do not hold true for all demand priors examined, it should be taken into account when choosing a κ for practical applications. Fig. 3b shows the position of sensors for the hour 7 am and $\kappa = 7 \times 10^{-5}$ which is exactly the same as the set for the hour 9 am choosing $\kappa = 1 \times 10^{-4}$. Similarly as in this example the chosen sensor positions for different hours tend to match each other closely if κ is set in such a way that similar cost arise in the optimum. It can therefore be expected that a set of sensors optimized for one hour provides good results for the other hours as well.

The mapping $\mathbf{y} = \mathbf{h}(\boldsymbol{\theta})$ can be used to estimate the network's state \mathbf{y} . Since we assume normal distributed uncertainty for the consumers demand, we can use the linearised model to estimate an normal distribution for the state $\mathbf{y}_{\text{est}} \in \mathcal{N}(\mathbf{y}_0, \boldsymbol{\Sigma}_y)$ with the covariance matrix $\boldsymbol{\Sigma}_y$ given by

$$\boldsymbol{\Sigma}_y \propto \mathbf{J} \mathbf{C}_{\text{post}} \mathbf{J}^\top, \quad (36)$$

where \mathbf{J} is the sensitivity matrix from (21). In Fig. 5 the estimated states for the prior and the posterior estimation over the demands for hour 9 am and $\kappa = 1 \times 10^{-4}$ are compared, by calculating the change in variance for each individual state variable. The graphic shows the temperatures variance as node color and the mass flow variance as edge color for the supply side of the network. It can be seen directly, that the uncertainty does not only decrease near the measurements, but for almost all state variables. When separating the grid into different sections it seems as if within each section the gains are relatively even, but differ strongly between the sections. This can be motivated by the different consumer structure. By comparing with Fig. 1 it can be observed that the lowest section on the right side mostly consists of residential buildings. For these the variances were already rather low for the prior assumption as seen in Fig. 2a. In the middle section on the right side, many consumers are classified as "others", showing comparable large demands and large variations in the prior. Therefore the relative gain is larger for this area.

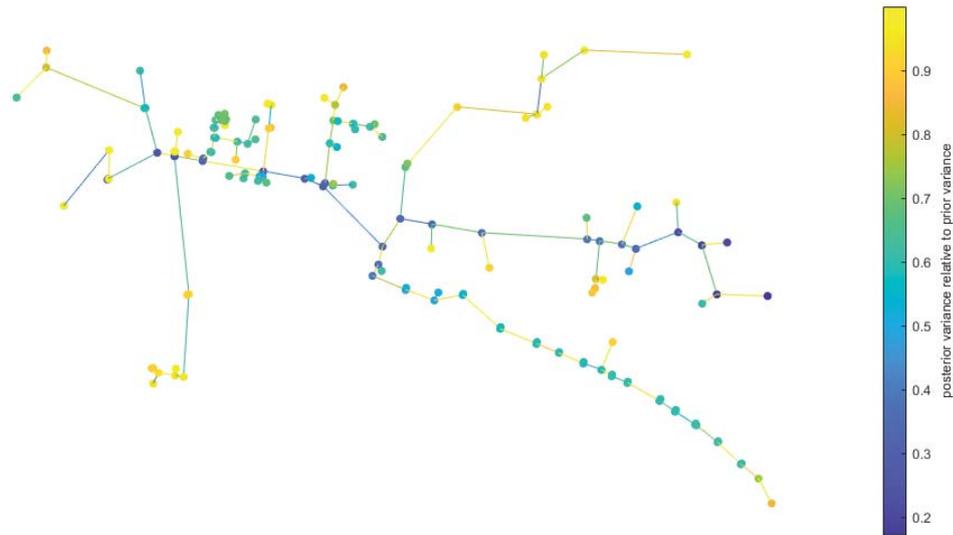


Figure 5: The linearized model can be used as a state estimator. Using the posterior estimation for the consumers demand significantly reduces the uncertainty compared to the prior assumption. This plot shows the uncertainty change for the temperatures (node color) and the mass flow (edge color) for the supply side of the network.

5 CONCLUSION

In this paper we presented a Bayesian approach for a sensor placement problem in heating networks to improve the reliability of our knowledge about one of the most important system parameters – the consumer’s heat consumption. The sensor placement task can be interpreted as an optimal experimental design problem which we modeled using the solution of the state equation of the heating network, its sensitivity matrix and the Bayes formula for the parameter’s covariance. We solved this optimization problem by a BFGS-SQP method with a reiteration scheme to obtain sparse and almost $\{0, 1\}$ -valued sensor weights. The optimally positioned sensors measure temperatures, pressures and mass flows in the network at a trade-off between small covariance values and low costs. In subsequent numerical experiments we applied our method to the heating network of the northern district of the German city Darmstadt. We showed that the optimal placement of a few sensors significantly increased the informational value of the uncertain demands at low cost when compared to randomly placed sensors. This shows that our method is superior to Monte-Carlo approaches. An enhanced knowledge of the demand values provides the basis for a detailed monitoring of the network state in order to reduce industrial waste-heat.

Even though the numerical results are very promising, some barriers remain for practical use of this approach. In order to optimize the sensor positions, a prior distribution for the demands is required. For our investigated network, this was done by analyzing each consumer’s past demand which was measured with hourly resolution for most buildings. However, measurements of this kind are rather uncommon in district heating networks. The expected demand as well as the variance of the demands who were not measured, were estimated by comparing the buildings with measured ones. In order to treat these consumers equally to the measured ones, the non-

diagonal entries of the covariance matrix should be estimated as well. However, this task is not trivial since estimating the coefficients, e.g., by the mean of the coefficients in the same buildings group, will most likely lead to a matrix which is not positive definite and therefore no valid covariance matrix.

The optimal sensor positions found in our setting may differ from the sensors that would be chosen in real life application. For example, sensors are still placed in the connection lines between single buildings and the main grid, which is equivalent to directly measuring the demand at the consumers heat exchange station. These measurement positions would be unsuitable for real life applications due to fast demand changes that are likely to occur. However, this possibility is not considered by the steady state model. Another example is given by the two pressure sensors in Fig. 3b, which are next to each other. In our model, this is equivalent to measuring the mass flow through this pipe as given by (9). For practical applications this would not be a preferable setup, as the pressure differences between the two nodes are too small to gain usable information.

In order to install sensors in real district heating grids, many additional factors need to be taken into account, e.g., how well a potential sensor position could be reached, for installation or maintenance. The optimization scheme proposed in this paper can contribute to this decision process by suggesting optimal sensor positions under simplified conditions or by comparing different possible settings in the context of the chosen model.

ACKNOWLEDGMENTS

We would like to thank the German Research Foundation (DFG) – project number 57157498 – CRC 805 and the German Federal Ministry for economic affairs and energy (BMWi) – project number 03EN3012A – for funding this research. We kindly acknowledge the help of ENTEGA AG who provided the necessary data and helped interpreting the results.

REFERENCES

- [1] Alen Alexanderian, Noemi Petra, Georg Stadler, and Omar Ghattas. A-optimal design of experiments for infinite-dimensional Bayesian linear inverse problems with regularized ℓ_0 -sparsification. *SIAM Journal on Scientific Computing*, 36(5):A2122–A2148, 2014.
- [2] Alen Alexanderian, Noemi Petra, Georg Stadler, and Omar Ghattas. A fast and scalable method for A-optimal design of experiments for infinite-dimensional Bayesian nonlinear inverse problems. *SIAM Journal on Scientific Computing*, 38(1):A243–A272, 2016.
- [3] Irene Bauer, Hans G. Bock, Stefan Körkel, and Johannes P. Schlöder. Numerical methods for optimum experimental design in dae systems. *Journal of Computational and Applied Mathematics*, 120(1-2):1–25, 2000.
- [4] Ilyes Ben Hassine and Ursula Eicker. Impact of load structure variation and solar thermal energy integration on an existing district heating network. *Applied Thermal Engineering*, 50(2):1437–1446, 2013.
- [5] Hanmin Cai, Shi You, and Jianzhong Wu. Agent-based distributed demand response in district heating systems. *Applied Energy*, 262:114403, 2020.
- [6] Bronislav Chramcov. Heat demand forecasting for concrete district heating system. *International Journal of Mathematical Models and Methods in Applied Sciences*, 2010.

- [7] Jean Duquette, Andrew Rowe, and Peter Wild. Thermal performance of a steady state physical pipe model for simulating district heating grids with variable flow. *Applied Energy*, 178:383–393, 2016.
- [8] ENTEGA AG, Darmstadt, Germany. *Technische Anschlussbedingungen (TAB) Fernwärme Darmstadt*, 2017.
- [9] Tingting Fang and Risto Lahdelma. State estimation of district heating network based on customer measurements. *Applied Thermal Engineering*, 73(1):1211–1221, 2014.
- [10] Valerii V. Fedorov and Sergei L. Leonov. *Optimal Design for Nonlinear Response Models*. CRC Press, 1st edition, 2013.
- [11] Eric B. Flynn and Michael D. Todd. A Bayesian approach to optimal sensor placement for structural health monitoring with application to active sensing. *Mechanical Systems and Signal Processing*, 24(4):891–903, 2010.
- [12] Gaia Franceschini and Sandro Macchietto. Model-based design of experiments for parameter precision: State of the art. *Chemical Engineering Science*, 63(19):4846–4872, 2008.
- [13] Jean-Baptiste Hiriart-Urruty and Adrian S. Lewis. The Clarke and Michel-Penot subdifferentials of the eigenvalues of a symmetric matrix. *Computational Optimization and Applications*, 13(1):13–23, 1999.
- [14] Stefan Körkel, Ekaterina Kostina, Hans G. Bock, and Johannes P. Schlöder. Numerical methods for optimal control problems in design of robust optimal experiments for nonlinear dynamic processes. *Optimization Methods and Software*, 19(3-4):327–338, 2004.
- [15] Xuezhi Liu, Jianzhong Wu, Nick Jenkins, and Audrius Bagdanavicius. Combined analysis of electricity and heat networks. *Applied Energy*, 162:1238–1250, 2016.
- [16] Henrik Lund, Sven Werner, Robin Wiltshire, Svend Svendsen, Jan Eric Thorsen, Frede Hvelplund, and Brian Vad Mathiesen. 4th generation district heating (4gdh): Integrating smart thermal grids into future sustainable energy systems. *Energy*, 68:1–11, 2014.
- [17] Ira Neitzel, Konstantin Pieper, Boris Vexler, and Daniel Walter. A sparse control approach to optimal sensor placement in PDE-constrained parameter estimation problems. *Numerische Mathematik*, 143(4):943–984, 2019.
- [18] Tinsley Oden, Robert Moser, and Omar Ghattas. Computer predictions with quantified uncertainty, Part I. *SIAM News*, 43(9):1–3, 2010.
- [19] Michael Papadopoulos and Ephraim Garcia. Sensor placement methodologies for dynamic testing. *AIAA journal*, 36(2):256–263, 1998.
- [20] Michael J. D. Powell. A fast algorithm for nonlinearly constrained optimization calculations. *Lecture Notes in Mathematics*, 630:144–157, 1978.
- [21] Friedrich Pukelsheim. *Optimal Design of Experiments*. Society for Industrial and Applied Mathematics, 2006.

- [22] Andrew M. Stuart. Inverse problems: A Bayesian perspective. *Acta Numerica*, 19:451–559, 2010.
- [23] Guoqiang Sun, Wenxue Wang, Yi Wu, Wei Hu, Zijun Yang, Zhinong Wei, Haixiang Zang, and Sheng Chen. A nonlinear analytical algorithm for predicting the probabilistic mass flow of a radial district heating network. *Energies*, 12(7):1215, 2019.
- [24] Shoujun Zhou, Zheng O’Neill, and Charles O’Neill. A review of leakage detection methods for district heating networks. *Applied Thermal Engineering*, 137:567–574, 2018.

IMPROVING THE RATE OF CONVERGENCE OF THE QUASI-MONTE CARLO METHOD IN ESTIMATING EXPECTATIONS ON A GEOTECHNICAL SLOPE STABILITY PROBLEM

Philippe Blondeel¹, Pieterjan Robbe¹, Dirk Nuyens¹, Geert Lombaert²
and Stefan Vandewalle¹

¹ KU Leuven, Department of Computer Science
Celestijnenlaan 200A, 3001 Leuven, Belgium
{philippe.blondeel, pieterjan.robbe, dirk.nuyens, stefan.vandewalle}@kuleuven.be
² KU Leuven, Department of Civil Engineering
Kasteelpark Arenberg 40, 3001 Leuven, Belgium
geert.lombaert@kuleuven.be

Keywords: Slope Stability, Geotechnical Engineering, Uncertainty Quantification, quasi-Monte Carlo Methods

Abstract. *The propagation of parameter uncertainty through engineering models is a key task in uncertainty quantification. In many cases, taking into account this uncertainty involves the estimation of expected values by means of the Monte Carlo method. While the performance of the classical Monte Carlo method is independent of the number of uncertainties, its main drawback is the slow convergence rate of the root mean square error, i.e., $O(N^{-1/2})$ where N is the number of model evaluations. Under appropriate conditions, the quasi-Monte Carlo method improves the order of convergence to $O(N^{-1})$ by using deterministic sample points instead of random sample points. Two examples of such point sets are rank-1 lattice sequences and Sobol' sequences. However, it is possible to further improve the order of convergence by applying the so-called "tent transformation" to a rank-1 lattice sequence, and by "interlacing" a Sobol' sequence. In this work, we benchmark these two techniques on a slope stability problem from geotechnical engineering, where the uncertainty is located in the cohesion of the soil. The soil cohesion is modeled as a lognormal random field of which realizations are computed by means of the Karhunen–Loève (KL) expansion. The quasi-Monte Carlo points are mapped to the normal distribution required in the KL expansion using a novel truncation strategy. We observe an order of convergence of $O(N^{-1.5})$ in our numerical experiments.*

1 INTRODUCTION

In practical engineering problems, uncertainty plays an essential role. This uncertainty can, for example, be present in the material parameters such as the cohesion of the soil in a slope stability problem. In this type of problem, the goal is to assess the stability of natural or man-made slopes. Classically, this assessment is carried out in a deterministic way, i.e., no uncertainty is taken into account. However, this approach offers only limited insight. In order to gain a better insight into the stability of the slope, the uncertainty of the soil needs to be propagated through its mathematical model, which consists of a discretized partial differential equation (PDE). A popular and straightforward method to account for this uncertainty is by using a “sampling method”. The most well known method belonging to this family is the Monte Carlo method, see [1]. In this method, the expected value of a user-chosen quantity of interest is computed as an average of multiple simulation outputs, each resulting from a different “sample” of the uncertainty. While the performance of the Monte Carlo method is independent of the stochastic dimension, i.e., the number of random variables used to represent the uncertain parameters, the computational cost measured in terms of the number of model evaluations is often still too large. This high computational cost stems on the one hand from the fact that all the samples are computed on one, possibly fine, discretization level (e.g., in order to approximate a PDE), and on the other hand from the slow convergence rate of the root mean square error, i.e., $O(N^{-1/2})$ where N is the number of samples. As engineering problems grow more complex and thus more costly, improvements which lower the computational cost of the Monte Carlo method have been proposed. One such improvement consists of converting the (single-level) Monte Carlo method into a Multilevel Monte Carlo method, see, e.g., [2]. In the Multilevel Monte Carlo method, the engineering problem is discretized multiple times with different mesh resolutions. The meshes resulting from the discretization are then grouped in a mesh hierarchy. The Multilevel Monte Carlo method achieves a speedup by taking many samples on computationally cheap low resolution meshes, and few samples on computationally expensive high resolution meshes. Another possible improvement consists of replacing the Monte Carlo sampling method by a quasi-Monte Carlo sampling method, see [3]. Instead of the random points used in the Monte Carlo method, the quasi-Monte Carlo method computes its samples at well chosen deterministic points. By using this approach, the order of convergence can be improved to $O(N^{-1})$, see [3, 4]. Most of these methods employ only a first-order Finite Element discretization of the underlying PDE. In previous work, see [5], we obtained an order of convergence close to $O(N^{-1})$ when combining the Multilevel Monte Carlo method with a quasi-Monte Carlo sampling method. In [6, 7], we combined the p -refinement of the Finite Element method (FEM) discretization with the Multilevel quasi-Monte Carlo sampling method, applied to a slope stability problem. There, the multilevel mesh hierarchy is constructed following a p -refinement approach, i.e., the order of the elements in the subsequent meshes is increased.

However, it is known that the $O(N^{-1})$ order of convergence can be improved for sufficiently smooth problems by using certain techniques, such as the *tent transformation* applied to a rank-1 lattice sequence [8] or by using an *interlacing* technique applied to a Sobol’ sequence, see, e.g., [3, 9]. In this work, we investigate if a higher-order quasi-Monte Carlo convergence can be obtained in a single-level setting, by applying the *tent transformation* and the *interlacing* technique. The investigation is carried out by applying the above mentioned techniques on a slope stability problem, where realizations of the random field, that is used to model the uncertainty, are computed using a truncated KL expansion.

The slope stability problem itself is discretized by means of triangular quadratic Finite Elements.

The paper is structured as follows. First, we present the model problem, and briefly discuss the underlying Finite Element solver. Second, we review the theoretical background of the quasi-Monte Carlo method as well as the tent transformation and interlacing techniques. Last, we present numerical results obtained by tent-transformed rank-1 lattice sequences and interlaced Sobol' sequences. Both results will be compared to the ones obtained by means of the standard rank-1 lattice and Sobol' sequence.

2 MODEL PROBLEM AND MESH DISCRETIZATION

The model problem we consider consists of a slope stability problem where the cohesion of the soil has a spatially varying uncertainty, see [10]. In a slope stability problem, the safety of the slope can be assessed by evaluating the vertical displacement of a point near the top of the slope, when sustaining its own weight. We consider the displacement in the plastic domain, which is governed by the Drucker–Prager yield criterion. A small amount of isotropic linear hardening is taken into account for numerical stability reasons. Because of the nonlinear stress-strain relation arising in the plastic domain, a Newton–Raphson iterative solver is used. In order to compute the displacement, an incremental load approach is applied, i.e., the total load resulting from the slope's weight is added in discrete load steps, starting with a force of 0N. These loads steps are added until the total downward force resulting from the weight of the slope is reached. This approach results in the following system of equations for the displacement,

$$\mathbf{K} \Delta \mathbf{u} = \mathbf{f} + \Delta \mathbf{f} - \mathbf{k}, \tag{1}$$

where $\Delta \mathbf{u}$ stands for the displacement increment and \mathbf{K} is the global stiffness matrix resulting from the assembly of element stiffness matrices \mathbf{K}^e , see § 3.2. The right hand side of Eq. (1) stands for the residual. Here, \mathbf{f} is the sum of the external force increments applied in the previous steps, $\Delta \mathbf{f}$ is the applied load increment of the current step and \mathbf{k} is the internal force resulting from the stresses. For a more thorough explanation on the methods used to solve the slope stability problem we refer to [11, Chapter 2 §4 and Chapter 7 §3 and §4].

For the mesh discretization of the slope stability problem we use second-order triangular Lagrangian Finite Elements, see Fig. 1. Here, the Finite Element nodal points are represented as black dots.

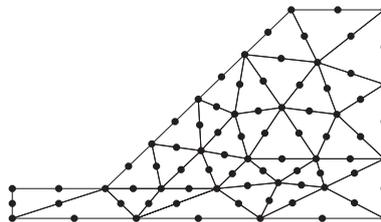


Figure 1: Mesh discretization used for the slope stability problem.

3 THEORETICAL BACKGROUND

In this section, we present some basic background on the usage of quasi-Monte Carlo (QMC) methods in estimating the expected value of a quantity of interest pertaining to the solution of

a PDE under uncertainty. We first explain the QMC estimator for estimating an integral and how to obtain an estimator on its variance, both for (tent-transformed) lattice sequences and (interlaced) Sobol' sequences. Next, we review how the uncertainty is modeled, in our slope stability problem, by means of a Karhunen–Loève expansion, and how it is accounted for in the equations of the Finite Element model. Last, we discuss how quasi-Monte Carlo points are generated according to (tent-transformed) lattice sequences and (interlaced) Sobol' sequences.

3.1 Quasi-Monte Carlo Estimator

The expected value of a function P against an s -dimensional probability density function ϕ is defined by

$$\mathbb{E}[P] := \int_{\mathbb{R}} \cdots \int_{\mathbb{R}} P(x_1, \dots, x_s) \phi(x_1, \dots, x_s) dx_1 \cdots dx_s = \int_{\mathbb{R}^s} P(\mathbf{x}) \phi(\mathbf{x}) d\mathbf{x}. \quad (2)$$

The integral in Eq. (2) can be approximated by means of an equal-weight quadrature rule, such as the Monte Carlo and quasi-Monte Carlo methods. In our setting, the function P stands for the quantity of interest based on the solution of our PDE which has stochastic parameters modeled by the variables $\mathbf{x} \sim \phi$. In order to approximate this expected value, both the multivariate integral, as well as the solution to the PDE for a given sample of the stochastic parameters, i.e., $P(\mathbf{x})$, will have to be approximated. We obtain an approximation for the quantity $P(\mathbf{x})$, resulting from our FEM discretization, which will be referred to as $P_L(\mathbf{x})$, where the L stands for the “discretization level”. Likewise, we do not compute the exact solution of the multidimensional integral, but approximate it by means of the quasi-Monte Carlo method.

To approximate Eq. (2) we employ a randomized quasi-Monte Carlo estimator of the form

$$Q_L^{\text{QMC}} := \frac{1}{R_L} \sum_{r=1}^{R_L} \frac{1}{N_L} \sum_{n=1}^{N_L} P_L(\Phi^{-1}(\mathbf{u}_L^{(n,r)})). \quad (3)$$

In here, $\mathbf{u}_L^{(n,r)}$ represent the points of our quasi-Monte Carlo point set, where n denotes the index of the point and r denotes the particular “random shift”. Since quasi-Monte Carlo points are defined on the unit cube $[0, 1]^s$ with respect to the uniform distribution, we need a mapping such that they act as samples from the density ϕ . For product densities, this can be achieved by applying the inverse of the cumulative distribution function component-wise. This is denoted by the mapping Φ^{-1} . Note that we have $\mathbb{E}[P] \approx \mathbb{E}[P_L] \approx Q_L^{\text{QMC}}$. The “random shifting” and mapping will be explained in §3.3 and §3.3.3, see also Fig. 2 for an illustration of Monte Carlo (MC) sampling points versus quasi-Monte Carlo (QMC) sampling points.

The reason that we use “randomized” quasi-Monte Carlo estimators is to obtain an unbiased estimator, as well as an error estimator.

By means of the R_L independent random shifts, Eq. (3) is in fact averaged over R_L estimators. Hence, the variance of the estimator can be estimated by

$$\mathbb{V}[Q_L^{\text{QMC}}] = \frac{\mathbb{V}[P_L]}{N_L R_L} \approx V_L^{\text{QMC}} := \frac{1}{(R_L - 1) R_L} \sum_{r=1}^{R_L} (P_L^{(n,r)} - Q_L^{\text{QMC}})^2, \quad (4)$$

where $P_L^{(n,r)} := P_L(\Phi^{-1}(\mathbf{u}_L^{(n,r)}))$. From Eq. (4), the root mean square error is estimated as

$\text{RMSE}(Q_L^{\text{QMC}}) = \sqrt{\mathbb{V}[Q_L^{\text{QMC}}]} \approx \sqrt{V_L^{\text{QMC}}}$. The RMSE will be used as an error estimator for the

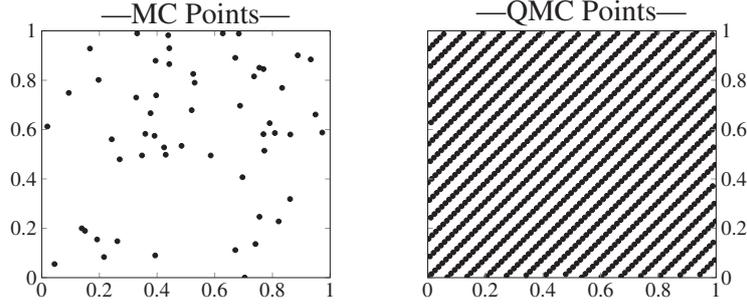


Figure 2: Example of MC and QMC sample points.

QMC estimator. We will plot this quantity in §4 in order to assess the accuracy of our numerical experiments.

3.2 Modeling the Uncertainty

The uncertainty present in the cohesion of the soil of the slope stability problem is modeled as a random field. Realizations of the random field are computed by means of the truncated Karhunen–Loève (KL) expansion,

$$Z(\mathbf{x}, \omega) = \bar{Z}(\mathbf{x}) + \sum_{n=1}^s \sqrt{\theta_n} \xi_n(\omega) b_n(\mathbf{x}), \quad (5)$$

where s is the number of terms in the expansion, i.e., the number of stochastic dimensions. Here, $\bar{Z}(\mathbf{x})$ is the mean of the field and $\xi_n(\omega)$ denote i.i.d. standard normal random variables. The eigenvalues θ_n and eigenfunctions $b_n(\mathbf{x})$ are the solutions of the eigenvalue problem

$$\int_D C(\mathbf{x}, \mathbf{y}) b_n(\mathbf{y}) d\mathbf{y} = \theta_n b_n(\mathbf{x}), \quad (6)$$

where $C(\mathbf{x}, \mathbf{y})$ is a given covariance kernel. The covariance kernel $C(\mathbf{x}, \mathbf{y})$ we consider for the random field is the Matérn covariance kernel

$$C(\mathbf{x}, \mathbf{y}) := \frac{\sigma^2}{2^{\nu-1} \Gamma(\nu)} \left(\frac{\sqrt{2\nu} \|\mathbf{x} - \mathbf{y}\|_2}{\lambda} \right)^\nu K_\nu \left(\sqrt{2\nu} \frac{\|\mathbf{x} - \mathbf{y}\|_2}{\lambda} \right), \quad (7)$$

where ν is the smoothness parameter, $K_\nu(\cdot)$ is the modified Bessel function of the second kind, $\Gamma(\cdot)$ is the gamma function, σ^2 is the variance, λ is the correlation length, and $\|\cdot\|_2$ is the L^2 norm. The integral in Eq. (6) is approximated by means of a numerical collocation scheme. For more information, we refer to [12, Chapter 7 Section 2].

In order to incorporate the uncertainty in the Finite Element model, we consider the integration point method, i.e., point evaluations of the random field are computed by means of Eq. (5) at the quadrature points and accounted for during numerical integration of the element stiffness matrix, see [13]. Here, the uncertainty resides in the elastoplastic constitutive matrix \mathbf{D} . This matrix is used for constructing the element stiffness matrices

$$\mathbf{K}^e = \int_{\Omega_e} \mathbf{B}^T \mathbf{D} \mathbf{B} d\Omega_e, \quad (8)$$

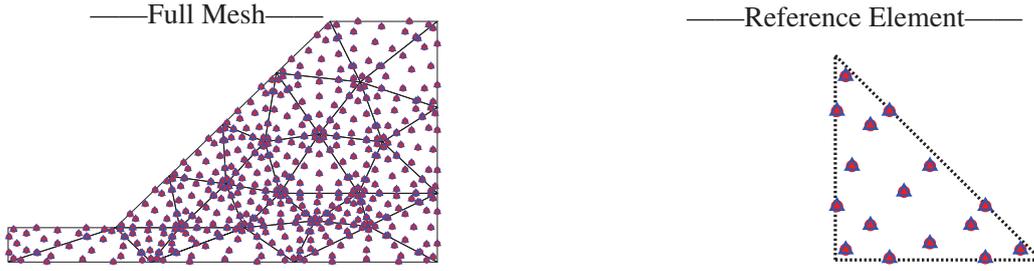


Figure 3: Locations of the quadrature points \triangle and the evaluation points of the random field \bullet .

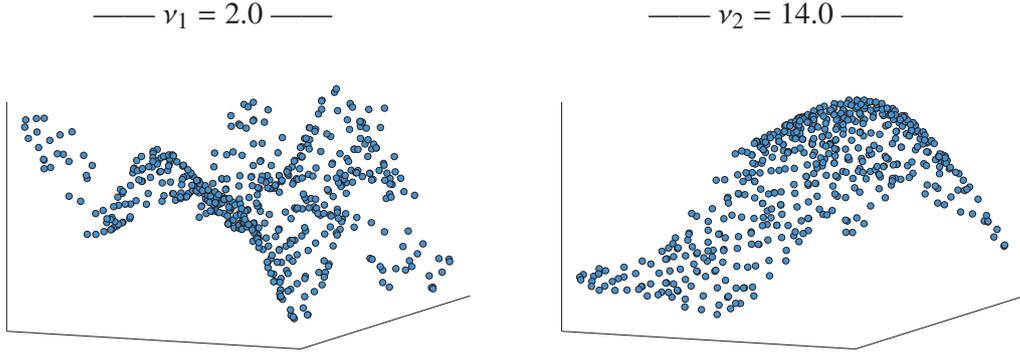


Figure 4: Point evaluations of an instance of a random field with $\nu_1 = 2.0$ and $\nu_2 = 14.0$.

with \mathbf{B} the matrix containing the derivatives of the element shape functions. In practice, the right-hand side in Eq. (8) is computed as

$$\mathbf{K}^e \approx \sum_{i=1}^{|\mathbf{q}|} \mathbf{B}_i^T \mathbf{D}_i \mathbf{B}_i w_i, \quad (9)$$

with $\mathbf{B}_i = \mathbf{B}(\mathbf{q}^{(i)})$ the matrix \mathbf{B} evaluated at the i th quadrature point $\mathbf{q}^{(i)}$, $\mathbf{D}_i = \mathbf{D}(\omega^{(i)})$ the elastoplastic constitutive matrix \mathbf{D} containing the value of the uncertain soil cohesion $\omega^{(i)}$, computed as a point evaluation of the random field at $\mathbf{q}^{(i)}$, and w_i the quadrature weight.

With the integration point method, Eq. (5) is evaluated in the quadrature points used for the numerical integration of Eq. (9), i.e., $\mathbf{x} = \mathbf{q}$. This is illustrated in Fig. 3, where the set \mathbf{q} is represented by the \triangle , and the random field evaluation points, \mathbf{x} , are represented by \bullet .

In order to represent the uncertainty of the cohesion of the soil, we use a lognormal random field. This field is obtained by applying the exponential to the field obtained in Eq. (5) component-wise, i.e., $Z_{\text{lognormal}}(\mathbf{x}, \omega) = \exp(Z(\mathbf{x}, \omega))$. After this mapping, the lognormal random field has a mean of 8.02 kPa and a standard deviation of 100 Pa. We consider a random field with a correlation length $\lambda = 1.5$, a variance $\sigma^2 = 1$, and a stochastic dimension $s = 100$. Two different values for the smoothness ν of the random field are considered: $\nu_1 = 2.0$ and $\nu_2 = 14.0$. The smoothness parameter governs the smoothness of the random field: a lower value for ν implies a rougher random field and vice versa. In Fig. 4, we show instances of this random field, for the two different smoothness parameters ν_1 and ν_2 .

3.3 Quasi-Monte Carlo Sampling

Quasi-Monte Carlo points are deterministic low-discrepancy points used for numerical integration. Different approaches exist to generate these points. We consider two approaches, rank-1 lattice sequences and Sobol' sequences. Classically, these point sets have been constructed in order to be used for integration against the uniform distribution on the unit cube $[0, 1]^s$. In §3.3.3 we will describe how to use these point sets for integration against the normal distribution, as required in the KL expansion in Eq. (5). Important to note is that although we describe the technical details behind the construction of these point sets here, the end user can just use a library routine to generate them, see, e.g., [14, 15, 16].

3.3.1 Lattice Sequences

The points belonging to a rank-1 lattice sequence are determined by a generating vector $\mathbf{z} \in \mathbb{Z}^s$, which consists of one integer value per considered stochastic dimension s . The choice of this generating vector determines the quality of the sample points. When the total number of points is a fixed number N then the n th point of the “lattice rule” (instead of a lattice sequence) is given by

$$\mathbf{u}^{(n)} := \text{frac}\left(\frac{n}{N} \mathbf{z}\right) \quad \text{for } n = 0, \dots, N-1, \quad (10)$$

where $\text{frac}(\cdot)$ denotes the function that takes the fractional part. When written in this form, the number of points cannot be extended beyond the maximal number of points N , and there is no guarantee that using an initial amount of these points will have a “nice” distribution in the unit cube. In order to obtain a sequence of “nicely” distributed sample points, we instead define the n th point by

$$\mathbf{u}^{(n)} := \text{frac}(\phi_2(n) \mathbf{z}) \quad \text{for } n = 0, 1, \dots, \quad (11)$$

where ϕ_2 stands for the radical inverse function in base 2, see, e.g., [17, 18], and possibly limiting $n < N_{\max}$ for some large enough N_{\max} . The radical inverse function in base 2 mirrors the binary representation of a number around its binary point, e.g., $6 = (110)_2$, then $\phi_2((110)_2) = (0.011)_2 = 0.375$. Algorithms to find good generating vectors for Eq. (10) and Eq. (11) are known, and such vectors can be found in the literature, see, e.g., [18].

As already touched upon in §3.1, the deterministic nature of quasi-Monte Carlo sample points introduces an additional bias on the stochastic quantities of the computed solutions. Therefore, “randomness” needs to be reintroduced in order to obtain unbiased estimates. This is accomplished by a procedure called “random shifting”. The procedure consists of adding to each point of the lattice sequence, a uniformly distributed number $\mathbf{w} \in [0, 1]^s$, after which the fractional part is taken. This is illustrated in Fig. 5. By using R independent random shifts, the resulting R independent estimators can also be used to estimate the variance of our QMC estimator, and hence providing us with an error estimator.

The shifted version of Eq. (11) is then given by

$$\mathbf{u}^{(n,r)} := \text{frac}(\phi_2(n) \mathbf{z} + \mathbf{w}_r) \quad \text{for } n = 0, 1, \dots \quad \text{and } r = 1, 2, \dots, R. \quad (12)$$

The use of such a lattice sequence, constructed with a “good” generating vector, in the QMC estimator from Eq. (3), can achieve a theoretical order of convergence of $O(N^{-1})$ in a certain

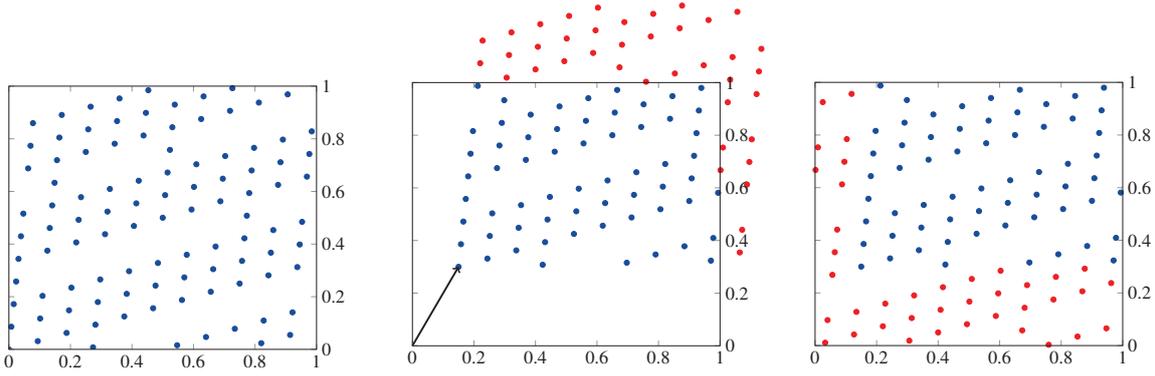


Figure 5: Random shifting procedure applied to points belonging to a rank-1 lattice sequence.

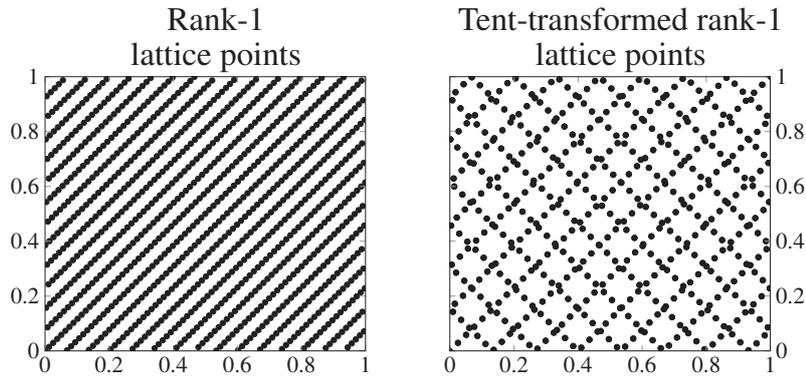


Figure 6: Rank-1 lattice points and tent-transformed rank-1 lattice points in the unit cube.

Sobolev space which contains integrands with square-integrable mixed first derivatives see, e.g., [4] for details. However, by applying the tent transformation

$$T(y) := 1 - |2y - 1|, \quad (13)$$

component-wise to the points generated by Eq. (12), and defining

$$\mathbf{v}^{(n,r)} := T(\mathbf{u}^{(n,r)}), \quad (14)$$

for usage in Eq. (3) instead of the original points $\mathbf{u}^{(n,r)}$, it is possible to obtain a better order of convergence than $O(N^{-1})$, provided that the integrand is sufficiently smooth, see, e.g., [8, 19, 20]. In our numerical experiments, this smoothness will be influenced by the smoothness parameter ν of the random field, and by the order of approximation of the FEM solution. We note that in order to achieve higher-order convergence it is necessary to only consider estimators where the value for N_L is a power of 2. This has the effect that the point set will act like a sequence of embedded lattice rules. See Fig. 6 for an illustration of tent-transformed lattice points.

3.3.2 Sobol' Sequences

The Sobol' sequence is a “digital net” in base 2, that uses a binary generating matrix per dimension. We denote these matrices by C_1, \dots, C_j for $j = 1, \dots, s$. For the Sobol' sequence these

Decimal	Binary	Gray code	Gray code binary	Multiplication with C_4	Shifted with $(0.010)_2$	Decimal
0	000	0	000	0.000	0.010	0.25
1	001	1	001	0.100	0.110	0.75
2	010	3	011	0.010	0.000	0.00
3	011	2	010	0.110	0.100	0.50
4	100	6	110	0.111	0.101	0.625

Table 1: Gray coded and digitally shifted Sobol' points in dimension 4.

matrices are upper triangular. To obtain the j th dimension of the n th Sobol' point, we write $n = (n_{m-1} \cdots n_1 n_0)_2$ in binary representation, and use the $m \times m$ binary matrix C_j to compute

$$\begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix} = \begin{pmatrix} & & \\ & C_j & \\ & & \end{pmatrix} \begin{pmatrix} n_0 \\ \vdots \\ n_{m-1} \end{pmatrix}, \quad (15)$$

where all additions and multiplications are carried out modulo 2. The j th dimension of the n th point is then given by interpreting the output vector as the binary expansion, i.e., $u_j^{(n)} = (0.y_1 y_2 \cdots y_m)_2$. We demonstrate the approach stated above for the computation of the first five points of the fourth dimension. Suppose the 3×3 subset of the generating matrix C_4 is given by

$$C_4 = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (16)$$

The first five points generated by C_4 for $n = 0, 1, 2, 3, 4$, which in binary are $0 = (000)_2, 1 = (001)_2, 2 = (010)_2, 3 = (011)_2$ and $4 = (100)_2$, hence are equal to $(0.000)_2 = 0, (0.100)_2 = 0.5, (0.110)_2 = 0.75, (0.010)_2 = 0.25$, and $(0.001)_2 = 0.125$. In practice, one avoids multiplying with the full matrix C_j by generating the points in Gray code ordering. This has the benefit that the next point can be obtained from the previous one by adding only the column of the generating matrix where the bit was changed. This changes the ordering of the Sobol' points, but they will still have their "nice" distribution properties.

In order to obtain an unbiased estimator, and an error estimator, we need to introduce some "randomness" on the deterministic points. This is accomplished by means of a "random digital shift". The digitally shifted n th Sobol' point is obtained by adding the bits of the binary expansion of the shift to each digit of the Sobol' point modulo 2, for each dimension, i.e., $\mathbf{u}^{(n,r)} = \mathbf{u}^{(n)} \oplus \mathbf{w}_r$. This is illustrated in Tab. 1 for the first 5 Gray coded points in dimension 4 with a shift \mathbf{w} that has the value $w_4 = 0.25 = (0.010)_2$ as its fourth component.

It has been shown in the work of Dick, see e.g., [21], that higher-order convergence can be obtained by a "digit interlacing" technique if the integrand is sufficiently smooth. Again, in our numerical experiments this smoothness will be influenced by the smoothness of the random field and by the order of approximation of the FEM solution. In order to obtain Sobol' points with interlacing factor α in s dimensions, one starts with Sobol' points in αs dimensions and then "interlaces" α dimensions into a single dimension by interlacing the bits of the points. For example, for an interlacing factor of 2 we take the binary representations of the Sobol' points in the first two dimensions as $x = (0.x_1 x_2 \cdots x_m)_2$ and $y = (0.y_1 y_2 \cdots y_m)_2$ and then form the point $(0.x_1 y_1 x_2 y_2 \cdots x_m y_m)_2$. In practice, the interlacing of the Sobol' sequence can also be done by interlacing the rows of the generating matrices of the original Sobol' sequence. The interlaced

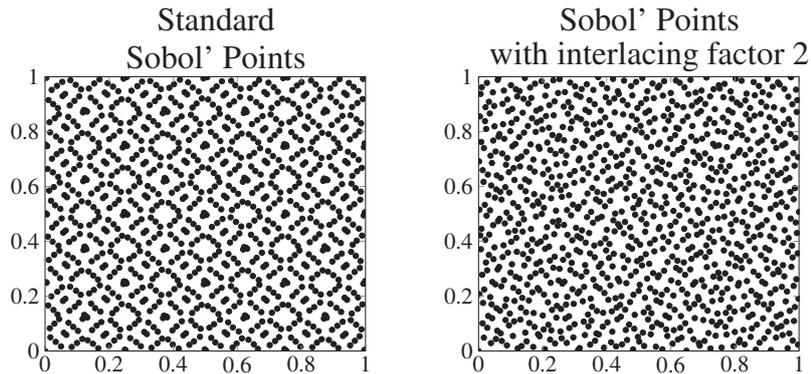


Figure 7: Standard Sobol' points and Sobol' point with interlacing factor 2.

Sobol' sequence points are then generated from these interlaced matrices which have dimension $am \times m$. As was the case with the tent-transformed lattice points, we need to ensure that we only consider estimators where N_L is a power of 2 in order to achieve higher order convergence. An illustration of interlaced Sobol' points is given in Fig. 7.

3.3.3 Using QMC points for integration against the truncated normal density

Up till now the discussion was centered on quasi-Monte Carlo points which are sampled uniformly from the unit cube $[0, 1]^s$. In order to use these points for integration against some other distribution and domain, see Eq. (2), we need to map them to the adequate distribution. This can be achieved by applying the inverse of the cumulative distribution function component-wise.

Our modeling of the random field, see § 3.2, involves standard normally distributed numbers. However, for our experiments in which we hope to achieve a convergence better than order 1 in estimating the expectations, i.e., $O(N^{-1})$, we will truncate the domain from $(-\infty, \infty)^s$ to $[-b, b]^s$ for some choice of $0 < b < \infty$. With respect to our general scheme of approximating Eq. (2) this will introduce an additional domain truncation error in our sequence of approximations $\mathbb{E}[P] \approx \mathbb{E}[P_L] \approx Q_L^{\text{QMC}}$. There is no established analysis for obtaining higher order quasi-Monte Carlo convergence on the infinite domain $(-\infty, \infty)^s$, except for on the unit cube, see, e.g., [3, 9]. Therefore, we follow the truncation approach which has been used in other references as well, see, e.g., [22, 23]. The following errors need to be balanced: (1) the Finite Element discretization error which is adjusted by choosing different discretization parameters L and p , (2) the dimension truncation error which results from truncating the infinite KL expansion to only s terms, see Eq. (5), (3) the domain truncation error which results from the truncation of $(-\infty, \infty)^s$ to $[-b, b]^s$ and (4) the quadrature/cubature approximation error which results from approximating an s -dimensional integral with an s -dimensional QMC rule using N_L sample points. The theoretical analysis and careful balancing of these different sources of error will be the topic of future research. In this work, we demonstrate numerically that it is possible to achieve a higher-order convergence.

4 Results

In this section, we present numerical results obtained by applying a tent-transformed lattice sequence and an interlaced Sobol' sequence to the slope stability problem introduced in §2.

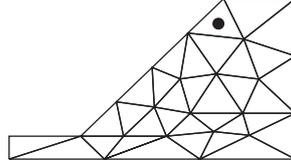


Figure 8: The QoI as the vertical displacement of the center point of the upper left element of the model, indicated by a dot.

Rank-1 lattice points are computed with the Julia package **LatticeRules.jl**, where the generating vector \mathbf{z} was constructed with the component-by-component (CBC) algorithm with order 2 weights, see [24]. The Sobol' points are computed with the Julia package **DigitalNets.jl**, see [15]. We used the Sobol' points and the pre-interlaced Sobol' generating matrices from [16, 25]. Instances of the random field are computed with the Julia package **GaussianRandomFields.jl**, see [26]. We use the in-house Finite Element **MATLAB** code developed by the Structural Mechanics section of the KU Leuven. All results have been computed on a workstation equipped with 2 physical cores, Intel Xeon E5-2680 v3 CPU's, each with 12 logical cores, clocked at 2.50 GHz, and a total of 128 GB RAM.

As the quantity of interest (QoI), we choose the vertical displacement of the center point of the upper left element of the model, see Fig. 8. By choosing a QoI located inside the element, we ensure that the displacement is represented by a quadratic polynomial, since quadratic shape functions are used to compute a displacement at this center point. This approach is followed as to ensure that the higher-order derivatives are continuous. Between elements there exists only C0 continuity. However, inside the elements, the continuity is as high as the order of the shape functions used for the representation of the solution. Hence, we interpolate to a point located inside the element. The spatial dimensions of the slope, in our slope stability problem, consist of a length of 20m, a height of 14m and a slope angle of 30° . The material characteristics are, a Young's modulus of 30MPa, a Poisson ratio of 0.25, a density of 1330kg/m^3 and a friction angle of 20° . Plane strain is considered for this problem. The characteristics of the random field considered to model the uncertainty in the cohesion of the soil, are given in § 3.2. For the truncation of the domain, see § 3.3.3, we take a value $b = 2$, and thus truncated the domain to $[-2, 2]^s$.

First, in §4.1, we numerically verify that each method computes the same expected value. Next, in §4.2, we show the convergence of the root mean square error of each method with respect to the number of samples taken.

4.1 Convergence of the Expected Value

In Fig. 9, we show the maximum absolute error on the bound of the 95% confidence interval of the expected value in function of the number of samples,

$$\text{Max Error} := \max\left\{\left|Q[P_L]_{95\%,\text{top}} - Q[P_{L,\text{Ref}}]\right|, \left|Q[P_L]_{95\%,\text{bottom}} - Q[P_{L,\text{Ref}}]\right|\right\}, \quad (17)$$

where $Q[P_L]$ is computed according to Eq. (3) and the top and the bottom of the 95% confidence interval are obtained by $Q[P_L]_{95\%,\text{top}} := Q[P_L] + 1.96 V_L^{\text{QMC}}$ and $Q[P_L]_{95\%,\text{bottom}} := Q[P_L] - 1.96 V_L^{\text{QMC}}$ with V_L^{QMC} computed according to Eq. (4) for each method. As the reference value $Q[P_{L,\text{Ref}}]$ we take the average of our final approximations obtained by the interlaced Sobol'

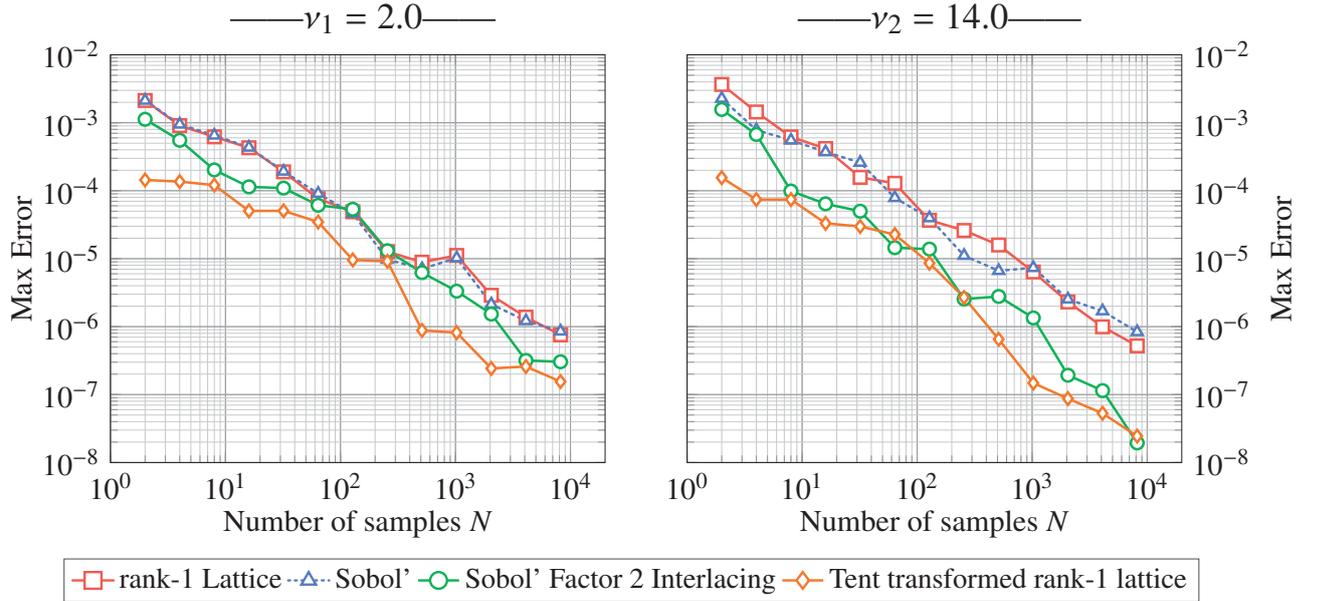


Figure 9: Absolute error on the expected value in function of the number of samples.

sequence and the tent-transformed lattice sequence, computed with 8192 samples and 8 shifts. The results from this simulation are chosen, because the root mean square error yields the lowest value, see Fig. 10, and we thus expect these values to be the most accurate. The numerical values for $Q[P_{L,Ref}]$ are 0.05415452 for $\nu_1 = 2.0$, and 0.05419605 for $\nu_2 = 14.0$.

From the graphs in Fig. 9 we can confirm that all estimators converge to the same value. For the case with $\nu_1 = 2.0$ all estimators converge approximately with $O(N^{-1})$ which is as expected due to the limited smoothness of the random field. For the case with $\nu_2 = 14.0$ we obtain $O(N^{-1})$ convergence for the plain QMC sequences, i.e., the lattice sequence and the Sobol' sequence, and we observe, as expected, an improved convergence of $O(N^{-1.5})$ for the tent-transformed lattice sequence and the interlaced Sobol' sequence.

4.2 Convergence of the Root Mean Square Error of the Estimator

In Fig. 10, we show the RMSE (root mean square error) of the estimators in function of the number of samples N , see Eq. (4). We observe that for a smoothness parameter $\nu_1 = 2.0$ in the Matérn kernel, the order of convergence for all approaches is $O(N^{-1})$ which is as expected. We do notice that in our example the tent-transformed lattice sequence has an RMSE which is approximately a factor 10 lower than the standard lattice sequence. For the case of the higher smoothness $\nu_2 = 14.0$, we observe that using the tent-transformed lattice sequence and the interlaced Sobol' sequence achieves an order of convergence close to $O(N^{-1.5})$. For the vanilla versions of the lattice sequence and the Sobol' sequence, a classical quasi-Monte Carlo convergence of $O(N^{-1})$ is achieved, as expected. We conclude that the combination of a smooth random field, i.e., ν large, and higher order elements, in combination with a tent-transformed lattice sequence or an interlaced Sobol' sequence shows higher order convergence in our experiment.

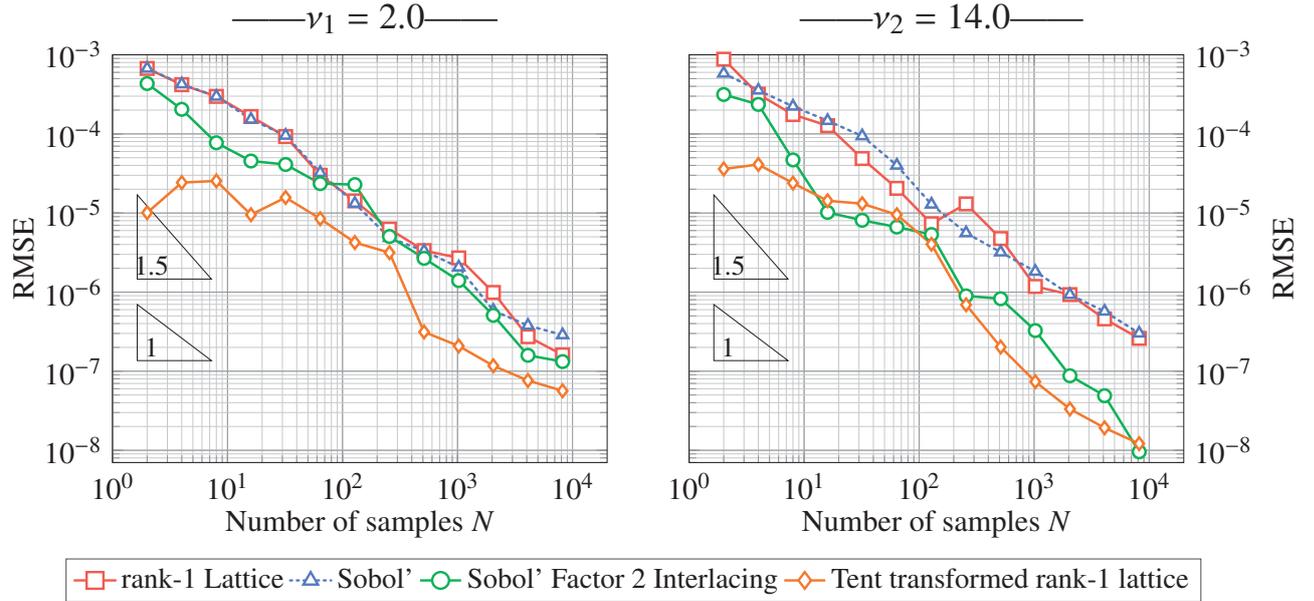


Figure 10: RMSE of the estimator in function of the number of samples.

5 CONCLUSIONS

In this work, we investigated two techniques to attain higher-order convergence by means of quasi-Monte Carlo methods in approximating expectations in a geotechnical slope stability problem. These techniques are tent-transformed lattice sequences and interlaced Sobol' sequences. We discussed how the deterministic quasi-Monte Carlo points for these point sets are obtained. We benchmarked these two techniques on a slope stability problem where the cohesion of the soil has a spatially varying uncertainty. The uncertainty is represented as a lognormal random field, and realizations of the random field are computed using a truncated KL expansion. We illustrated that, for a sufficiently smooth random field combined with tent-transformed rank-1 lattice points or with interlaced Sobol' points, and higher order elements, an order of convergence of $O(N^{-1.5})$ can be obtained. We compared the numerical results obtained by these two techniques against numerical results where a standard rank-1 lattice sequence, or a non-interlaced Sobol' sequence is used. For these latter two approaches, we observed the classical order of convergence of $O(N^{-1})$.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the support from the Research Council of KU Leuven through project C16/17/008 "Efficient methods for large-scale PDE-constrained optimization in the presence of uncertainty and complex technological constraints". The computational resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation - Flanders (FWO) and the Flemish Government - department EW. I.

REFERENCES

- [1] Fishman, G. S. *Monte Carlo: Concepts, algorithms and applications*. Springer-Verlag, New York (1996).
- [2] Giles, M. B. Multilevel Monte Carlo methods. *Acta Num.* (2015)**24**:259–328.
- [3] Kuo, F. Y. and Nuyens, D. Application of quasi-Monte Carlo methods to elliptic PDEs with random diffusion coefficients: A survey of analysis and implementation. *Found. Comput. Math.* (2016)**16**(6):1631–1696.
- [4] Dick, J., Kuo, F. Y., and Sloan, I. H. High-dimensional integration: The quasi-Monte Carlo way. *Acta Num.* (2013)**22**:133–288.
- [5] Blondeel, P., Robbe, P., Van hoorickx, C., Lombaert, G., and Vandewalle, S. Multilevel sampling with Monte Carlo and Quasi-Monte Carlo methods for uncertainty quantification in structural engineering. In *Proceedings of the 13th International Conference on Applications of Statistics and Probability in Civil Engineering, ICASP13, Seoul, South Korea*. Published in S-space Seoul National University Open Repository (2019) pages 383–390.
- [6] Blondeel, P., Robbe, P., Van hoorickx, C., François, S., Lombaert, G., and Vandewalle, S. P-Refined Multilevel Quasi-Monte Carlo for Galerkin Finite Element Methods with Applications in Civil Engineering. *Algorithms* (2020)**13**(5):1–30.
- [7] Blondeel, P., Robbe, P., François, S., Lombaert, G., and Vandewalle, S. Benchmarking the p-MLQMC Method on a Geotechnical Engineering Problem. In F. Chinesta, R. Abgrall, O. Allix, and M. Kaliske, editors, *Proceedings of the 14th World Congress on Computational Mechanics (WCCM) and ECCOMAS Congress 2020, Virtual Congress*. Published in Scipedia (2021) pages 1–12.
- [8] Dick, J., Nuyens, D., and Pillichshammer, F. Lattice rules for nonperiodic smooth integrands. *Num. Math.* (2014)**126**:259–291.
- [9] Dick, J., Kuo, F. Y., Le Gia, Q. T., Nuyens, D., and Schwab, C. Higher order QMC petrov-galerkin discretization for affine parametric operator equations with random field inputs. *SIAM J. Numer. Anal.* (2014)**52**(6):2676–2702.
- [10] Whenham, V., De Vos, M., Legrand, C., Charlier, R., Maertens, J., and Verbrugge, J.-C. Influence of soil suction on trench stability. In T. Schanz, editor, *Experimental Unsaturated Soil Mechanics*. Springer Berlin Heidelberg (2007) pages 495–501.
- [11] de Borst, R., Crisfield, M. A., and Remmers, J. J. C. *Non Linear Finite Element Analysis of Solids and Structures*. Wiley, U.K. (2012).
- [12] Lord, G. J., Powell, C. E., and Shardlow, T. *An Introduction to Computational Stochastic PDEs*. Cambridge Texts in Applied Mathematics. Cambridge University Press (2014).
- [13] Brenner, C. E. and Bucher, C. A contribution to the sfe-based reliability assessment of nonlinear structures under dynamic loading. *Probab. Eng. Mech.* (1995)**10**(4):265 – 273. ISSN 0266–8920.
- [14] Robbe, P. Latticerules.jl (2017). Online <https://github.com/PieterjanRobbe/LatticeRules.jl>, accessed on 05/11/2020.

- [15] Blondeel, P. Digitalnets.jl (2020). Online <https://github.com/Philippe1123/DigitalNets.jl>, accessed on 05/11/2020.
- [16] Nuyens, D. The “Magic Point Shop” of QMC point generators and generating vectors. Online <https://people.cs.kuleuven.be/~dirk.nuyens/qmc-generators/>, accessed on 05/05/2020.
- [17] Hickernell, F. J., Hong, H. S., L’Ecuyer, P., and Lemieux, C. Extensible lattice sequences for Quasi-Monte Carlo quadrature. *SIAM J. Sci. Comput.* (2000)**22(3)**:1117–1138.
- [18] Cools, R., Kuo, F. Y., and Nuyens, D. Constructing embedded lattice rules for multivariate integration. *SIAM J. Sci. Comput.* (2006)**28(6)**:2162–2188.
- [19] Cools, R., Kuo, F. Y., Nuyens, D., and Suryanarayana, G. Tent-transformed lattice rules for integration and approximation of multivariate non-periodic functions. *J. Complexity* (2016)**36**:166–181.
- [20] Hickernell, F. J. Obtaining $o(n^{-2+\epsilon})$ convergence for lattice quadrature rules. In K.-T. Fang, H. Niederreiter, and F. J. Hickernell, editors, *Monte Carlo and Quasi-Monte Carlo Methods 2000*. Springer Berlin Heidelberg, Berlin, Heidelberg (2002) pages 274–289.
- [21] Dick, J. Higher order scrambled digital nets achieve the optimal rate of the root mean square error for smooth integrands. *Ann. Stat.* (2011)**39(3)**:1372 – 1398.
- [22] Nguyen, D. T. P. and Nuyens, D. MDFEM: Multivariate decomposition finite element method for elliptic PDEs with lognormal diffusion coefficients using higher-order QMC and FEM. *arXiv: 1904.13327 [math.NA]* (2020).
- [23] Dick, J., Irrgeher, C., Leobacher, G., and Pillichshammer, F. On the optimal order of integration in hermite spaces with finite smoothness. *SIAM J. Numer. Anal.* (2018)**56(2)**:684–707.
- [24] Nuyens, D. Lattice rule generating vectors (2007). Exod2_base2_m20_CKN at Online <https://people.cs.kuleuven.be/~dirk.nuyens/qmc-generators/>, accessed on 12/04/2020.
- [25] Joe, S. and Kuo, F. Sobol sequence generator. Online <https://web.maths.unsw.edu.au/~fkuo/sobol/>, accessed on 05/05/2020.
- [26] Robbe, P. Gaussianrandomfields.jl (2017). Online <https://github.com/PieterjanRobbe/GaussianRandomFields.jl>, accessed on 05/11/2020.

UNCERTAINTY QUANTIFICATION IN THE CLOUD WITH UQCLOUD

Christos Lataniotis¹, Stefano Marelli¹, and Bruno Sudret¹

¹Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich
Stefano-Francini-Platz 5, 8093, Zurich, Switzerland
e-mail: {lataniotis,marelli,sudret}@ibk.baug.ethz.ch

Keywords: Uncertainty Quantification, UQLab, UQCloud

Abstract. *General-purpose uncertainty quantification software has become a well established requirement in modern engineering workflows. Different communities (e.g. applied maths, engineering, economics, etc.), however, generally employ diverse arrays of technologies and workflows, from computing infrastructure to programming languages. To overcome the intrinsic limitation of a single language, standalone software package, we introduce UQCLOUD, an OS- and programming language- independent, cloud-based version of UQLAB. UQCLOUD follows a software-as-a-service (SaaS) model, that allows anyone to take advantage of the well-established UQLAB suite, without the need to adapt their computational workflows to include MATLAB.*

1 INTRODUCTION

Including uncertainty quantification (UQ) in modern research and engineering practice has become a well established trend in the past decades. From academic research to *validation, verification and uncertainty quantification* (VVUQ) in engineering design [8], dealing with uncertainty is becoming a standard requirement, even for regulatory bodies. This trend has been enabled by the constantly increasing availability of accurate computational models and related high performance computing infrastructure, and at the same time by recent developments in general-purpose uncertainty-quantification software. A remarkable offering of the latter is available at the time of this writing, from the well-established C++-based Dakota project in the United States [3], to the PYTHON-based Open-TURNS in France [1], the MATLAB-based COSAN in the UK [10], and the more recent, once again PYTHON-based [4] and [9], to mention a few. Along the same lines, the MATLAB-based UQLAB software framework [5] began its design and development phase back in 2013, with the goal of providing a state-of-the-art uncertainty quantification software that would be accessible to applied scientists of all backgrounds, regardless of their programming experience. Given its widespread adoption, counting over 3,500 unique registered users worldwide since July 2015, it is clear that both its set of features and its intuitive, beginner-friendly interface have hit a sweet-spot in our intended audience.

As an effort to further disseminate the adoption of advanced UQ tools across disciplines, in 2019 we launched the UQWORLD online community ([6], <https://uqworld.org/>), which now counts hundreds of users worldwide. Among the feedback collected by several users on that platform, what is commonly seen as a deterrent in the adoption of UQLAB in a number of communities, especially outside engineering, is the programming language choice. The classical “standalone software” paradigm, poses an important limitation in its adoption, especially in industrial contexts: it can be difficult to incorporate it into existing workflows that make use of different software.

Increasing portability outside the pure cross-OS compatibility offered by MATLAB is not an easy task to achieve. UQLAB has reached a remarkable degree of maturity, with over a hundred thousands lines of optimized code that in many cases capitalize on the efficient linear algebra facilities offered by MATLAB. Porting the entire software to a different language would therefore be complex both from a technical perspective (significant expertise would be needed in both the underlying theoretical tools and in the programming language of choice), and from a practical one (it may result in actually slower results, and it would create a maintenance nightmare). UQLAB is also an active project, constantly evolving at its own pace thanks to the work of the researchers and developers from the Chair of Risk, Safety and Uncertainty Quantification in ETH Zurich. Adding to this the rapidly evolving landscape of scientific programming languages (e.g., PYTHON 3 vs PYTHON 2.7, the rise of Julia, the prevalence of C in certain modeling communities), makes it clear that any port to a specific language would only be a temporary patch to fix a greater underlying issue.

With this in mind, we started exploring modern alternatives to the classical standalone software model, that could provide an OS- and programming-language-agnostic, portable version of UQLAB. In this contribution we present UQCLOUD, a modern *software-as-a-service* (SaaS) incarnation of UQLAB that aims at solving the three key aspects of portability, continuous integration with the main UQLAB software, and performance.

In this paper we present a brief review of the state of the UQLAB ecosystem, and then introduce the main structure of UQCLOUD, with an outlook to its implications. To showcase the advantage of such an infrastructure, we also showcase UQ[PY]LAB, a set of software bindings

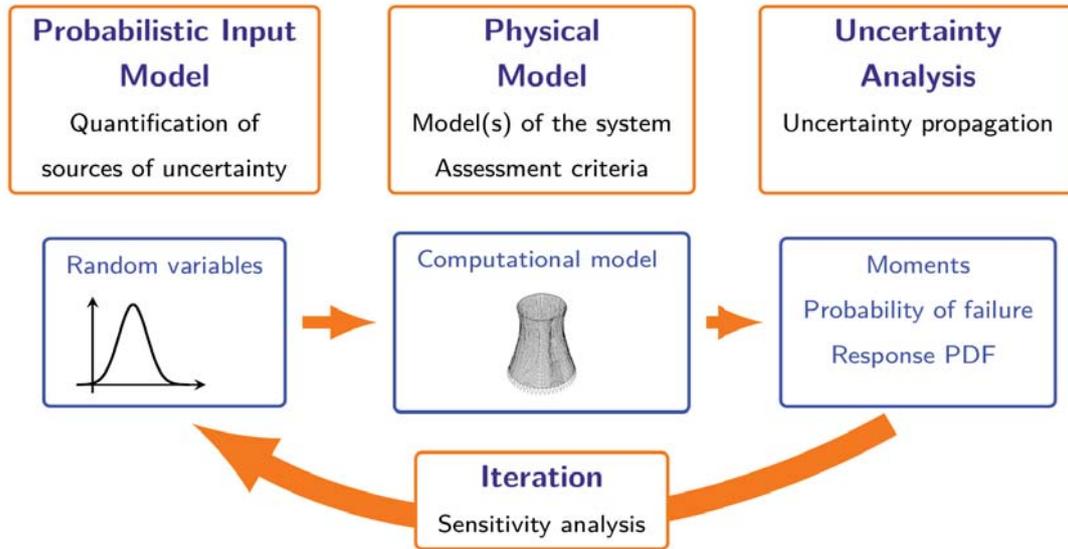


Figure 1: The general UQ framework for managing uncertainty (after [12, 2]) that defines the UQLAB semantics.

that provide full access to UQLAB in PYTHON.

2 THE UQLAB ECOSYSTEM

2.1 The UQLab software framework

At the core of the UQLAB project, lies the UQLAB software framework (www.uqlab.com), initially released in 2017 after a two years-long beta testing phase. Designed around the general framework for UQ originally developed in [12, 2] and illustrated in Figure 1, UQLAB provides a natural sandbox for the design and development of complex UQ analyses, without requiring extensive technical knowledge from the users. As of March 2021, with over 3,500 unique registered users from 90 countries worldwide, UQLAB has become a household name in the UQ software community.

In terms of software and development model, UQLAB has reached the maturity stage. At the same time, its wide adoption in largely different fields (ranging from theoretical physics to medical engineering, from computational macroeconomics to disease propagation) is a testimony to the growing need of a software designed around perspective users, rather than just around the profiles of the developers. Its self-imposed target of disseminating UQ outside the bounds of traditionally UQ-savvy fields has been successfully maintained in the past few years.

UQLAB is a public mirror to the research conducted at the Chair of Risk, Safety and Uncertainty Quantification at ETH Zurich (www.rsuq.ethz.ch). Its scientific content (embedded in the open source UQLABMODULES) keeps therefore growing at a steady pace, and it is not expected to relent any time soon.

2.2 UQWORLD: a global UQ community

While on the one hand the rapid diffusion of UQLAB demonstrated that it helped filling the niche of general-purpose, accessible UQ software, it also highlighted a second need in the scientific community. Since its early stages, the UQLAB userbase has been highly varied in the degree of technical skills, academic background, and applications. No single resource, however, was available to group together such a large melting pot of different professionals. In

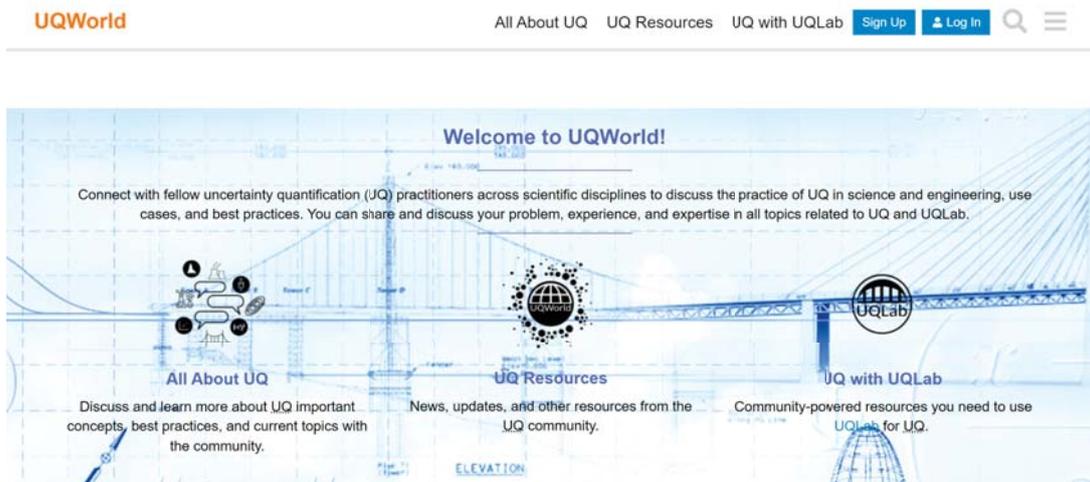


Figure 2: The UQWORLD homepage

2019, the UQWORLD online community [6] was created, with the goal of providing an open forum for UQ practitioners and experts to get in touch and discuss about their needs, or share their expertise.

While a section of the community website (see Figure 2) is dedicated to the use of UQLAB, the vast majority of the available resources are software-independent, and as of today several hundreds of users interact on the community forums, seeking for answers and providing feedback and suggestions. While still in its infancy, UQWORLD is rapidly gaining momentum, and it provides a good starting point to get a pulse of the needs of UQ practitioners worldwide.

A UQLAB-related topic that comes up the community discussions (see, e.g., <https://uqworld.org/t/matlab-but-but-python-r-c-cobol/147>), is the choice of developing UQLAB in MATLAB. Many community members felt the choice of MATLAB somewhat exclusive, as it does not cater to the entire userbase of UQLAB, something that is essentially impossible due to its sheer variety. With this in mind, we started brainstorming about a possible solution to this issue, that would not require translating UQLAB in a number of different languages, each with their own strengths and weaknesses, and highly specialized syntax. The result is UQCLOUD, a modern take on software that is independent on the user platform.

3 INTRODUCING UQCLOUD

3.1 A paradigm shift: from local software, to cloud-based software-as-a-service

UQCLOUD aims at bridging the gap between the UQLAB software and any development or production environment that does not include MATLAB. Due to the numerous challenges involved in translating and maintaining several UQLAB variants in different programming languages, UQCLOUD adopts a more modern software-as-a-service (SaaS) paradigm. In a nutshell, it provides an online programming interface (API) that provides UQ computations on demand, by relying on a UQLAB-powered service running on one or more cloud computing instances. Because the UQLAB service is a deployed program created through the MATLAB deployment toolbox, it is itself, as a matter of fact, MATLAB independent.

The UQCLOUD platform consists of several instances that are deployed on cloud premises. An abstract view of the components of such an instance is shown in Figure 3. Each box denotes

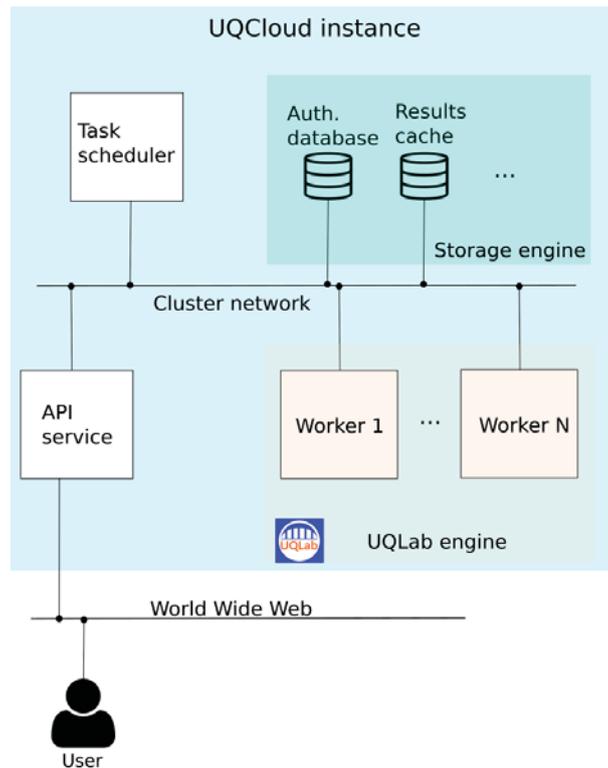


Figure 3: An abstract view of the components of a UQCLOUD instance.

a containerised computing environment [7]. Such environments are virtual lightweight replicas of separate computing nodes, each performing a predefined set of tasks.

At the core of UQCLOUD lies the so-called UQLAB *engine*. It consists of several containers, the *workers*. Each worker is designed to execute any UQLAB-related computation that it is tasked with. To ensure licensing compliance, no MATLAB session is running in any component of UQCLOUD. Rather, a standalone UQLAB-based custom designed executable server is deployed through the MATLAB Compiler^{TM,1}. The workers require therefore no MATLAB installation (and associated licensing).

In practice, users do not directly interface with the UQLAB workers. Instead, they contact the API service container, either directly or through dedicated software bindings. The communication follows the secure industry-standard REST API paradigm (see e.g. [11]). The role of this service is to receive computation requests from the users and return the results in an asynchronous manner. Only authenticated users can submit such requests, which serves two purposes: (1) access control to cloud computing resources, and, (2) performing each UQ computation within a persistent UQLAB session that is dedicated to that user.

The latter is a major requirement for the core UQCLOUD offering, which aims to deliver virtually all the features provided by UQLAB. In this setting, users can perform arbitrary workflows regardless of their complexity. This is made possible by an underlying persistent, user-specific UQLAB session on the cloud that can contain several re-usable INPUT, MODEL and ANALYSIS objects similar to a native UQLAB session within MATLAB.

Finally, the communication between the API service and the workers is mediated by a task-scheduler, that deals with the distribution and bookkeeping of user tasks across workers, and giving appropriate responses to user queries through the API service.

¹<https://mathworks.com/products/compiler.html>

Such an infrastructure also enable full bi-directional asynchronous communication with the client. In other words, this provides support for workflows that require computational model evaluations outside the cloud (e.g. local). This is relevant, e.g., when local software is required to perform an analysis, or in the case of strict software licensing. As an example, users can perform reliability analysis on the cloud, but every limit state function evaluation can be executed on either local or dedicated hardware. This is even supported in adaptive settings, where continuous transactions between the user and UQCLOUD are needed to identify the most parsimonious solution path to the problem.

Although one can communicate with UQCLOUD directly through so-called REST API calls (structured messages in JSON format), this type of communication is recommended only for very advanced usage, e.g. to interface existing software to UQCLOUD. In general, this lack of user-friendliness is expected to deter most of the users with limited expertise in implementing such calls. Hence, as an integral part of UQCLOUD, we are also developing user-facing software bindings. Those are language-specific packages that handle the communication with UQCLOUD, while providing a near-native UQLAB experience to the end user.

The first such software binding, called UQ[py]Lab, is developed for PYTHON and will be presented next. Note that, while the software bindings are indeed language-specific, they consist only of the few hundreds lines of code needed to prepare and interpret the JSON structured messages needed by the API, a much more manageable cost w.r.t. translating and maintaining the entire UQLAB software into multiple languages.

3.2 UQ[py]Lab: UQLAB in PYTHON

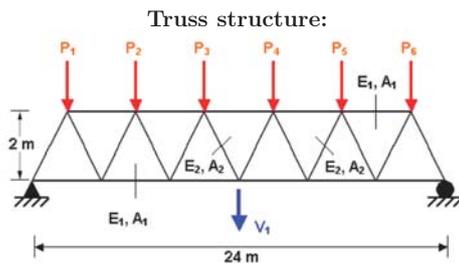
Arguably one of the more mainstream languages in data science, PYTHON was chosen as the first language for UQCLOUD bindings, resulting in the UQ[py]Lab package. Although UQ[py]Lab is still under development (beta stage at the time of writing), it already supports most of the UQLAB features².

To showcase this tool, we revisit a textbook UQ problem that was included in [5], where the UQLAB framework was initially introduced. Consider the truss structure illustrated in Figure 4. It is characterised by 10 uncertain parameters, namely the Young moduli, E_1 and E_2 , the beam sections A_1 and A_2 and the six variable loads, P_1, \dots, P_6 . The input variables are distributed according to independent lognormal or Gumbel distributions, parametrised by their first- and second-order moments and all variables are assumed statistically independent. The failure criterion (*limit state*) in this analysis corresponds to a threshold on the deflection at mid-span V_1 , which is calculated by an in-house simple finite elements model (FEM), available locally in the native PYTHON file `truss.py` (within a function called `model`).

The code that runs the reliability analysis is given in Figure 4 in two flavours. On the left side, we showcase the standard UQLAB code within MATLAB, whereas on the right side we show the equivalent UQ[py]Lab code in PYTHON. In both cases, the sequence of commands is summarised as follows:

- An INPUT object is created, based on the provided information about the distributions of the input parameters.
- A MODEL object is created by specifying that it is based on a function in a file named `truss_model.m` (resp. `truss.py`, function `model`) in MATLAB (resp. PYTHON). In both cases, this script executes the same FEM code.

²<https://www.uqlab.com/features>



Parameter distributions:

Name	Type	μ	σ/μ
E_1, E_2 (Pa)	Lognormal	2.1×10^{11}	10%
A_1 (m^2)	Lognormal	2.0×10^{-3}	10%
A_2 (m^2)	Lognormal	1.0×10^{-3}	10%
P_1, \dots, P_6 (N)	Gumbel	5.0×10^4	15%

UQLab (Matlab) code:

```

Input.Marginals(1).Name = 'E1';
Input.Marginals(1).Type = 'Lognormal'
;
Input.Marginals(1).Moments = [2.1e11,
    2.1e10];
Input.Marginals(2).Name = 'E2';
...

myInput = uq_createInput(Input)

Model.mFile = 'truss_model';

myModel = uq_createModel(Model);

Analysis.Type = 'Reliability';
Analysis.LimitState.Threshold = 0.13;
Analysis.Method = 'IS';
Analysis.MaxSamples = 1e4;

myAnalysis = uq_createAnalysis(
    Analysis)
    
```

UQ[py]Lab (Python) code:

```

Input = {
'Marginals' : [
{'Name' : 'E1',
'Type' : 'Lognormal',
'Moments' : [2.1e11, 2.1e10]},
{'Name' : 'E2',
'Type' : 'Lognormal',
'Moments' : [2.1e11, 2.1e10]},
...
]}

myInput = uq.createInput(Input)

Model = {
'Type': 'Model',
'ModelFun': 'truss_model'
}

myModel = uq.createModel(ModelOpts)

Analysis = {
'Type' : 'Reliability',
'LimitState':
{ 'Threshold': 0.13 },
'Method': 'IS',
'MaxSamples': 1e4
}

myAnalysis = uq.createAnalysis(
    Analysis)
    
```

Figure 4: Reliability analysis of a truss structure: problem representation (top left), uncertain input parameters (top right) and UQLAB/UQ[py]Lab pseudo-code to perform the analysis in MATLAB (bottom left) and PYTHON (bottom right).

- An ANALYSIS object is created to perform the reliability analysis. In this example, a maximum admissible mid-span displacement of 0.13 cm is specified, together with the reliability estimation method of choice, importance sampling. The maximum allowed cost is set to 10^4 . After executing the `uq_createAnalysis()` function in MATLAB (resp. `uq.createAnalysis()` in PYTHON), the results are stored in the workspace, inside the `myAnalysis` variable.

Next, we would like to highlight the similarity and essential equivalence between the two versions of the code. The overall simplicity and almost natural-language-based syntax of the original UQLAB is fully preserved in its python-based counterpart, UQ[Py]Lab. Of course, language specific choices need to be made, due to the different data structures available in different languages. As an example, UQLAB makes extensive use of the MATLAB native *structures*, which are not available in PYTHON, where they are substituted instead by the equivalent *dictionaries*.

Although the user experience with UQ[py]Lab is reminiscent of UQLAB, there are significant differences in the actions that take place in the background. In the UQLAB case, all operations take place locally on the machine of the user. In the UQ[py]Lab case, most of the UQ computations (creation and evaluation of INPUT, MODEL or ANALYSIS objects) take place in the cloud instead. The main exception to this is related to computational model evaluations (non surrogate-based), which are mostly executed locally. This applies to the truss example, where the FEM code is wrapped inside a PYTHON script that runs locally.

Despite the remote execution, the service latency is minimal (order of milliseconds), providing a user experience that is very close to standalone local software.

4 CONCLUSIONS AND OUTLOOK

In this work we presented a short review of the state of the UQLAB project as a whole, with a focus on its latest offspring, UQCLOUD.

To address the growing need of powerful, easy-to-learn UQ software in diverse fields of applied science and engineering, we introduce UQCLOUD, a programming language- and OS-agnostic version of UQLAB that runs on the cloud in a SaaS paradigm. We demonstrated how it can be used through minimalistic software bindings to nearly replicate the full UQLAB experience without the need of a local MATLAB installation, in PYTHON.

Shifting to a cloud-based software framework has the potential to change the way UQ is performed, as it can remove the computational burden from the client device. Among others, we plan on providing bindings in a number of different languages, including Julia, C++/C#, R, and many others. The development of dedicated online services/mobile applications based on the UQCLOUD platform, e.g. for sensitivity and reliability analysis, is also planned.

5 ACKNOWLEDGMENTS

This paper is a part of the project “Surrogate Modeling for Stochastic Simulators (SAMOS)” funded by the Swiss National Science Foundation (Grant #200021_175524), whose support is gratefully acknowledged.

REFERENCES

- [1] Baudin, M., Dutfoy, A., Iooss, B. and Popelin, A.-L. 2015 , Open turns: An industrial software for uncertainty quantification in simulation, *arXiv preprint arXiv:1501.05242* .

-
- [2] De Rocquigny, E., Devictor, N. and Tarantola, S., eds 2008 , *Uncertainty in industrial practice – A guide to quantitative uncertainty management*, John Wiley & Sons.
- [3] Eldred, M. S., Giunta, A. A., van Bloemen Waanders, B. G., Wojtkiewicz, S. F., Hart, W. E. and Alleva, M. P. 2006 , DAKOTA, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis, Technical report, Citeseer.
- [4] Feinberg, J. and Langtangen, H. P. 2015 , Chaospy: An open source tool for designing methods of uncertainty quantification, *Journal of Computational Science* **11**, 46–57.
- [5] Marelli, S. and Sudret, B. 2014 , UQLab: A framework for uncertainty quantification in Matlab, *in* Vulnerability, Uncertainty, and Risk (Proc. 2nd Int. Conf. on Vulnerability, Risk Analysis and Management (ICVRAM2014), Liverpool, United Kingdom), American Society of Civil Engineers, pp. 2554–2563.
- [6] Marelli, S., Wicaksono, D. and Sudret, B. 2019, The UQLab project: steps toward a global uncertainty quantification community, *Proc. 13th Int. Conf. on Applications of Stat. and Prob. in Civil Engineering (ICASPI3)*, Seoul, South Korea.
- [7] Merkel, D. 2014 , Docker: lightweight linux containers for consistent development and deployment, *Linux journal* **2014**(239), 2.
- [8] National Research Council 2012 , *Assessing the reliability of complex models: mathematical and statistical foundations of verification, validation, and uncertainty quantification*, National Academies Press.
- [9] Olivier, A., Giovanis, D., Aakash, B., Chauhan, M., Vandanapu, L. and Shields, M. D. 2020 , UQpy: A general purpose python package and development environment for uncertainty quantification, *Journal of Computational Science* **47**, 101204.
- [10] Patelli, E., Broggi, M., Angelis, M. d. and Beer, M. 2014 , Opencossan: An efficient open tool for dealing with epistemic and aleatory uncertainties, *in* Vulnerability, Uncertainty, and Risk: Quantification, Mitigation, and Management, pp. 2564–2573.
- [11] Richardson, L., Amundsen, M. and Ruby, S. 2013 , *RESTful Web APIs*, O’Reilly Media, Inc.
- [12] Sudret, B. 2007 , *Uncertainty propagation and sensitivity analysis in mechanical models – Contributions to structural reliability and stochastic spectral methods*, Université Blaise Pascal, Clermont-Ferrand, France. Habilitation à diriger des recherches.

OTBENCHMARK: AN OPEN SOURCE PYTHON PACKAGE FOR BENCHMARKING AND VALIDATING UNCERTAINTY QUANTIFICATION ALGORITHMS

Elias Fekhari¹, Michaël Baudin¹, Vincent Chabridon¹, Youssef Jebroun¹

¹EDF R&D, 6 Quai Watier, 78400 Chatou
e-mail: {elias.fekhari, michael.baudin, vincent.chabridon}@edf.fr

Keywords: Benchmark, Uncertainty Quantification, Reliability Analysis, Sensitivity Analysis, Python, OpenTURNS.

Abstract.

Over the past decade, industrial companies and academic institutions pooled their efforts and knowledge to propose a generic uncertainty management methodology for computer simulation. This framework led to the collaborative development of an open source software dedicated to the treatment of uncertainties, called “OpenTURNS” (Open source Treatment of Uncertainty, Risk’N Statistics). This paper aims at presenting a new Python package, called “otbenchmark”, offering tools to evaluate the performance of a large panel of uncertainty quantification algorithms. It provides benchmark classes containing problems with their reference values. Two categories of benchmark classes are currently available: reliability estimation problems (i.e., estimating failure probabilities) and sensitivity analysis problems (i.e., estimating sensitivity indices such as the Sobol’ indices). This package can either be used for validating a new algorithm or automatically comparing various algorithms on a set of problems. Additionally, the package provides several convergence and accuracy metrics to compare the performance of each algorithm. To face high-dimensional problems, otbenchmark offers graphical tools to draw multidimensional events, functions and distributions based on cross-cuts visualizations. Finally, to ensure otbenchmark’s accuracy, a test-driven software development method has been adopted (using, among others, Git for collaborative development, unit tests and continuous integration). Ultimately, otbenchmark is an industrial platform gathering problems with reference values of their solutions and various tools to achieve a robust comparison of uncertainty management algorithms.

1 INTRODUCTION

Complex computer simulation often requires implementing uncertainty management methods to evaluate associated risks, robustness and design margins. Several industrial companies and academic institutions pooled their efforts and knowledge to propose a generic uncertainty management methodology for computer simulation. This framework led to the collaborative development of an open source software dedicated to the treatment of uncertainties, called “OpenTURNS” (Open source Treatment of Uncertainty, Risk’N Statistics) [3]. Initially created by EDF R&D, Airbus Group and Phimeca Engineering, later joined by IMACS and ONERA, OpenTURNS is a generic, modular, transparent and multi-accessibility industrial software dedicated to serve several purposes (e.g., uncertainty quantification, uncertainty propagation, surrogate modeling, reliability, sensitivity analysis and calibration).

In the vein of a first benchmark challenge organized in 2019 (the “Black-box Reliability Challenge” [12]), this paper aims at presenting a new Python module, called “`otbenchmark`”¹, which aims at providing several automatic tools to evaluate the performance of a large panel of uncertainty quantification algorithms by relying on the probabilistic programming framework offered by OpenTURNS. In other words, this module provides benchmark classes for OpenTURNS. It sets up a framework to create use-cases or problems associated with reference values. Such a benchmark problem may be used in order to check that a given algorithm works as expected and to measure its performance in terms of speed and accuracy.

Two categories of benchmark classes are currently provided: the first one is devoted to reliability estimation problems (i.e., estimating failure probabilities), the second one is devoted to sensitivity analysis problems (i.e., estimating sensitivity indices such as the Sobol’ indices). `otbenchmark` is currently composed of 26 reliability problems and 4 sensitivity problems. For all these problems, reference solutions are provided. These solutions are obtained, either from a crude Monte Carlo estimation with a controlled convergence, or using (when possible) an exact resolution (e.g., provided by algebraic operations on input distributions). Additionally, `otbenchmark` provides several convergence and accuracy metrics to compare the performance of each algorithm. Finally, in order to perform a complete benchmark, a loop can be automatically set to evaluate a large panel of algorithms over the complete set of examples.

Graphical representations are often useful to help the analyst to understand the underlying behavior of complex reliability or sensitivity problems. Since many of these problems have dimensions larger than two, it raises numerous practical issues. `otbenchmark` offers graphical tools to draw multidimensional events, functions and distributions based on cross-cuts. Finally, to ensure `otbenchmark`’s accuracy, a test-driven software development method was followed (using, among others, Git and github.com for collaborative development, unit tests and continuous integration).

Thus, `otbenchmark` is an industrial benchmark platform for uncertainty management algorithms, and can be seen as a versatile tool offering diverse problems with corresponding solutions, robust metrics and graphical representations for high-dimensional problems.

In this paper, section 2 gives a formulation reminder for reliability and sensitivity problems, presents the probabilistic programming framework of OpenTURNS and demonstrates the added value of `otbenchmark` over the existing benchmark repositories. Section 3 defines the package’s object-oriented architecture by detailing its main classes. Section 4 introduces the benchmark problems and the research for the most accurate associated reference values. Finally, section 5 is an illustrative example of the results automatically produced by the package

¹Official repository: <https://github.com/mbaudin47/otbenchmark>

for a range of problems and algorithms.

2 MOTIVATIONS AND OBJECTIVES

2.1 Reliability analysis and sensitivity analysis formulations

Formally, the general scope of this paper is to address typical problems defined by considering a model given by $g : \mathbb{R}^d \rightarrow \mathbb{R}^p$, for which one considers a set of d uncertain inputs X_i ($i \in \{1, \dots, d\}$) gathered in a random vector $\mathbf{X} \in \mathbb{R}^d$. This random vector is associated to its joint probability distribution denoted in the following by the joint probability density function (PDF) f_X , built from marginal densities and a copula. Propagating the uncertainties can be achieved through the following relationship:

$$Y = g(\mathbf{X}) \quad (1)$$

where the model output Y is a random variable (either univariate or multivariate). In the following, one will only assume a scalar output variable for the sake of clarity (note that OpenTURNS is not limited to scalar outputs since most of its classes are designed to naturally handle multivariate outputs). Based on this formulation, several types of analyses, associated to various *quantities of interests* can be solved. Among others, one will focus in this paper only on the following two types of analyses:

- Reliability analysis: in this case, one desires to compute a *failure probability*, denoted by p_f and defined through a *threshold event* E which characterizes the failure. This quantity of interest simply reads:

$$p_f = \int_{\mathbb{R}^d} \mathbb{1}_E(\mathbf{x}) f_X(\mathbf{x}) d\mathbf{x}. \quad (2)$$

Many standard or advanced algorithms can be used such as, typically, simulation-based ones (e.g., Monte Carlo sampling, importance sampling, subset sampling), approximation-based methods (FORM/SORM) or hybrid algorithms (e.g., FORM coupled with importance sampling, surrogate-model based strategies).

- Sensitivity analysis: in such as case, one desires to compute a *sensitivity index* (or a set of indices) which reflects the way an input (or a set of inputs) influence the variability of an output quantity of interest. For instance, by considering the variance of Y , one can compute the well-known Sobol' indices [14] as follows:

$$S_i = \frac{\text{Var} [\mathbb{E}[Y|X_i]]}{\text{Var}[Y]}, \quad S_{T_i} = \frac{\mathbb{E} [\text{Var}[Y|\mathbf{X}_{-i}]]}{\text{Var}[Y]}, \quad (3)$$

where S_i is the first-order index, S_{T_i} the total-order index of the variable X_i and \mathbf{X}_{-i} stands for \mathbf{X} without the i -th component. Several algorithms can be used to estimate these indices (e.g., sampling-based, surrogate-based or given-data algorithms) [8].

Other types of analyses (e.g., estimating the mean and variance of the model output, calibration, surrogate model fitting) will be further considered in future work.

These two types of analyses can be performed by using dedicated algorithms proposed in various open source or commercial software. OpenTURNS is one of them and provides several classes and algorithms to do so. In the next paragraph, one specifically focuses on a few core elements of the library and briefly explains why OpenTURNS is particularly well-dedicated to solve efficiently the problems presented hereabove.

2.2 OpenTURNS as a tool for uncertainty quantification

Originally founded by an industrial partnership between EDF R&D, Airbus and Phimeca Engineering in 2005, IMACS joined the OpenTURNS partnership in 2014, followed by ONERA in 2019. As an open source software developed under the LGPL license, OpenTURNS offers several features: if the core is written in C++, the application programming interface is in Python which makes it usable as a standard Python library. The architecture of the code is fully object-oriented and provide a large panel of classes and methods which are supported by a rigorous continuous integration process which provides a high-quality code for both industrial studies and research projects. The tool, in addition to its development and maintenance, is intensively used to support several industrial uncertainty quantification studies of the various partners.

OpenTURNS provides tools for what is usually called “probabilistic programming”, a programming paradigm which facilitates the combination of probabilistic objects for statistical modeling. Among the OpenTURNS probabilistic object used in `otbenchmark`, the most iconic ones must be introduced for a better understanding.

The `Distribution` class defines the probability distribution function of a random variable. It provides more than one hundred methods, including `getMean`, `getStandardDeviation`, `computePDF`, `computeQuantile`, `drawPDF`, `getSample`, `computeConditionalCDF`, etc. Dealing with a set of random variables (a.k.a marginals) intertwined with a dependency model (e.g., variance-covariance matrix or copulas) a.k.a a random vector, can easily be modeled using OpenTURNS.

The `SymbolicFunction` is an efficient tool one can use when an analytical expression of the function is known. The additional advantage of this class is evaluating the gradient and hessian when mathematically defined.

The `ThresholdEvent` class defines the event which probability is to be estimated. It is based on a `RandomVector`, an operator and a threshold. The event occurs when the realization of the random vector exceeds the threshold.

Together with these fundamental tools, OpenTURNS contains several algorithms designed for efficiently solving a large variety of problems as the ones given in Eq. (2) and Eq. (3). However, such an environment would benefit from a dedicated benchmark platform to assess the performance of both the existing algorithms in the library together with any new proposed method which needs to be tested.

2.3 From current benchmark repositories to the `otbenchmark` package

According to Oxford English dictionary, a “benchmark” is “something that can be measured and used as a standard that other things can be compared with”. As shown in this section, there is a long history of benchmark problems in various fields of applied mathematics (e.g., numerical methods, statistical software, optimization, design of computer experiments and uncertainty quantification).

Several fields in applied mathematics (e.g., linear algebra, numerical analysis) and computer simulation benefited from the development of test problems libraries (see, e.g., [17, 7, 6]). Uncertainty quantification naturally benefited from all the tools developed in those fields. However, uncertainty quantification problems have something specific that they can involve both of the previous fields (i.e., statistical inference, numerical integration, optimization). In the recent years, much effort has been put in the development of open source or commercial efficient uncertainty quantification platforms. However, benchmarking tools have not been deeply ex-

plored. Well-known repositories of benchmark problems are the Virtual Library of Simulation Experiments² managed by S. Surjanovic and D. Bingham [15]. [15], or the one maintained by the GDR Mascot-Num³ research group. However, as a remark, one can mention that these two repositories mainly propose test functions in Matlab and R (not Python), but without aiming at providing an automated benchmark framework.

In 2018, concluding a workshop⁴ organized by the Department of Structural Reliability at TNO (Netherlands Organization for Applied Scientific Research) about the computational challenges and aspects in the reliability analysis of engineering structures, several members of the community decided to create the first “Black-box Reliability Challenge”, whose first edition happened in 2019 [12]. For this challenge, almost thirty benchmark problems have been provided to participants on a public repository⁵. More details about this challenge and the results can be found in the websites of the second workshop in 2020⁶. Thus, by providing a set of reliability problems in open source (available in their original form here⁷), this first initiative offered a great opportunity to start with the project of a dedicated benchmark tool: namely the proposed `otbenchmark` package.

2.4 Objectives of the proposed tool

The objective of the proposed package is twofold: firstly, the idea is to provide a benchmark tool for any potential external user (either an OpenTURNS user or anyone interested in performing a benchmark); secondly, to provide to the OpenTURNS development team a tool helping the implementation of new algorithms.

From the external user point of view, one can imagine mostly two possible scenarios:

- (Scenario #1) A user would like to design, implement and test a new algorithm which is not proposed in the library yet. Then, `otbenchmark` would provide a range of test problems and reference values so as to compare the performance of the new algorithm with respect to reference results (or other existing algorithms);
- (Scenario #2) A user would like to apply, all (or part of) the algorithms available in the library on a given user-defined problem (e.g., either an analytical or a real industrial black-box problem).

More generally, in the context of a real industrial problem (e.g., typically a reliability or sensitivity analysis of a complex, potentially costly-to-evaluate, simulation model), a user could be interested in using this benchmark module as a catalogue of test functions displaying various features (e.g., input dimension, independence/dependence of the inputs, distribution types, rareness of the failure probability) which could help him/her to test, choose and validate a choice of several “candidate algorithms”.

In the next section, the core architecture of the `otbenchmark` package is presented.

²<http://www.sfu.ca/ssurjano/index.html>

³<https://www.gdr-mascotnum.fr/benchmarks.html>

⁴<https://www.reliabilitytno.com/>

⁵<https://rprepo.readthedocs.io/en/latest/index.html>

⁶<https://reliabilityworkshop2020.com/>

⁷<https://gitlab.com/rozsasarp/rprepo/>

3 ARCHITECTURE OF THE PACKAGE

3.1 Classes for reliability problems

In the sequel, `ot` denotes the short name for the OpenTURNS platform and associated classes. The basic class for reliability problems is the `ReliabilityBenchmarkProblem`, which defines a generic reliability problem. This class defines three constructor parameters:

- `name`: a string representing the name of the benchmark problem. This is a short string, typically less than a dozen of characters;
- `thresholdEvent`: a `ot.ThresholdEvent` object representing the event to estimate;
- `probability`: a float which represents the exact probability.

The essential information is the reference probability $p_{f,\text{ref}}$, which should be as accurate as possible. The best possible accuracy for a Python `float` is 53 significant (binary) bits, which approximately corresponds to 15 (up to 17) decimal digits. If this accuracy is not available, then a reference value may be used, for example, obtained from a large Monte Carlo sample. In general, the exact probability should be a constant value, e.g., 0.123456789. However, we may be forced to compute this probability at the creation of the problem, for example if the threshold of the problem can be set at the creation of the object. In this case, the unit test must check that the default value of the parameters correspond to a reference constant value.

The `ReliabilityBenchmarkProblem` provides several methods, including `getEvent`, `getProbability` and `getName`, which returns the corresponding attributes. Moreover, other methods are provided, including pretty printing services. More importantly, the `computeBeta` method, which computes the Hasofer-Lind reliability index β_{HL} using the relationship $p_f = \Phi(-\beta_{\text{HL}})$ [10] based on the reference probability value.

In order to implement a specific reliability problem, a derived class is defined from the `ReliabilityBenchmarkProblem` mother class. For example, one can mention the `ReliabilityProblem54` derived from the `ReliabilityBenchmarkProblem` class to implement the so-called “RP54” problem. The practical implementation involves the definition of the input distribution of \mathbf{X} , the model function $g(\cdot)$ and the threshold value s . All these elements are defined within an instance of the `ThresholdEvent` class.

Specific reliability problems have specific attributes which can be provided in the constructor of the problem. For example, the `ReliabilityProblem28` class provides, as an optional extra, an attribute to set the mean and standard deviations of the input Gaussian distributions. The default values of these parameters are the ones which originally defines the reliability problem, but the user may want to modify these default values, for example to make the problem easier or more difficult or easier. Thus, the user can tune the problem regarding his specific needs, but has the drawback to require to update the reference probability $p_{f,\text{ref}}$ depending on the actual parameters values: this cannot be done with guaranteed accuracy, but for a very limited number of problems.

The following piece of code provides a step-by-step illustration of how to create a problem, run an algorithm and compare the computed probability with a reference probability $p_{f,\text{ref}}$. For the `RminusSReliability` (Resistance-Sollicitation) benchmark problem, the `ProbabilitySimulationAlgorithm` class from OpenTURNS is used to estimate the probability based on a sequential Monte Carlo algorithm. After running the algorithm, the probability is estimated with the `getProbabilityEstimate` method and the absolute error computed

using the reference probability provided by the `getProbability` of the benchmark problem.

```

1 # Import the packages
2 import openturns as ot
3 import otbenchmark as otb
4 # Select a problem, get the associated event and the reference probability
5 problem = otb.RminusSReliability()
6 event = problem.getEvent()
7 pfReference = problem.getProbability()
8 # Create a Monte Carlo algorithm and set its stopping criteria
9 algoProb = ot.ProbabilitySimulationAlgorithm(event)
10 algoProb.setMaximumOuterSampling(1000)
11 algoProb.setMaximumCoefficientOfVariation(0.01)
12 # Run the algorithm
13 algoProb.run()
14 # Get the result and compare it to the reference
15 resultAlgo = algoProb.getResult()
16 pf = resultAlgo.getProbabilityEstimate()
17 absoluteError = abs(pf - pfReference)

```

3.2 Classes for sensitivity analysis problems

The `SensitivityBenchmarkProblem` class defines a generic sensitivity analysis problem, depending on the following constructor parameters:

- `name`: a string representing the name of the benchmark problem. This is a short string, typically less than a dozen of characters;
- `distribution`: a `ot.Distribution` which represents the input distribution of the random vector \mathbf{X} ;
- `function`: a `ot.Function` to define the model $g(\cdot)$;
- `firstOrderIndices`: a `ot.Point` representing the exact first-order Sobol' indices;
- `totalOrderIndices`: a `ot.Point` representing the exact total-order Sobol' indices.

The corresponding `get` methods provides a way to get the current values of the parameters, e.g., `getFirstOrderIndices` and `getTotalOrderIndices`.

In order to implement a specific sensitivity analysis problem, a derived class of the `SensitivityBenchmarkProblem` mother class is provided. This requires to know the reference first-order and total-order Sobol' indices for the problem of interest. For some sensitivity analysis problems, the user is given a leeway to customize more specific parameters. As an example, one can mention the G-Sobol' test function, for which the `GSobolSensitivity` class provides the optional parameter `a` which, by default, contains an array of three floating point numbers (equal respectively to 0, 9 and 99). In this case, the reference first- and total-order Sobol' indices must be updated according to the actual values of these tuning parameters, which is easy for the G-Sobol' test function, but might be more difficult or impossible for some other problems.

3.3 Classes to manage the results of a benchmark

When a benchmark problem is run, it is interesting to compare the results obtained by various algorithms on this problem. To do so, one needs to define the set of features that one wants to assess.

The `ReliabilityBenchmarkResult` class is used to define the output of a reliability benchmark problem. Its constructor parameters are:

- `exactProbability`: a floating point number representing the exact probability;
- `computedProbability`: a floating point number representing the estimated probability;
- `numberOfFunctionEvaluations`: an integer representing the number of function evaluations.

Based on these parameters, the class computes several attributes:

- `absoluteError`: a floating point number representing the absolute error of the estimated probability;
- `numberOfCorrectDigits`: a floating point number representing the log-relative error in base 10;
- `numberOfDigitsPerEvaluation`: a floating point number representing the number of correct digits per function evaluation.

This last attribute measures the efficiency of the algorithm in computing the significant digits of the probability (typically, the larger the better).

3.4 Classes to perform a benchmark

The `ReliabilityBenchmarkProblemList` static method returns a list of benchmark problems available in the library. In the following example, one gets the list of problems and compute its length:

```
14 benchmarkProblemList = otb.ReliabilityBenchmarkProblemList()
15 numberOfProblems = len(benchmarkProblemList)
```

The number of problems is currently equal to 26. Then the following script performs a loop over the problems and prints the name, the reference probability $p_{f,ref}$ and the dimension of each problem:

```
16 for i in range(numberOfProblems):
17     problem = benchmarkProblemList[i]
18     name = problem.getName()
19     pfReference = problem.getProbability()
20     event = problem.getEvent()
21     antecedent = event.getAntecedent()
22     distribution = antecedent.getDistribution()
23     dimension = distribution.getDimension()
24     print("#", i, ":", name, " : pfReference = ", pfReference, ",
25         dimension=", dimension)
```

As a result, the previous script prints:

Distribution	Symbol	Parameters	PDF
Uniform	\mathcal{U}	(a, b)	$f_X(x) = \begin{cases} \frac{1}{b-a} & , x \in [a, b] \\ 0 & , x \notin [a, b] \end{cases}$
Normal	\mathcal{N}	(μ, σ)	$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$
Log-normal	\mathcal{LN}	(μ, σ)	$f_X(x) = \frac{1}{x} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(\ln(x)-\mu)^2}{2\sigma^2}\right)$
Exponential	\mathcal{E}	λ	$f_X(x) = \lambda e^{-\lambda x}$
Gumbel	\mathcal{G}	(μ, β)	$f_X(x) = \frac{1}{\beta} \exp\left(-\frac{x-\mu}{\beta} - \exp\left(-\frac{x-\mu}{\beta}\right)\right)$

Table 1: Probability distribution parametrization used in `otbenchmark`.

```
# 0 : RP8 : pfReference = 0.000784 , dimension = 6
# 1 : RP14 : pfReference = 0.00752 , dimension = 5
# 2 : RP22 : pfReference = 0.00416 , dimension = 2
[...]
```

4 DESCRIPTION OF THE BENCHMARK PROBLEMS

4.1 Input random variable parametrization

The first element of a benchmark problem is the input random vector $\mathbf{X} = (X_1, \dots, X_d) \in \mathbb{R}^d$. Although it might look obvious, precisising the way random variables are defined is crucial in the context of a benchmark, especially since several distributions can be parametrized differently (e.g., the log-normal distribution). Table 1 provides the parametrization used in `otbenchmark`. All together, `OpenTURNS` provides 36 basic probability distributions, which allow the user to build an infinite number of distributions by truncating, transforming and combining them. Moreover, a large panel of copulas can be used to model input dependency.

4.2 Reliability problems description

As described in the previous sections, a reliability problem is defined by an input random vector X , a function $g(\cdot)$, a threshold s and the corresponding reference failure probability $p_{f,\text{ref}}$. One of the major challenge of a reliability benchmark is getting the most accurate reference failure probability. Without such an accurate reference solution, the benchmark is worthless. Depending on the complexity of the function, the distribution of the input random vector and the rareness of the failure event, computing $p_{f,\text{ref}}$ can be more or less challenging. The safest way to achieve this regardless of the previous parameters is using the a very large crude Monte Carlo simulation, however, this method is very costly.

For some specific cases, one can compute an exact solution using quadrature methods. These techniques are extremely powerful since they allows to exactly compute the full output distribution Y without sampling the function. Overall, among the reference probabilities provided in `otbenchmark`, some are estimated with a huge Monte Carlo sampling, some are borrowed from [12] (and the associated references from the literature), while others are computed exactly using quadrature methods on the input distributions.

The implementation of the exact quadrature computation is problem-dependent and should progressively be applied to as many problems as possible. These reference values are continuously improved and offer interesting work perspectives. Table 2 presents the current reference values for $p_{f,\text{ref}}$ available in `otbenchmark` and the corresponding values of the Hasofer-Lind

reliability index β_{HL} . Note that the values tagged with an asterisk (*) differ from the initial reference probabilities provided by [12].

Table 2: Reliability problems definition.

Problem label	d	$p_{f,ref}$	β_{HL}
RP8	6	7.84e-04	3.16
RP14	5	7.52e-03	2.42
RP22	2	4.16e-03	2.64
RP24	2	2.86e-03	2.76
RP25	2	6.14e-06	4.36
RP28	2	1.46e-07	5.11
RP31	2	1.8e-04	3.58
RP33	3	2.57e-03	2.8
RP35	2	3.54e-03	2.7
RP38	7	8.1e-03	2.48
RP53	2	3.13e-03	1.86
RP54	20	9.98e-04	3.09
RP55	2	5.6e-01*	-0.15
RP57	2	2.84e-02	1.91
RP60	5	4.56e-02	1.7
RP63	100	3.79e-04	3.36
RP75	2	1.07e-02	2.33
RP77	3	2.87e-07	5.0
RP89	2	5.43e-03	2.55
RP91	5	6.97e-04	3.19
RP107	10	2.92e-07	5.0
RP110	2	3.19e-05	4.0
RP111	2	7.65e-07	4.81
R-S	2	7.86e-02	1.41
Axial stressed beam	2	2.92e-02	1.89
Four-branch serial system	2	2.19e-03	2.85

4.3 Sensitivity problems description

The `otbenchmark` package currently contains 4 sensitivity analysis problems. More precisely, it includes the sum of several Gaussian random input variables, the product of several Gaussian random input variables, the G-Sobol' function [13] and the Ishigami function [9]. The first two test functions have variable dimensions. More sensitivity analysis problems will be added in the next release.

5 BENCHMARK RESULTS

This section presents how `otbenchmark` allows to compute and compare various metrics on multiple problems with several algorithms. At first, a simple “for” loop is performed to

solve each problem with a list of algorithms (e.g., Monte Carlo, FORM, SORM, FORM-IS and Subset) and compute a list of metrics (e.g., failure probability, number of correct digits, absolute error). This loop uses the same structure as the one described in subsection 3.4 and various metrics can be directly computed using the methods described in the subsection 3.3.

In the following, a maximum simulation budget is set to $n_{\max} = 10^4$ calls. Such a value is set for illustration purposes here, even if this budget is clearly not enough to reach low failure probabilities proposed by some problems. The Monte Carlo algorithm stopping criterion used is this maximum number of function evaluations (i.e., n_{\max}). FORM and SORM methods are used with the Abdo-Rackwitz algorithm for the search of the design point [10, 1]. Regarding SORM, the Breitung approximation formula is used [4]. For the two approximation methods, the maximum number of function evaluations is set to n_{\max} , the maximum absolute error, maximum relative error, maximum residual error and maximum constraint error are set to 10^{-3} . The FORM-IS algorithm [10] first uses the FORM analysis to find the design point and then uses importance sampling with a Gaussian importance distribution in the standard space, centered on the design point. The FORM-IS and Subset algorithms [2] are used with the same stopping criterion as the Monte Carlo algorithm.

Overall, the results are in three dimensions and summarize both the problems, the algorithms and the metrics. To handle such a complex data structure, the multi-columns `DataFrame` class offered by the `pandas` package [16] is used. In addition to being a reference for data manipulation, `pandas` is known to well perform with multi-columns, provides powerful styling options and allows one to export any `DataFrame` to a \LaTeX table. Table 3 is the result of an automated `pandas` export of the failure probabilities. It is obvious here that, with such a limited budget (i.e., n_{\max}), some problems are too difficult for some algorithms which fail to converge towards the reference failure probabilities. This explains the numerous zero values in Table 3 (or with an hyphen symbol when one does not want to run the Monte Carlo algorithm since one already knows it will not converge). Note that the purpose here is not to solve the problems but discuss the way `otbenchmark` works, produces, displays and compares the results.

This short illustration was performed on five algorithms but could easily be extended to many more reliability analysis algorithms available in `OpenTURNS`, including the adaptive directional stratification [11], the multiple design points strategy adapted to FORM/SORM [5] or FORM-system algorithm [10]. Adaptive surrogate-based algorithms will also be added in future work.

6 CONCLUSION

The `otbenchmark` package offers a open source benchmark tool for any user interested in performing reliability or sensitivity analysis, but more generally, to much more analyses usually encountered in uncertainty quantification. This versatile tool can serve many objectives such as helping in the development, testing and validation of new algorithms, or applying several algorithms to a given problem. It mainly relies on powerful classes and methods inherited from the `OpenTURNS` library, but also propose several new classes and dedicated tools specifically derived for benchmarking purposes. It gives access to a collection of problems with their reference solutions and allows to compare the performances of algorithms. Moreover, various tools to make this comparison more automated, robust and visual are available. Several convergence and accuracy metrics are provided to compare the algorithms performances, graphical tools and result tables are proposed to ease the results analysis. The quality of the reference values and, more generally, of the software are of paramount importance to ensure a consistent benchmark.

Table 3: Estimation of p_f for all the problems using 5 algorithms ($n_{\max} = 10^4$ calls).

	$p_{f,\text{ref}}$	Monte Carlo	FORM	SORM	FORM-IS	Subset
RP8	7.840e-04	9.000e-04	6.599e-04	7.837e-04	7.737e-04	8.863e-04
RP14	7.520e-03	9.000e-04	7.003e-04	6.988e-04	7.598e-04	8.720e-04
RP22	4.160e-03	3.500e-03	6.210e-03	4.391e-03	4.259e-03	4.117e-03
RP24	2.860e-03	3.600e-03	6.209e-03	6.209e-03	2.749e-03	2.486e-03
RP25	6.140e-06	1.000e-04	2.105e-03	1.064e-05	4.644e-05	3.415e-05
RP28	1.460e-07	–	2.850e-08	0.000e+00	1.332e-07	1.756e-07
RP31	1.800e-04	2.300e-03	2.275e-02	2.275e-02	3.319e-03	3.919e-03
RP33	2.570e-03	1.600e-03	1.350e-03	1.350e-03	2.322e-03	2.718e-03
RP35	3.540e-03	3.000e-03	1.350e-03	2.134e-03	2.377e-03	3.430e-03
RP38	8.100e-03	8.500e-03	7.902e-03	8.029e-03	8.146e-03	7.848e-03
RP53	3.130e-02	3.260e-02	1.180e-01	2.986e-02	3.143e-02	2.971e-02
RP55	5.600e-01	5.660e-01	5.000e-01	1.093e-05	5.645e-01	5.655e-01
RP54	9.980e-04	1.100e-03	5.553e-02	3.552e-03	9.767e-04	9.611e-04
RP57	2.840e-02	2.950e-02	4.504e-01	0.000e+00	2.746e-02	2.772e-02
RP75	1.070e-02	1.030e-02	0.000e+00	0.000e+00	0.000e+00	9.409e-03
RP89	5.430e-03	5.000e-03	2.009e-09	2.009e-09	9.002e-05	5.460e-03
RP107	2.920e-07	–	2.867e-07	2.867e-07	2.896e-07	2.337e-07
RP110	3.190e-05	–	3.167e-05	3.167e-05	3.078e-05	7.116e-06
RP111	7.650e-07	–	0.000e+00	0.000e+00	0.000e+00	7.308e-07
RP63	3.790e-04	1.000e-04	1.000e+00	0.000e+00	0.000e+00	4.063e-04
RP91	6.970e-04	1.000e-03	6.984e-04	7.001e-04	6.964e-04	6.838e-04
RP60	4.560e-02	4.860e-02	4.484e-02	4.484e-02	4.503e-02	4.230e-02
RP77	2.870e-07	–	6.687e-02	6.687e-02	4.002e-07	3.683e-07
Four-branch serial system	2.186e-03	2.900e-03	0.000e+00	0.000e+00	0.000e+00	2.428e-03
R-S	7.865e-02	7.870e-02	7.865e-02	7.865e-02	7.792e-02	7.633e-02
Axial stressed beam	2.920e-02	2.690e-02	2.998e-02	2.933e-02	2.867e-02	2.936e-02

This is managed first by providing reference values as accurate as possible (which may involve, for example and when possible, exact quadrature calculations). Furthermore, this quality expectation is made achievable using software development methods which includes source version control (using Git), unit tests, continuous integration and collaborative development. As part of the `otbenchmark` development roadmap, one can mention the following prospects: some of the reference values will be updated using, as much as possible, exact quadrature methods to get the largest possible number of significant digits for reference values; sensitivity analysis will be extended to new problems and more investigated; other types of analyses will be also investigated (e.g., central tendency, calibration).

ACKNOWLEDGMENTS

The Authors are grateful to Régis Lebrun (Airbus) for his contributions to the package and for providing exact reference values for several benchmark problems and to Bertrand Iooss (EDF R&D) for his help in proofreading the paper.

REFERENCES

- [1] T. Abdo and R. Rackwitz. *A New Beta-Point Algorithm for Large Time-Invariant and Time-Variant Reliability Problems*. 3rd IFIP Working Conf. 1990.
- [2] S.-K. Au and J. L. Beck. “Estimation of small failure probabilities in high dimensions by subset simulation”. In: *Probabilistic engineering mechanics* 16.4 (2001), pp. 263–277.
- [3] M. Baudin et al. “OpenTURNS: An Industrial Software for Uncertainty Quantification in Simulation”. In: *Handbook of Uncertainty Quantification*. Ed. by R. Ghanem, D. Higdon, and H. Owhadi. Cham: Springer International Publishing, 2017, pp. 2001–2038.
- [4] K. Breitung. “Asymptotic approximations for multinormal integrals”. In: *Journal of Engineering Mechanics* 110.3 (1984), pp. 357–366.
- [5] A. Der Kiureghian and T. Dakessian. “Multiple design points in first and second-order reliability”. In: *Structural Safety* 20.1 (1998), pp. 37–49.
- [6] A. Genz. “Testing multidimensional integration routines”. In: *Proc. of international conference on Tools, methods and languages for scientific and engineering computation*. 1984, pp. 81–94.
- [7] N. Higham. “Algorithm 694 A Collection of Test”. In: *ACM Transactions on Mathematical Software* 17.3 (1991).
- [8] B. Iooss and P. Lemaître. “A Review on Global Sensitivity Analysis Methods”. In: *Uncertainty Management in Simulation-Optimization of Complex Systems: Algorithms and Applications*. Ed. by G. Dellino and C. Meloni. Boston, MA: Springer US, 2015. Chap. 5, pp. 101–122.
- [9] T. Ishigami and T. Homma. “An importance quantification technique in uncertainty analysis for computer models”. In: *[1990] Proceedings. First International Symposium on Uncertainty Modeling and Analysis*. IEEE. 1990, pp. 398–403.
- [10] M. Lemaire, A. Chateaneuf, and J.-C. Mitteau. *Structural Reliability*. ISTE Ltd. - Wiley, 2009. ISBN: 978-1-848-21082-0.
- [11] M. Munoz Zuniga et al. “Adaptive directional stratification for controlled estimation of the probability of a rare event”. In: *Reliability Engineering & System Safety* 96.12 (2011), pp. 1691–1712.
- [12] A. Rozsas and A. Slobbe. *Repository and Black-box Reliability Challenge 2019*. <https://gitlab.com/rozsasarp/rprepo/>. 2019.

- [13] A. Saltelli and I. M. Sobol'. "Sensitivity analysis for nonlinear mathematical models: numerical experience". In: *Matematicheskoe Modelirovanie* 7.11 (1995), pp. 16–28.
- [14] I. M. Sobol. "Sensitivity estimates for nonlinear mathematical models". In: *Mathematical Modelling and Computational Experiments* 1 (1993), pp. 407–414.
- [15] S. Surjanovic and D. Bingham. *Virtual Library of Simulation Experiments: Test Functions and Datasets*. <http://www.sfu.ca/~ssurjano>. 2013.
- [16] The pandas development team. *pandas-dev/pandas: Pandas*. Version latest. Feb. 2020. DOI: 10.5281/zenodo.3509134. URL: <https://doi.org/10.5281/zenodo.3509134>.
- [17] "The LINPACK 1000x1000 benchmark program." <http://www.netlib.org/benchmark/>.

ANALYTICAL MODEL FOR FRACTURE IN RANDOM QUASIBRITTLE MEDIA BASED ON EXTREMES OF THE AVERAGING PROCESS

Miroslav Vořechovský¹

¹Brno University of Technology
Veveří 95, Brno, 602 00, Czech Republic
e-mail: vorechovsky.m@vut.cz

Keywords: quasibrittle structure, concrete, discrete model, fracture process zone, random strength field, local averaging, weakest-link model.

Abstract. *The paper presents an analytical model for prediction of the peak force in concrete specimens loaded in bending (both notched and unnotched). The model is capable of predicting peak force statistics by computing the extreme values of sliding averages of random strength fields. The local strength of the specimen is modeled by a stationary isotropic random field with Gaussian distribution and a given autocorrelation function. The averaging operation represents the progressive loss in material integrity and the associated stress redistribution that takes place prior to reaching the peak load. Once the (linear) averaging process is performed analytically, the resulting random field of averaged strength is assumed to represent a series of representative volume elements (RVEs) and the global strength is found by solving for the minimum of such an effective strength field. All these operations can be written analytically and there are only four model parameters: the three dimensions of the averaging volume (RVE) and the length of the final weakest-link chain. The model is verified using detailed numerical computations of notched and unnotched concrete beams simulated by mesoscale discrete simulations of concrete fracture performed with probabilistic distributions of model parameters. The numerical model used for verification represents material randomness both by assigning random locations to the largest aggregates and by simulating random fluctuations of material parameters via a homogeneous random field.*

1 INTRODUCTION

The mechanical responses of heterogeneous quasibrittle materials like concrete are intrinsically intertwined with non-homogeneity and randomness at finer scales of material resolution. In order to represent the cascading events of crack nucleation, growth and their interaction a variety of models have been developed.

Taking into account the spatial variation of material properties is also of paramount importance in the safety and reliability evaluation of engineering structures. Nowadays, the simplest and most oft-used method to account for material spatial randomness is classical Weibull theory [21, 20]. The structure is viewed as composed of many small reference volumes that are independent and have random strengths, no redistribution of stresses is considered, and the failure of any piece of material triggers the failure of the whole system. This holds even for nonuniform stress fields: the Weibull theory allows to compute an integral over the structure volume the result of which can be interpreted as the equivalent number of equally stressed material elements. The input to this Weibull integral is the shape of the stress field just before failure; see e.g. Sec. 4.1 in [19] for details. In this approach, all information about the mechanics of failure is lost, and the structural geometry becomes irrelevant: the integral transforms the structure into uniaxial bar with a constant stress. Indeed, from the viewpoint of Weibull theory, any piece of material can be viewed as a chain of elements (in series coupling) and thus statically determinate. This is unrealistic for real materials. Another caveat relates to the assumption of the spatial independence of local strengths. Weibull distribution is one of the three stable forms of extreme value distributions [10]. Its derivation assumes the survival probability of the structure as the product of survival probabilities of all infinitesimal pieces of material, which is only correct if these survival probabilities are independent. Spatial correlation incorporates a length scale, though, and the Weibull theory must be modified accordingly [17]. There is also a difficulty associated with the correct determination of the effective number of dimensions, as not only the geometrical similarity influences it. If, for example, a 3D structure fails only after the whole thickness fails (no matter how much the thickness is), the 3D geometrical scaling represents 2D scaling in Weibull theory, which is sometimes a source of confusion in experimental data interpretation; see [19] or Sec. 12.3.3 in [2]. Another example is when the failure can be triggered by two different and independent failure mechanisms: either a piece of material fails in the volume or by failure triggered at surface flaw. This is possible in the theory and as the structure is scaled in 3D, the amount of volume scales with the third power but the amount of surface material is scaled quadratically. Another problem with a direct application of Weibull theory based on the elastic stress field is that in cracked bodies or bodies with sharp notches, the singularity of the stress field causes the Weibull integral to diverge, thus predicting an infinite effective volume and zero strength. The self-similarity embodied in classical Weibull theory means that the strength of any piece of material is Weibullian, and only the volume and effective dimensionality decide the scale parameter; the shape parameter is size-invariant. These assumptions are not acceptable, and an alternative model must be developed to predict statistical strength in a manner that reflects the true behavior of heterogeneous materials such as concrete.

In this paper, we use results obtained with a particular class of discrete mesoscale models [5, 6] which was enhanced [9] by additional spatial variability in material properties via random fields. The discrete mesoscale models exhibit a certain variability in response due to the random placement of numerical aggregates which leads to random dimensions and orientations of “bars” connecting the aggregate centers. These random geometrical properties, when combined with deterministic materials properties, lead to scatter in structural response such as the peak

loads, sequences of local events and the related crack trajectories etc. Studies performed in [9] showed that scatter is not representing the experimental data obtained with concrete and therefore an additional sources of randomness was introduced. Random fields of material parameters provide spatial variability in a controlled fashion. Not only that the distribution of local properties can be described by the distribution of a random field, but also the spatial correlation is under control. It has been found that there exist a meaningful set of parameters of random fields used for modeling the otherwise deterministic parameters in the discrete model to accurately mimic the experimental data [9]. However, the connection of these spatial variability parameters to material and specimen production is not completely clear. Therefore, one can assume a range of possible setting of random fields to obtain a range of possible structural responses when predicting the true behavior.

A systematic study regarding the effect of parameters of the random fields has been performed in [8] to show the effect on both notched and unnotched structural elements made of concrete. It was shown that notched structures with stress concentration are affected by statistical variability in a different way than unnotched ones and that behavior of concrete bars under pure tension may behave very differently depending on a particular realization of the local strength field. The present paper, which is a promotion of a recent journal paper [18] provides a simple explanation to the different behavior of notched and unnotched specimens and proposes a simple analytical model that can replace the expensive random discrete mesoscale simulations. The analytical model is able to predict the peak force statistics by computing the extreme values of sliding averages of random strength fields.

2 OVERVIEW OF THE PROPOSED ANALYTICAL MODEL

The Introduction section discussed the classical Weibull theory along with a critique of its weaknesses. Several assumptions were found unacceptable and therefore an alternative model for the prediction of statistical strength is developed here so as to better reflect the true behavior of heterogeneous concrete-like material. We formulate the following simple *hypotheses* embodied in the proposed analytical model:

1. There is a *representative volume element* (RVE) that can be identified for a given material, geometry, dimensions, boundary conditions, etc., which is defined in such a manner that its failure leads to the exhaustion of structural strength (peak load). The dimensions of the RVE are *not* dependent on the parameters of the local strength random field. The volume of structure can be discretized into many RVE subvolumes.
2. Each RVE is composed of many microbonds that contribute to its strength. Within each RVE, one can define *effective strength* as the *moving average* (within RVE volume) of local strengths found within that RVE. This RVE strength is random and is a result of combinations of both serial and parallel couplings of these microbonds. The inside of each RVE can potentially undergo stress redistribution in response to a change in the external loading. Depending on the redistribution potential of the material, the RVE can have different degrees of brittleness/ductility and the probabilistic distribution of its strength is influenced correspondingly. Effective strength may be a random function and it represents a *barrier* for the effective stress (*action*) defined in the next item.
3. During loading, one can define *effective stress* at each potential RVE center as the local average of stresses found within that RVE. Thus effective stress is a homogenized part of stress obtained as an average over the RVE volume defined in item 1. The effective stress evolves during the loading process.
4. The structure can be discretized into many potential RVEs that may or may not share

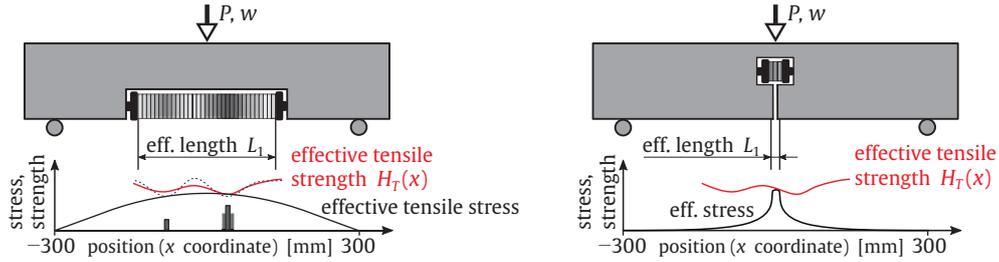


Figure 1: Illustration of the transformation of the random 3D discrete mesoscale model into an effective 1D model of a chain of random RVEs. From the left: unnotched bending, notched bending and uniaxial tension.

an identical level of load expressed via the effective stress. We assume that at the moment of attaining the *peak load* of the structure, the inelastic strains *localize* into a single macrocrack inside one of the RVEs. At this moment the effective (tensile) strength of that critical RVE attains its effective stress. This moment corresponds to the structural failure because the exhaustion of the effective tensile strength of a critical RVE signals the peak load of the whole structure.

5. The failure of any RVE occurs after redistribution of stresses takes place inside the RVE volume. The failure typically occurs after many of the bonds within the RVE exceed their random capacity. The number of these bonds increases with the redistribution potential expressed via toughness or effective fracture energy. Even though the model features a strength-based failure criterion, fracture energy influences the dimensions of the RVE. The effective strength of a potential RVE is effectively Gaussian within a wide range around the mean strength; the degree of normality (spread of the Gaussian core) increases with the number of parallel couplings involved in the averaging operation (and thus with the RVE size). In concrete, the strength probability distribution function can be reasonably considered Gaussian in the central region. This distribution can be considered to have gradual transitions towards power law tails [7, 11, 1, 17].

Item 1 defines the RVE in a different way than is usual in the modeling of heterogeneous materials. We remark that the term ‘representative volume element’ may have various meanings [12, 13, 14, 15]. The problem with the definition of an RVE is that it depends on the property is to be represented [4], and therefore its size depends on the type of treated physical phenomena, microstructure geometry and the contrast between microstructure constituents [16].

Moreover, the effective strengths of a potential RVE can generally be *statistically dependent* and we assume that the *effective strength is an autocorrelated random field*, which is not considered in applications of the extreme value theory to the weakest-link model of independent links. The random field of effective strength is a result of the redistribution of stresses within an RVE volume/window, and the redistribution is taken into account simply via the moving average operation. Therefore, the above-mentioned weaknesses of the classical Weibull theory are removed: stress redistribution is taken into account (it incorporates a length scale) and so is the spatial correlation of RVE volumes (it incorporates another length scale, the effective autocorrelation length, which depends on both the autocorrelation of local strengths of a finer scale and the RVE window size). Another deviation from the Weibull theory is that the distribution of effective strength is no longer Weibullian (with the size-independent shape and scale parameter being scaled from a reference one corresponding to a reference size).

3 APPLICATION TO RANDOMIZED MESOSCALE DISCRETE MODEL

Consider a unit-mean Gaussian random field $h(\mathbf{x})$ where the spatial coordinate \mathbf{x} is defined in three dimensions (the beam volume). This random field represents the local strength (or simply its dimensionless multiplier as in the current application). It is a random function that depends on the spatial coordinate, \mathbf{x} . We consider h to be a *homogeneous* random field in the strong sense, meaning that the distribution of the random field is independent of the location, and that the autocorrelation structure among the local random variables is also shift-invariant, i.e. it only depends on the mutual distance between the points. We also consider that $h(\mathbf{x})$ is ergodic (the spatial average is equal to the ensemble average). Thus, our random field is fully defined by the constant mean value $\mu_h = 1$, standard deviation δ_h and autocorrelation function $\rho(\boldsymbol{\tau}; \ell_\rho)$, where $\|\boldsymbol{\tau}\|$ is the lag (distance between two spatial points). This function is considered to be a separable isotropic Gaussian (squared-exponential) autocorrelation function

$$\rho(\boldsymbol{\tau}; \ell_\rho) = \exp \left[- \left(\frac{\|\boldsymbol{\tau}\|}{\ell_\rho} \right)^2 \right] = \prod_{v=1}^3 \exp \left[\left(- \frac{\tau_v}{\ell_\rho} \right)^2 \right], \quad |\tau_v| \geq 0 \quad (1)$$

The separability means that the correlation between two different random variables $h(\mathbf{x}_1)$ and $h(\mathbf{x}_2)$ is a product of autocorrelations that depend solely on distances τ_v , i.e. projections of the lag $\|\boldsymbol{\tau}\| = \sqrt{\sum_v^3 \tau_v^2}$ along individual dimensions $v = 1, 2, 3$. Therefore, for such a fully separable autocorrelation, we can write

$$\rho(\boldsymbol{\tau}) = \rho(\tau_1, \tau_2, \tau_3) = \rho(\tau_1)\rho(\tau_2)\rho(\tau_3) \quad (2)$$

The isotropy means that the autocorrelation length, ℓ_ρ , is identical in all three directions.

The additional randomness due to the spatial variability of material properties in the discrete model is incorporated into the constitutive relation via modifying the main parameters that control the tensile response of the material using a random field $h(\mathbf{x})$. These parameters, which are associated with the individual contacts/bonds of the discrete model, are the tensile strength and fracture energy f_t and G_t . As discussed in [8], the parameters are randomized via only one random spatially varying multiplier $h(\mathbf{x})$ in such a way that the local Irwin/Hillerborg characteristic length is kept unmodified and constant throughout the beam volume. This is achieved [19] by taking $f_t(\mathbf{x}) = \bar{f}_t h(\mathbf{x})$ and $G_t(\mathbf{x}) = \bar{G}_t [h(\mathbf{x})]^2$, where \bar{f}_t and \bar{G}_t denote the deterministic model parameters. If h is a constant (independent of \mathbf{x}), two beam simulations that differ only in two values of a random multiplier, say h_1 and $h_2 = c \cdot h_1$, $c > 0$, will follow exactly the same cracking process and the computed forces will simply have a ratio of c .

The probabilistic part of the model is fully characterized by a unit-mean Gaussian homogeneous random field $h(\mathbf{x})$ which has two free parameters: the coefficient of variation, δ_h , and the autocorrelation length, ℓ_ρ , which is the parameter of the selected squared-exponential isotropic autocorrelation function. The random field is stationary in the strong sense, meaning that the distribution is identical throughout the whole domain (specimen volume), and also the autocorrelation structure is shift-invariant. In fact, the distribution function of the strength multiplier, $h(\mathbf{x})$, should be considered to have a modified left tail – it was assumed to follow Weibull-Gaussian distribution, i.e. Gauss distribution with a Weibullian left tail grafted at very low probability, see [8]. The cumulative distribution function of Weibullian random variable h reads $F_W(h) = 1 - \exp[-(h/s)^m]$ where m and s are the shape and scale parameters, respectively. The left tail grafted at very small values of strength practically does not influence the analysis performed here with purely Gaussian distribution.

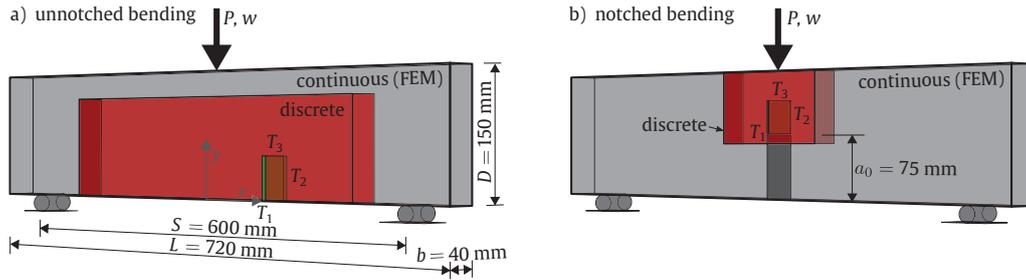


Figure 2: Specimen geometry for a) unnotched and b) notched beams loaded in three point bending. The narrow blocks of dimensions T_1 , T_2 and T_3 are the three RVE length parameters of the proposed analytical model.

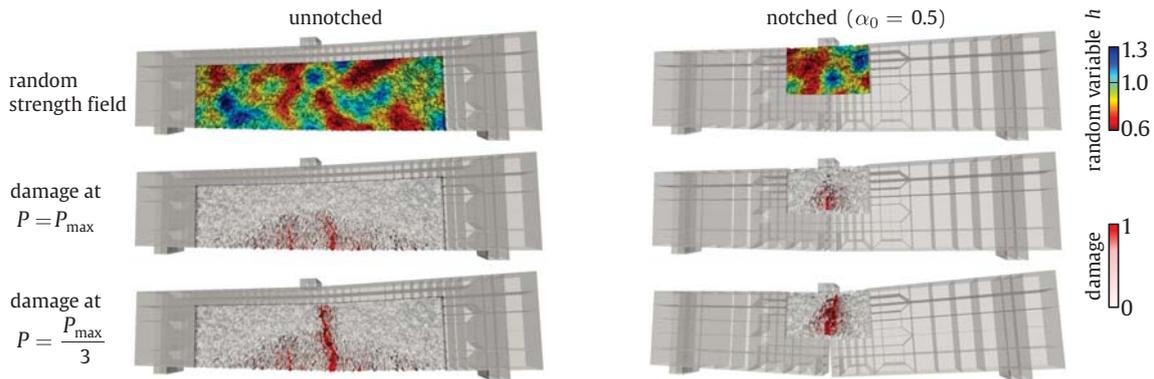


Figure 3: Distribution of damage at the inter-particle contacts of one realization of the *probabilistic* model ($\ell_p = 25$ mm) at the peak load (top) and at the termination of the simulation (bottom) of unnotched and notched beams. For definition of “damage”, please see the description of the mesoscale discrete model in [8].

The random fields used to represent material parameters are generated with various autocorrelation lengths spanning from $\ell_p \rightarrow 0$ (independently sampled random variables) up to the infinitely long autocorrelation length $\ell_p \rightarrow \infty$, for which the realizations are random constant functions, and therefore the whole structure shares the same value in a single realization. The strengths of the beams were statistically evaluated as functions of the autocorrelation length and variance of the random field.

The model was employed to simulate the three point bending of concrete beams with and without a central notch, see Fig. 2. Fig. 3 presents examples of the typical patterns of damage for selected realizations of the random strength field multiplier. One can see that the damage is more localized in the notched beams.

In the two bending geometries considered here (unnotched and notched three point bend beams), the identification of the “chain of RVEs” is particularly simple. By studying the stress fields of the beams in the discrete model at the peak load, we can compute the effective stresses at the peak load. Fig. 1 illustrates the idealization used in the analytical model. The failure of the weakest RVE thus corresponds to the flexural strength (peak load) of the beam. Each of such RVEs is assumed to be a cuboid; see the thin green cuboids in Fig. 2. The series coupling of potential RVEs over the effective length is depicted in the highly tensioned zones. At the bottom, the averaged stress fields and a realization of the effective strength field are depicted together with the decisive macrocrack location. Thanks to the ability to redistribute stress internally, the tensile stress fields can be considered roughly constant over a certain length at the bottom part of the beams. To support this claim, the effective tensile stresses obtained from one realization of the beams modeled by the discrete mesoscale simulations are depicted

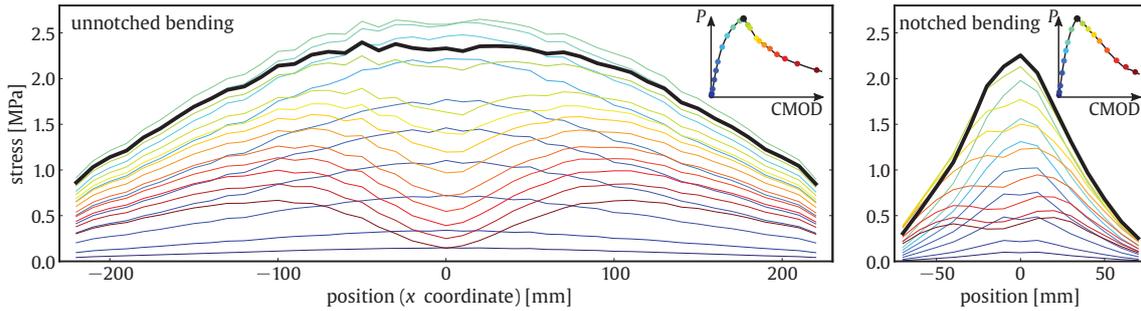


Figure 4: Evolution of the *effective stress* in three point bend beams as predicted by the discrete mesoscale model. The line colors correspond to various stages of loading: pre-peak stages (thin blue), peak load (thick black), right after peak load (thick green) and postpeak stages (red to brown); [18].

in Fig. 4. These profiles of the effective tensile stresses are calculated as sliding averages of windows corresponding to the RVE sizes (length about 10 mm, depth 40 mm or 50 mm and beam thickness 40 mm), see below. In the case of unnotched beams, the length of an effective chain can become as great as 300 mm, which is one half of the bending span. In the notched beams, however, the length is very limited as the effective stress field is very localized around the notch tip. Note that the effective stress profile does not have to be constant and more complicated forms can also be considered when computing this effective chain length. The effective stress field (action) is considered approximately deterministic from here on. It is true that the averaged stress exhibits a certain degree of variability influenced mainly by the size of the averaging window. Indeed, various realizations of the mesoscale model (different positions of the grains) return slightly different the stress fields. However, the averaging window contains many grains and therefore the variance in the effective stress can be disregarded; see Fig. 4. A rigorous approach to incorporate this variability has been presented in [22].

The most important information gained from the analyses [8] performed with the probabilistic mesoscale model is that the volume within which massive redistribution takes place right at the peak load is almost independent of the parameters of the random field. As shown in [18], when considering the strengths of individual bonds being described by a random field, the effective strength parameters of each RVE (an averaging window) can be predicted analytically, together with the spatial correlation of these averaged strengths. Therefore, the three-dimensional problem is transformed into a one-dimensional problem (a chain) with an effective strength variable along the beam span, see Fig. 1. The effective strength becomes a random process that can be mathematically described as a result of the moving average of a local random strength field [18].

When this effective strength profile is being attained by a constant effective stress, it suffices to merely obtain the *minimum of a random field* over a certain effective length [18].

Since local strengths at the mesolevel are almost entirely Gaussian, the effective strength of individual potential RVEs is Gaussian, too. This is due to the averaging operation that suppresses the tails (by virtue of the central limit theorem the Gaussian core spreads wider). Therefore, it suffices to focus on the extremes of Gaussian random fields over a closed interval: the effective length L_1 . This length must be determined by considering both the stress field and the properties of the sample paths of the effective strength (its gradients). In the studied examples of three point bending and tension, the estimation of chain length is provided in [18].

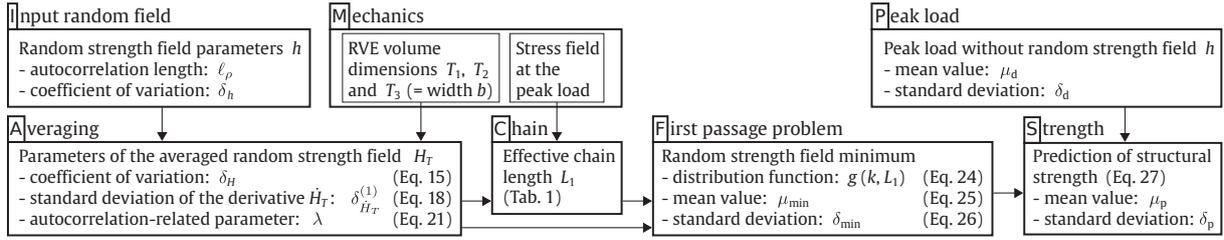


Figure 5: Flow chart of the model. The top row represents the input information for the proposed model sketched in the bottom row. Equation numbers correspond to equations in [18].

3.1 Three point bend beams

The application of the proposed model to bending requires either (i) an analysis of the exceedance probability function that features non-constant effective threshold (stress) $u(x)$ over the whole bending span, or (ii) determination of the effective length of chain in which the macrocracks appear and in which the effective stress can be considered approximately constant. The first alternative is generally possible, but it leads to a somewhat more complicated formula for the probability density function of exceedance.

In this work we have selected the assumption of a roughly constant effective stress profile by taking advantage of the fact that there is an effective chain length L_1 which covers the range of high stresses where a crack occurs and in which the effective stress does not vary considerably.

The flow of the information in the proposed model is sketched in flowchart in Fig. 5. The information about random strength field (dimensionless local strength multiplier; see box I) is processed analytically to represent averaging using the estimated RVE size (the three dimensions of RVE can be estimated or identified from a detailed model for the mechanics; see box M). The result of the averaging operation is a detailed description of the transformed effective strength random field (A). Another information needed that represents the mechanics is the stress field at the peak load (M); in particular, we need to estimate the extent of regions that share almost identical effective stress at the peak load. Using this information and the data from effective strength random field, the effective “chain length” can be deduced (C). The next step is simply computation of the minimum strength over the effective length (F). In cases when there is no other source of randomness, the computed mean value and standard deviation can be readily used as multipliers of the deterministic peak load. In our case, there is another source of variability in the peak load predicted by the discrete mesoscale model, see below. This source is not controlled by the random strength field and therefore this information (P) is additionally passed to obtain the final result: the prediction of random structural strength parameters (S); see also Sec. 3.3 on this topic. Note that box (P) can also be used to carry information about additional sources of randomness independent from local strength random field; e.g. due to testing imperfections etc.

3.2 Identification of the model parameters

There are in total four parameters of the proposed analytical model that must be inferred (i.e. identified) from the discrete simulations: T_1, T_2, T_3 and L_1 ; see box M in Fig. 5. These are (i) the three dimensions of the local averaging RVE window (a volume whose failure triggers the failure of the whole structure), and (ii) the length of the effective chain (plus the information about the effective stress function over the length of this chain). The rectangular cubes of the RVE are assumed to have identical dimensions: the length T_1 measured along the beam span, the depth T_2 measured along the vertical axis, and the width T_3 ; see the illustration of one such

RVE in Fig. 2. With these parameters, the probability distribution function of structural strength is obtainable analytically [18].

We first focus on the three dimensions of the averaging RVE volume. The width T_3 is not a free parameter: it must be taken as the beam width, $b = 40$ mm, through which the crack front must pass. The lengths T_1 and T_2 are two length parameters that must be obtained either from a nonlinear analysis (e.g. with a discrete model) or estimated. As argued in, e.g. [3, 2], in the case of modulus of rupture (unnotched beams) there is a boundary layer of microcracking that develops prior to reaching the peak load. The depth of this layer, D_b , is approximately proportional to the maximum aggregate size, and thus independent of the structural size. We use the averaging depth $T_2 \approx 2D_b$, which is based on simulations performed in [8]. Indeed, the lengths T_1 and T_2 represent the width and depth of the region surrounding the macrocrack that forms at the peak load, or right after it has been reached: see Fig. 7 in [8]. Numerous analyses with the discrete model have confirmed that these lengths are not considerably dependent on the parameters of the random field; they can be obtained from the “deterministic” model in [8]. The depths T_2 of RVEs in the case of unnotched and notched beams do not differ much [18].

The averaging dimension T_1 is controlled by the irregular inner structure and also by the macroscopic stress field. In the direction of x_1 , the averaging width T_1 is considered because it is known that the crack, once it has localized, is not perfectly planar. The width T_1 is related to the maximum aggregate size, $d_{\max} = 10$ mm, and we conjecture that the fracture energy and the stress field have additional influence: in the case of unnotched beams, we have found that the crack “planes” are somewhat tortuous and therefore the RVE width T_1 is greater than in the case of notched beams that are exposed to strongly localized stress fields; see Fig. 4. Note that the relatively large averaging lengths T_2 and T_3 make the strengths within each RVE almost constant over directions x_2 and x_3 . The only variability in space takes place along direction x_1 , and this is what allows us to perform the one-dimensional idealization into the “effective chain” of RVEs. The RVE dimensions are thus selected.

The last, fourth, parameter of the model is the effective length L_1 . It is the extent of the zone within which cracks will frequently appear in the beam, and is therefore dependent on the effective stress field (obtained by averaging with the three RVE dimensions). In both types of beams, however, the length L_1 is also influenced by the random field of the effective strength. This is because both the autocorrelation length and the variance influence the random *gradient* of the wavy function describing the RVE strengths along the beam. The effective chain has the length of the interval where the effective stress may attain the effective strength, and therefore it depends on both processes, see [18] for more details.

The *unnotched* beams have, at the peak load, a very long zone of almost constant stress leading to large L_1 . These mild functions develop thanks to the redistribution capacity of the material. The shape of the effective stress (action) is parabolic; see Fig. 4. Therefore, its gradient is an almost exactly linear decreasing function. The gradient of the effective strength (barrier) is a Gaussian random variable with zero mean and standard deviation $\delta_{\dot{H}_T}$. The length over which the two processes can meet is therefore proportional to $\delta_{\dot{H}_T}$ (with the dimension of load per distance) and inversely proportional to the slope of the decreasing first derivative of the stress process (i.e. inversely proportional to its constant curvature). The situation is illustrated in Fig. 6. By displaying several realizations of the *averaged* random field, one can see how the strengths compare to the same effective stress profile for various variances and autocorrelation lengths of the local strength.

In the case of *notched beams*, the effective stress field is very localized and the length L_1 is very short. The crack must initiate at the notch tip, but various cracks form a “fan” depending

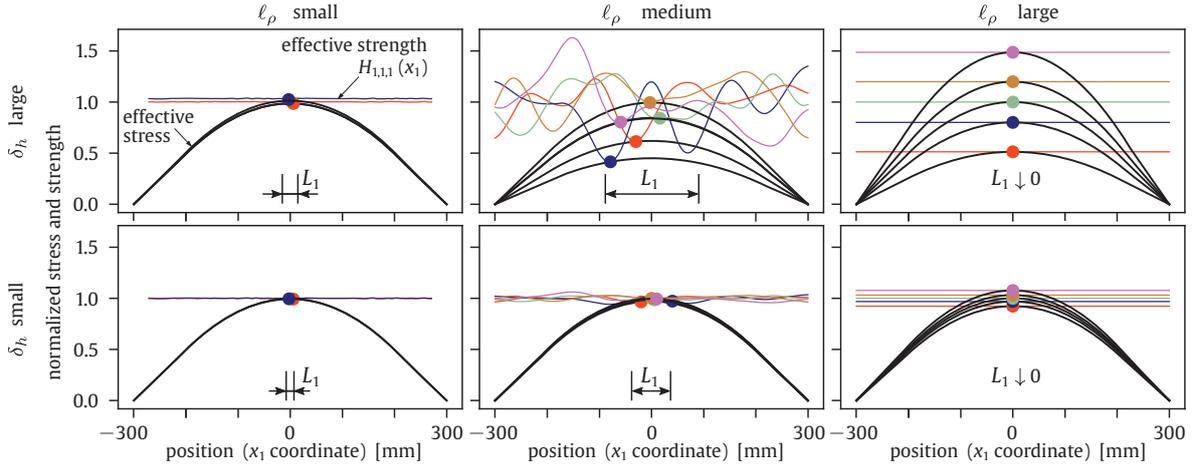


Figure 6: Illustration of the dependence of effective span L_1 (chain length) on the autocorrelation length ℓ_ρ and variance δ_h^2 of the local (non-averaged) random field, [18]. The effective stress field is depicted for the unnotched beams. Top row: low variance. Bottom row: large variance.

on the random barrier ahead of the current crack tip.

3.3 Comparison with the discrete mesoscale model

Having fixed all the parameters of the proposed analytical model, one could now compare the model with results of the parametric study presented in [8]. Fig. 7 presents a comparison between (i) the sets of the random discrete mesoscale simulations (a circle with errorbars representing $\mu_p \pm \delta_p$) and (ii) the results obtained by the proposed model (a solid line surrounded by a scatterband). An excellent agreement between the analytical predictions and the random discrete mesoscale simulations is obtained for the whole studied range of parameters ℓ_ρ and δ_h of the local random strength field.

We first comment on the asymptotic behavior, i.e. when the autocorrelation length approaches either zero or infinity. When $\ell_\rho \downarrow 0$, the local averaging within an RVE effectively removes any variability in the effective random strength field ($\delta_H = 0$). Therefore, the chain strength has zero variance, $\delta_{\min} = 0$ and $\mu_{\min} = 1$. Thus, the model predicts that the deterministic model solely governs the behavior

$$\mu_{p,0} = \mu_d, \quad \delta_{p,0} = \delta_d \quad (3)$$

The results obtained from the random discrete mesoscale model slightly differs for two reasons: (i) in the random discrete model, there is still room for a limited weakest-link principle, and the separation of randomness in the “deterministic model” and the local random strength field is only an approximation; and (ii) the finite size of contacts/particles does not allow ℓ_ρ to decrease below the model resolution. Indeed, especially when the random field variability is high ($\delta_h = 0.28$), the variability of local strengths is not averaged out completely within the FPZ. However, the differences between the predictions of the discrete model and the analytical model are minor.

On the other extreme, when $\ell_\rho \uparrow \infty$, the realizations of effective strength processes along the chain are random constant functions with a unit mean value and a standard deviation that is not affected by the averaging: $\delta_H = \delta_h$. Since the chain length $L_1 = 0$, the standard deviation of the weakest RVE is $\delta_{\min} = \delta_h$ and the mean value remains $\mu_{\min} = 1$. Therefore, the structural strength is simply the multiple of two independent sources of variability and the mean value

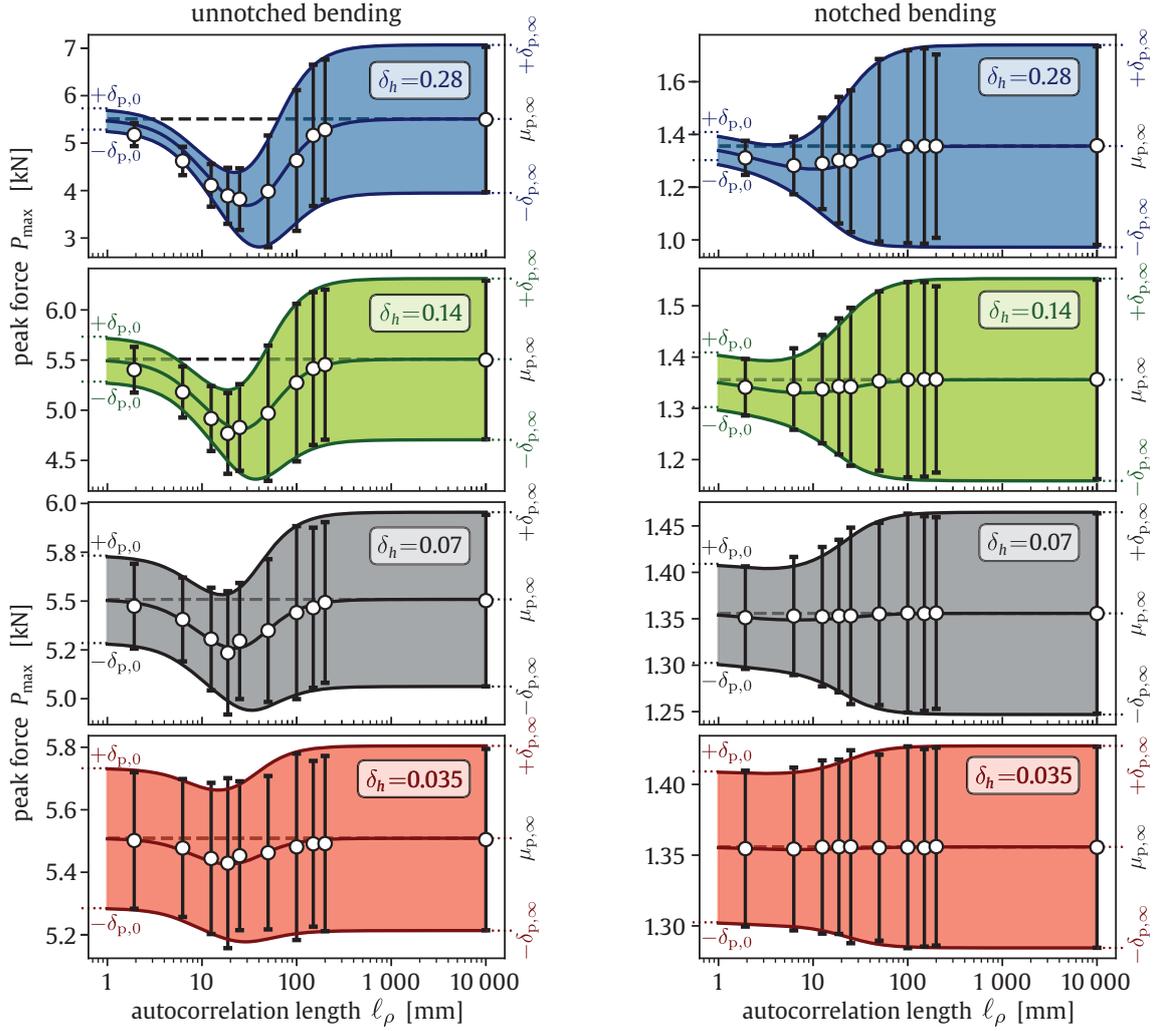


Figure 7: The mean value and standard deviation of the maximum load computed on unnotched (left) and notched (middle) beams loaded in three point bending and the tension of half-depth prisms (right) using the *probabilistic* discrete mesoscale model, denoted by black empty circles and errorbars. The colored curves show results obtained with the proposed model based on extremes of averaged random fields. The four different values of standard deviation (coefficients of variation) of the local strength random fields, δ_h , are highlighted in the boxes.

and standard deviation as

$$\mu_{p,\infty} = \mu_d, \quad \delta_{p,\infty} = \sqrt{\delta_d^2 \delta_h^2 + \mu_d^2 \delta_h^2 + \delta_d^2} \quad (4)$$

The match with random discrete simulations is absolute. Here, we recall the explanation from Sec. 2 of [8] that for $\ell_\rho \uparrow \infty$ the structural strength is simply proportional to variable h . This is a consequence of the selected alternative of scaling tensile strength, f_t , and fracture energy, G_t , in such a manner that the mesoscale Irwin/Hillerborg characteristic length is kept constant.

Let us now the study the behavior for intermediate values of autocorrelation length. The monotonic increase in the standard deviation $\delta_p(\ell_\rho, \delta_h)$ of the peak load with an increasing autocorrelation length ℓ_ρ is affected by the averaging volume $T_1 T_2 T_3$ via the variance reduction function. Having a good match with the standard deviation confirms the selection/identification of the RVE volume. Since T_3 must be the beam width b , the only free parameters of the model are the proportion T_2/T_1 and the chain length L_1 . The pair denoted T_1 and L_1 influences the dependence of the average strength $\delta_p(\ell_\rho, \delta_h)$ on the autocorrelation length and the random

field variance. When the autocorrelation length ℓ_ρ is roughly of the order of the averaging length T_1 , the average strength μ_p undergoes a noticeable drop, which becomes pronounced with increasing random field variance. The T_1/ℓ_ρ ratio is small enough to limit the averaging effect and yet large enough to activate the weakest-link effect. In this way, the probabilistic length scale ℓ_ρ and the deterministic length scale T_1 interact.

A deeper understanding of the mechanisms leading to the drop in average strength for certain combinations of the parameters of the random field can be achieved by analyzing the presented analytical model. It is a result of a combination of several factors, among which the derivative of the effective strength profile, $\delta_{\dot{H}}$, plays a role. Indeed, the increase in $\delta_{\dot{H}}$ is detrimental to the *average* structural strength in two different ways:

- The mean upcrossing rate of the averaged random field
- The effective chain length L_1 is proportional to the gradient: $L_1 \propto \delta_{\dot{H}}$ (in bent beams).

The length of the virtual chain is, however, not the only factor influencing the mean strength. Any increase in ℓ_ρ also leads to an increase in the standard deviation of the averaged random field, which is detrimental to the strength of the chain. The interplay of these effects leads to the relatively complicated dependence of the minimum average chain strength on the two parameters of the random field: ℓ_ρ and δ_h . For the studied range of standard deviations δ_h , the critical autocorrelation lengths vary between 20 and 35 mm (unnotched beams) and 6–10 mm (unnotched beams); see the left and middle columns in Fig. 7.

While the strength troughs are very pronounced in the *unnotched* beams, in *notched* simulations the average peak load is found to be only very weakly sensitive to the spatial variability in material parameters. The reason is that the stress concentration is so severe that the crack is forced to propagate from one specific location (the notch tip) and the spatial variability in material parameters is not sufficient to change the location of dissipative processes. In other words, the weakest-link effect is strongly reduced. However, the standard deviation of the peak load decreases with the decrease in the autocorrelation length due to the averaging of the fluctuations within the RVE, which was found to be independent of the applied random field in [8].

4 CONCLUSIONS

This paper promotes a recently published analytical model that predicts statistics for the random strength of beams made of quasibrittle materials in which strength is assumed to vary according to a random field, in particular a homogeneous isotropic random field with Gaussian distribution and an arbitrary separable autocorrelation function. These strength statistics can also be obtained via costly discrete mesoscale simulation with additional local strength variability modelled by a random field. We argue that if data obtained from the random discrete mesoscale model [8] are used, the erratic patterns of local *stresses* in a concrete body may be replaced by locally averaged (effective) stresses for the purpose of the overall strength analysis. Failure becomes governed by a nonlocal criterion. If the local stress average exceeds a critical threshold at any location, the system (structure) is assumed to attain its peak load. Here the critical threshold is the effective *strength* obtained as the local average of strengths with a certain volume of the heterogeneous material.

The proposed approach relaxes two assumptions of the classical Weibull theory: (i) the consideration of *effective strength* reflects the progressive loss in material integrity and the associated stress redistribution within a representative volume element (RVE) that takes place prior to reaching the peak load and, (ii) the consideration of spatial correlation departs from the Weibullian assumption of the independence of strengths at various locations. The RVE window encloses the smallest material volume whose failure may trigger the failure of the whole struc-

ture. The dependence of its strength on the point variance and on the autocorrelation length of the random field is elucidated. The analysis provides insight into the interaction between (i) the probabilistic length scale (introduced via the autocorrelation length of the random strength field) and (ii) the deterministic length scale (expressed via the dimension of the averaging RVE).

The model is shown to agree well with results obtained for three point bend specimens with and without a notch that were analyzed using the random discrete mesoscale model. The uniaxial tension of prisms is shown to exhibit more complicated behavior for which the model's assumptions do not hold. The presented model illustrates the transformation of a three-dimensional structure into a chain of RVEs. In the case of notched beams, the possibility of sampling a crack location randomly is quite limited by the stress concentration, and thus the weakest-link principle is suppressed to a high extent. The strength of notched beams is only modified in terms of its variance, but the distribution of beam strength is almost exactly proportional to the strength of a single RVE. In unnotched beams, the weakest-link principle modifies the distribution from that of a single RVE to the extreme value type.

The model has the deterministic/energetic features embodied by considering the redistribution within an RVE (three lengths). Given this simplification and the knowledge of the spatial variability of local material strength, the entire size effect (the dependence of the mean and standard deviation of structural strength on size) can be explained from a pure statistical viewpoint.

ACKNOWLEDGMENT

The author acknowledges financial support provided by the Czech Science Foundation via project no. GC19-06684J.

REFERENCES

- [1] Zdeněk P. Bažant and Sze-Dai Pang. Activation energy based extreme value statistics and size effect in brittle and quasibrittle fracture. *Journal of the Mechanics and Physics of Solids*, 55(1):91–131, 2007.
- [2] Zdeněk P. Bažant and Jaime Planas. *Fracture and Size Effect in Concrete and Other Quasibrittle Materials*. CRC Press, Boca Raton and London, 1998.
- [3] Z. P. Bažant and Zhengzhi Li. Modulus of rupture: Size effect due to fracture initiation in boundary layer. *Journal of Structural Engineering*, 121(4):739–746, 1 1995.
- [4] Zdenek P. Bažant. Can multiscale-multiphysics methods predict softening damage and structural failure? *International Journal for Multiscale Computational Engineering*, 8(1):61–67, 2010.
- [5] Gianluca Cusatis and Luigi Cedolin. Two-scale study of concrete fracturing behavior. *Engineering Fracture Mechanics*, 74(1-2):3–17, 2007.
- [6] Gianluca Cusatis, Daniele Pelessone, and Andrea Mencarelli. Lattice discrete particle model (LDPM) for failure behavior of concrete. I: Theory. *Cement & Concrete Composites*, 33(9):881–890, 2011.
- [7] Henry Ellis Daniels. The statistical theory of the strength of bundles of threads. I. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 183(995):405–435, 1945.

- [8] Jan Eliáš and Miroslav Vořechovský. Fracture in random quasibrittle media: I. Discrete meso-scale simulations of load capacity and fracture process zone. *Engineering Fracture Mechanics*, 235:107160, 2020.
- [9] Jan Eliáš, Miroslav Vořechovský, Jan Skoček, and Zdeněk P. Bažant. Stochastic discrete meso-scale simulations of concrete fracture: comparison to experimental data. *Engineering Fracture Mechanics*, 135:1–16, 2015.
- [10] R. A. Fisher and L. H. C. Tippett. Limiting forms of the frequency distribution of the largest and smallest member of a sample. *Mathematical Proceedings of the Cambridge Philosophical Society*, 24(2):180–190, 1928.
- [11] D. G. Harlow, R. L. Smith, and H. M. Taylor. Lower tail analysis of the distribution of the strength of load-sharing systems. *Journal of Applied Probability*, 20(2):358—367, 1983.
- [12] R. Hill. Elastic properties of reinforced solids: Some theoretical principles. *Journal of the Mechanics and Physics of Solids*, 11(5):357–372, 1963.
- [13] H. Moussaddy, D. Therriault, and M. Lévesque. Assessment of existing and introduction of a new and robust efficient definition of the representative volume element. *International Journal of Solids and Structures*, 50(24):3817–3828, 2013.
- [14] M. Ostoja-Starzewski. Microstructural randomness versus Representative Volume Element in thermomechanics. *Journal of Applied Mechanics*, 69(1):25–35, 06 2001.
- [15] Martin Ostoja-Starzewski. Material spatial randomness: From statistical to representative volume element. *Probabilistic Engineering Mechanics*, 21(2):112 – 132, 2006.
- [16] M. Stroeven, H. Askes, and L.J. Sluys. Numerical determination of representative volumes for granular materials. *Computer Methods in Applied Mechanics and Engineering*, 193(30):3221–3238, 2004. Computational Failure Mechanics.
- [17] Miroslav Vořechovský. Incorporation of statistical length scale into Weibull strength theory for composites. *Composite Structures*, 92(9):2027–2034, 2010.
- [18] Miroslav Vořechovský and Jan Eliáš. Fracture in random quasibrittle media: II. Analytical model based on extremes of averaging process. *Engineering Fracture Mechanics*, 235:107155, 2020.
- [19] Miroslav Vořechovský and Václav Sadílek. Computational modeling of size effects in concrete specimens under uniaxial tension. *International Journal of Fracture*, 154(1-2):27–49, 2008.
- [20] W. Weibull. The phenomenon of rupture in solids. *Royal Swedish Institute of Engineering Research (Ingenioersvetenskaps Akad. Handl.)*, Stockholm, 153:1–55, 1939.
- [21] W. Weibull. *A Statistical Theory of the Strength of Materials*, volume (Handlingar Nr.) 151 of *Royal Swedish Institute of Engineering Research (Ingeniörsvetenskapsakademiens)*. Generalstabens litografiska anstalts förlag, Stockholm, 1939.
- [22] Zhifeng Xu and Jia-Liang Le. A first passage based model for probabilistic fracture of polycrystalline silicon mems structures. *Journal of the Mechanics and Physics of Solids*, 99:225–241, 2017.

A SEQUENTIAL MULTI-POINT SAMPLING PROCEDURE FOR SURROGATE MODELS

Matthias Fischer¹, Carsten Proppe¹

¹Chair of Engineering Mechanics, Karlsruhe Institute of Technology
Kaiserstr. 10, Bdg. 10.23, 76131 Karlsruhe, Germany
e-mail: {matthias.fischer, proppe}@kit.edu

Keywords: Surrogate Models, Sequential Sampling, Parallel Sampling, Gaussian Process Regression, Polynomial Chaos Expansion.

Abstract. *A sequential sampling procedure is introduced for Gaussian process regression and polynomial chaos expansion. The procedure consists of several sequential sample sets, each with a certain number of sampling points. A grid-based method for sample selection in the context of Gaussian process regression is proposed which aims to improve the model accuracy. The demonstrated methods are investigated for a test case. The obtained surrogate models are validated after each added sample set where the benefit of the proposed sampling methods becomes evident.*

1 INTRODUCTION

In many research areas, simulations have become considerably more computationally intensive over time. In the context of surrogate modeling, space-filling designs such as Latin hypercube sampling or Sobol sequences have been applied to ensure good coverage of the input space. Furthermore, sequential sampling methods have been given more focus in the past. In the context of *active learning*, numerous methods have been developed to achieve an optimal experimental design [1]. Various methods aim to add new samples in the input space based on the location of existing samples. For example, distance measures for samples or nested Latin hypercube sampling may be applied [2, 3]. However, these methods do not take the model evaluations for existing samples into account. The goal of sequential sampling methods may thus be extended to generate new samples based on an existing experimental design, model evaluations and a surrogate model such that the quality of the surrogate model can be optimally improved by new samples and model evaluations. Depending on the applied surrogate method, different sampling methods are available or preferable. For instance, active learning for Gaussian process regression is investigated in [4] and [5]. For polynomial chaos expansion, active learning has been frequently applied in structural reliability analysis, e. g. in [6]. In this paper, sampling methods for Gaussian process regression and polynomial chaos expansion are investigated and compared.

Because computation cost has become an increasing factor, the opportunity of parallelization has become more attractive. Parallelized sequential sampling procedures have therefore become a promising possibility [5, 7]. A certain amount of new samples is generated in each cycle of the sampling procedure so that model evaluations for the obtained set of samples can be run in parallel.

For Gaussian process regression, new samples are usually selected at points in the input space where the maximum prediction variance is present [8]. However, this is not necessarily the best choice in order to achieve an optimal model improvement. Furthermore, those points are likely to occur on the boundaries of the input space, especially in high dimensional problems with a small sample size. A new sample selection technique for Gaussian process regression is introduced that is based on expected Gaussian process regression models with respect to new samples.

For polynomial chaos expansion, new samples may be selected based on the information matrix [9]. This matrix is composed of polynomial basis evaluations for each sample, respectively. For example, D-optimal sampling or S-optimal sampling can be applied to compute new samples.

The objective of this paper is to put different sequential sampling methods for surrogate models in a general framework and to include the proposed sample selection technique for Gaussian process regression. In section 2, essentials about applied surrogate models are outlined. In section 3, the sampling procedure and sample selection techniques are described. In section 4, a test case is investigated and discussed. Finally, conclusions with regard to the sampling methods are given in section 5.

2 SURROGATE MODELS

The quantity to be estimated is of the form $f : \mathbb{R}^p \mapsto \mathbb{R}$ which maps inputs \mathbf{x} of dimension p to scalar-valued outputs y . The inputs are assumed to be independent and uniformly distributed on the interval $[0, 1]$. If this is not the case, isoprobabilistic transformation, such as the Rosenblatt transformation, can be applied. The goal is to find a set of sequentially added

samples \mathbf{x}_i which yield an optimal surrogate model, i. e. a model with minimum validation error. The surrogate model is built based on a set of samples and corresponding evaluations $\{\mathbf{X}, \mathbf{y}\} : \{(\mathbf{x}_i, y_i), i = 1, \dots, n\}$. In the following, essentials about the applied surrogate models are given.

2.1 Gaussian process regression (GPR)

Gaussian Process regression, also known as kriging, is a powerful statistical regression technique originating from geostatistics where the input data is treated as a spatial Gaussian process. [10] One crucial advantage of this surrogate method is that, aside from the prediction, it provides the prediction variance as error indicator. The prediction variance can be used in sample selection methods.

The prediction is defined as a weighted sum of observations

$$\mathcal{M}^{\text{GPR}}(\mathbf{x}) = \sum_{i=1}^n \lambda_i(\mathbf{x}) y_i \quad (1)$$

where the weights λ_i depend on the position \mathbf{x} in the input space. The weights are calculated by applying two requirements to the prediction. The prediction $\mathcal{M}^{\text{GPR}}(\mathbf{x})$ is assumed to be unbiased with respect to the function f and the variance of the difference between prediction $\mathcal{M}^{\text{GPR}}(\mathbf{x})$ and function f is assumed to be minimum. This leads to the prediction value and prediction variance

$$\mathcal{M}^{\text{GPR}}(\mathbf{x}) = \mu + \mathbf{k}^\top \mathbf{K}^{-1} (\mathbf{y} - \mu \mathbf{I}) \quad (2)$$

$$\sigma^2(\mathbf{x}) = \sigma_0^2 - \mathbf{k}^\top \mathbf{K}^{-1} \mathbf{k}. \quad (3)$$

Here $\mu = \frac{\mathbf{I}^\top \mathbf{K}^{-1} \mathbf{y}}{\mathbf{I}^\top \mathbf{K}^{-1} \mathbf{I}}$ is the kriging mean value and $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, $k_i = k(\mathbf{x}, \mathbf{x}_i)$, $\sigma_0 = k(\mathbf{x}_i, \mathbf{x}_i)$ where $k(\mathbf{x}_i, \mathbf{x}_j)$ is a valid kernel function. In this paper, the radial basis function

$$k(\mathbf{x}, \mathbf{x}') = \exp(-(\mathbf{x} - \mathbf{x}')^\top \mathbf{M} (\mathbf{x} - \mathbf{x}')) \quad (4)$$

is chosen as anisotropic kernel function. In this expression, the diagonal matrix $\mathbf{M} = \text{diag}(\frac{1}{2l_1^2} \dots \frac{1}{2l_p^2})$ contains the length scale parameters l_i that will be treated as hyperparameters $\boldsymbol{\theta} = (l_1 \dots l_p)$. The hyperparameters $\boldsymbol{\theta}$ are either defined based on prior knowledge or determined by maximizing the log marginal likelihood [10]

$$\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}) = -\frac{1}{2} \mathbf{y}^\top \mathbf{K}^{-1} \mathbf{y} - \frac{1}{2} \log |\mathbf{K}| - \frac{n}{2} \log 2\pi. \quad (5)$$

2.2 Polynomial chaos expansion (PCE)

A truncated polynomial expansion

$$\mathcal{M}^{\text{PCE}}(\tilde{\mathbf{X}}) = \sum_{\alpha \in \mathcal{A}} \beta_\alpha \psi_\alpha(\tilde{\mathbf{X}}) \quad (6)$$

is considered to approximate the function f where $\tilde{\mathbf{X}} = \{\tilde{X}_1 \dots \tilde{X}_p\}$ denotes the input variables as independent random variables with marginal probability density functions $\{f_{\tilde{X}_i}(x_i), i = 1 \dots p\}$. $\beta_\alpha \in \mathbb{R}$ are the expansion coefficients and $\psi_\alpha(\tilde{\mathbf{X}})$ are the multivariate polynomials with index α that identifies the polynomials in the finite set \mathcal{A} . The polynomials ψ_α are chosen

such that they are orthogonal with respect to the probability density function of $\tilde{\mathbf{X}}$. Since all random variables \tilde{X}_i are assumed to be uniformly distributed, the Legendre polynomials as the corresponding class of multivariate orthogonal polynomials are used.

The information matrix $\mathbf{A} \in \mathbb{R}^{n \times n_\alpha}$ with $A_{ij} = \psi_j(x_i)$ is obtained by evaluating all n_α polynomial basis functions for each sample, respectively. This matrix can be used for sample selection techniques as will be demonstrated later.

The focus of this contribution is on the case where computer model f is computationally expensive and therefore a limited number of model evaluations is available. A crucial point of polynomial chaos expansion is that the number of samples must be considerably (about 2 or 3 times [11]) greater than the number of polynomial basis functions. Therefore, a sparse polynomial chaos expansion is favorable. In this paper, least-angle regression [12] is used to find a sparse polynomial basis \mathcal{A} that yields an optimal fit of the sample data while maintaining a limited number of polynomials. The regression coefficients $\beta = \{\beta_\alpha, \alpha \in \mathcal{A}\}$ are calculated by a least squares fit according to

$$\beta = (\mathbf{A}^\top \mathbf{A})^{-1} \mathbf{A}^\top \mathbf{y}. \quad (7)$$

The approach of a sparse expansion is suitable especially for high dimensional problems where full and even truncated designs are problematic regarding the high number of polynomials.

2.3 Model validation

Validation of the surrogate models is conducted by a validation set. The validation set consists of samples from the input parameters distribution and corresponding model evaluations $\{(\mathbf{x}_{\text{val},i}, y_{\text{val},i}), i = 1, \dots, n_{\text{val}}\}$. The usage of a validation set requires a large amount of additional model evaluations, which is not appropriate for expensive models. However, in this work a validation set is used in order to achieve a more accurate surrogate model assessment. For this purpose, the relative mean square error

$$\varepsilon_{\text{RMSE}} = \frac{1}{\sigma_y^2 n_{\text{val}}} \sum_{i=1}^{n_{\text{val}}} (y_{\text{val},i} - \mathcal{M}(x_{\text{val},i}))^2 \quad (8)$$

is calculated, where σ_y^2 is the variance of the model evaluations

$$\sigma_y^2 = \frac{1}{n_{\text{val}}} \sum_{i=1}^{n_{\text{val}}} (y_{\text{val},i} - \mu_y)^2 \quad \text{with} \quad \mu_y = \frac{1}{n_{\text{val}}} \sum_{i=1}^{n_{\text{val}}} y_{\text{val},i}. \quad (9)$$

When using demonstrated methods for computationally expensive models, cross-validation techniques, such as leave-one-out cross-validation or k-fold cross-validation, may be used instead. So, no additional model runs are necessary. For polynomial chaos expansion the leave-one-out error can be calculated analytically from a single surrogate model [12].

3 SEQUENTIAL SAMPLING

3.1 Sampling procedure

In the beginning, a space-filling sampling method is applied in order to generate a sample set that is used to build the first surrogate model. In this work, a maximin Latin hypercube design

with a small number of samples n_0 is used. The term *maximin* refers to a design that aims to maximize the minimum distance between any two samples in order to improve space-filling properties [13].

Let n_i be the number of desired samples in the i -th sample set. Each sample within a sample set is added one at a time. After each added sample a new surrogate model is generated based on the previous sample set and the new sample \mathbf{x}_i . As evaluation $f(\mathbf{x}_i)$, the predicted value from the previous model $\mathcal{M}(\mathbf{x}_i)$ is taken. The goal is to find a new sample \mathbf{x}_i that improves the surrogate model based on certain criteria. The idea behind this procedure is known as *expected improvement* [7]. In this paper, the treatment of model parameters is emphasized, i.e. hyperparameters of Gaussian process regression and the polynomial basis of polynomial chaos expansion.

After each added sample within a sample set, no new information about the true model f is taken into account. It would be unreasonable to update the model parameters then. Therefore, they are assumed to remain unchanged in these steps. In case of Gaussian process regression, the hyperparameters remain unchanged. In case of polynomial chaos expansion, the polynomial basis remains unchanged.

After applying the space-filling design in the beginning and after each sample set, the function f is evaluated at all new n_i samples. A new surrogate model is built based on all available samples \mathbf{x}_i and evaluations f . In these steps, for Gaussian process regression the hyperparameters are updated by maximizing the log marginal likelihood (eq. 5) and for polynomial chaos expansion, the polynomial basis is redefined by least-angle regression. The process is stopped when a defined cross-validation condition is fulfilled or when the cost limit for the number of sample sets m is reached. The sampling procedure is illustrated in Algorithm 1.

3.2 Sample selection technique

Based on the current model $\mathcal{M}_{\text{guess}}$ (see Algorithm 1), a new sample \mathbf{x}_{new} is selected based on a selection technique that is available for the surrogate method.

3.2.1 Gaussian process regression

A new sample is usually desired at the point in the input space where the maximum prediction variance σ^2 (eq. 3) occurs. In order to find that point, a highly nonlinear optimization problem with usually many local maxima has to be solved. In this work, particle swarm optimization is used as it has shown good performance in such cases. [14]

However, these points do not yield the best model improvement in general. Here, the model improvement is assessed by the prediction variance over the whole input space. Especially in high dimensions, the greatest prediction variance often occurs on the boundary of the input space. The model improvement is therefore limited to one side in relation to the added samples. It is thus likely that other adjusted points yield a lower global prediction variance.

A new grid based selection technique is introduced as follows. A Cartesian grid \mathbf{X}^G with n_G grid points per input dimension is defined on the input space. The total number of grid points is n_G^p . These grid points are used to determine the prediction variance of the obtained surrogate models. The goal is to find a new sample \mathbf{x}_0 in the input space which yields the greatest mean prediction variance reduction of all grid points.

The surrogate model $\mathcal{M}_{\text{guess}}$ is evaluated at point \mathbf{x}_0 in the input space that will be determined by optimization. The prediction value at this point $\mathcal{M}_{\text{guess}}(\mathbf{x}_0)$ is used to construct a new surrogate

Algorithm 1 Sequential multi-point sampling procedure for Gaussian process regression and polynomial chaos expansion.

```

generate space-filling design  $\mathbf{X} = \{\mathbf{x}_i, i = 1 \dots n_0\}$ 
 $\mathbf{y} = f(\mathbf{X})$ 
build surrogate model  $\mathcal{M}$  based on  $\{\mathbf{X}, \mathbf{y}\}$ 
    ↳ store hyperparameters  $\theta$  (GPR) / polynomial basis  $\mathcal{A}$  (PCE)
for sample set  $i = 1, \dots, m$ :
    initialize  $\mathbf{X}_{\text{guess}} = \{\}$ ,  $\mathbf{y}_{\text{guess}} = \{\}$ ,  $\mathcal{M}_{\text{guess}} = \mathcal{M}$ 
    for sample point  $j = 1, \dots, n_i$ :
        find  $\mathbf{x}_{\text{new}}$  based on selection technique for model  $\mathcal{M}_{\text{guess}}$ 
        append  $\mathbf{x}_{\text{new}}$  to  $\mathbf{X}_{\text{guess}}$ 
        if  $j < n_i$ :
            append  $\mathcal{M}(\mathbf{x}_{\text{new}})$  to  $\mathbf{y}_{\text{guess}}$ 
            build new surrogate model  $\mathcal{M}_{\text{guess}}$  based on  $\{(\mathbf{X}, \mathbf{X}_{\text{guess}}), (\mathbf{y}, \mathbf{y}_{\text{guess}})\}$ 
                ↳ use previous hyperparameters  $\theta$  (GPR) / polynomial basis  $\mathcal{A}$  (PCE)
        if  $j = n_i$ :
            append  $\mathbf{X}_{\text{guess}}$  to  $\mathbf{X}$ 
            append  $f(\mathbf{X}_{\text{guess}})$  to  $\mathbf{y}$ 
            build new surrogate model  $\mathcal{M}$  based on  $\{\mathbf{X}, \mathbf{y}\}$ 
                ↳ update hyperparameters  $\theta$  (GPR) / polynomial basis  $\mathcal{A}$  (PCE)
    validate model  $\mathcal{M}$ 
    break if validation criterion is reached
    
```

model \mathcal{M}_0 . The model quality of \mathcal{M}_0 is assessed based on grid \mathbf{X}^G . The mean prediction variance σ^2 of \mathcal{M}_0 over all grid points \mathbf{X}^G is used as objective function to be minimized with respect to \mathbf{x}_0 . Again, this is a highly nonlinear optimization problem. Therefore, particle swarm optimization is used to find \mathbf{x}_0 . Since a new surrogate model \mathcal{M}_0 has to be built in each iteration of the optimization, the computation cost is considerably higher compared to the conventional method. However, since hyperparameters θ remain unchanged in the optimization process, the cost can be clearly reduced. In case of expensive functions f , this effort may still be worthwhile. The method is summarized in Algorithm 2.

If modified importance should be assigned to certain regions of the input space or in case of nonuniform input distributions, a weight function with respect to \mathbf{x} may be multiplied to the prediction variance values for all grid points. Since this is not the case here, i. e. uniform distributions are assumed, this will not be elaborated further.

3.2.2 Polynomial chaos expansion

Sequential sampling methods for polynomial chaos expansion are described in [11]. These methods incorporate information matrix \mathbf{A} (section 2.2) to find new samples according to an optimality criterion. D-optimal sampling aims at maximizing the determinant $D(\mathbf{A}) = \det(\frac{1}{n}\mathbf{A}^\top \mathbf{A})^{\frac{1}{n\alpha}}$. This is related to minimizing the variance of the PCE coefficients β_α (eq. 6).

Algorithm 2 Grid-based sample selection technique for Gaussian process regression.

Input: model $\mathcal{M}_{\text{guess}}^{\text{GPR}}, \{(\mathbf{X}, \mathbf{X}_{\text{guess}}), (\mathbf{y}, \mathbf{y}_{\text{guess}})\}$, hyperparameters θ
 generate Cartesian grid $\mathbf{X}^G = \{\mathbf{x}_1^G \dots \mathbf{x}_{n_G^p}^G\}$ over input space
 with n_G grid points per input dimension (p dimensions)
minimize σ_m^2 for $\mathbf{x}_0 \in [0, 1]^p$:
 | $y_0 = \mathcal{M}_{\text{guess}}^{\text{GPR}}(\mathbf{x}_0)$
 | build surrogate model $\mathcal{M}_0^{\text{GPR}}$ based on $\{(\mathbf{X}, \mathbf{X}_{\text{guess}}, \mathbf{x}_0), (\mathbf{y}, \mathbf{y}_{\text{guess}}, y_0)\}$
 | \leftarrow use hyperparameters θ
 | $\sigma_m^2 = \frac{1}{n_G^p} \sum_{i=1}^{n_G^p} \sigma^2(\mathbf{x}_i^G)$ (mean prediction variance of \mathcal{M}_0)
 | **return** σ_m^2
Output: \mathbf{x}_0 for minimum σ_m^2

In this work, S-optimal sampling is used which aims at maximizing the S-value

$$S(\mathbf{A}) = \left(\frac{\sqrt{\det(\mathbf{A}^\top \mathbf{A})}}{\prod_{i=1}^{n_\alpha} \|A_i\|_2} \right)^{\frac{1}{n_\alpha}}. \quad (10)$$

While maximizing the determinant, the S-value additionally aims at maximizing the column orthogonality of \mathbf{A} . Here, A_i denotes the i -th column of information matrix \mathbf{A} . As denoted in Algorithm 1, polynomial basis \mathcal{A} remains unchanged between added samples within one sample set. Therefore, the column size of \mathbf{A} does not change, but one row is appended with each added sample containing corresponding polynomial basis evaluations. Hence, least-angle regression is not conducted in these steps and computation cost is relatively small.

As comparative study, new samples are selected based on a conventional distance-based measure. Here, a new sample

$$\mathbf{x}_{\text{new}} = \underset{\mathbf{x}_{\text{new}} \in [0, 1]^p}{\operatorname{argmax}} \left(\min_{i \in \{1 \dots n\}} \|\mathbf{x}_i - \mathbf{x}_{\text{new}}\|_2 \right) \quad (11)$$

is added at the point in the input space that maximizes the minimum euclidean distance to existing points $\{\mathbf{x}_1 \dots \mathbf{x}_n\}$ in the design. This procedure is known as *farthest point strategy* [2].

4 TEST CASE

A modified version of the Ishigami function

$$f(\mathbf{x}) = \sin(x_1) + 3 \sin^2(x_2) + 2x_2^4 \sin(x_1) \quad (12)$$

is chosen to investigate the sampling methods according to Algorithm 1. The modification is done in order to obtain a two-dimensional function so that sampling methods can be visualized graphically.

First, a maximin Latin hypercube design with $n_0 = 10$ samples is generated. Then, five sample sets are added, each with two samples ($n_i = 2, i = 1 \dots 5$). In case of Gaussian process regression, initially, the method of selecting samples at maximum prediction variance is applied. In another experiment, the proposed grid-based method according to Algorithm 2 is applied.

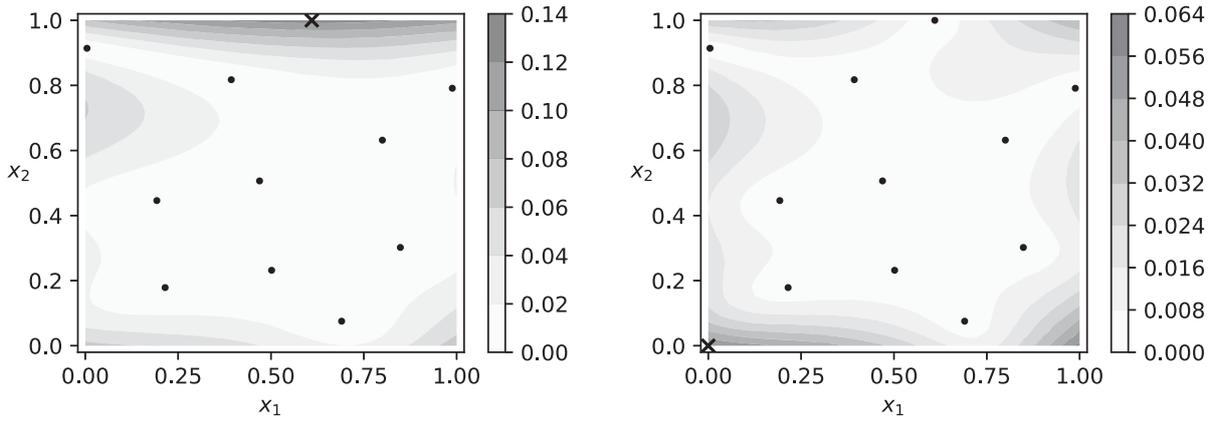


Figure 1: Gaussian process regression: prediction variance σ^2 (eq. 3) with respect to inputs x_1 and x_2 before adding the first sample (left) and the second sample (right) of the first sample set. Existing samples are marked as dots, added samples are marked as crosses.

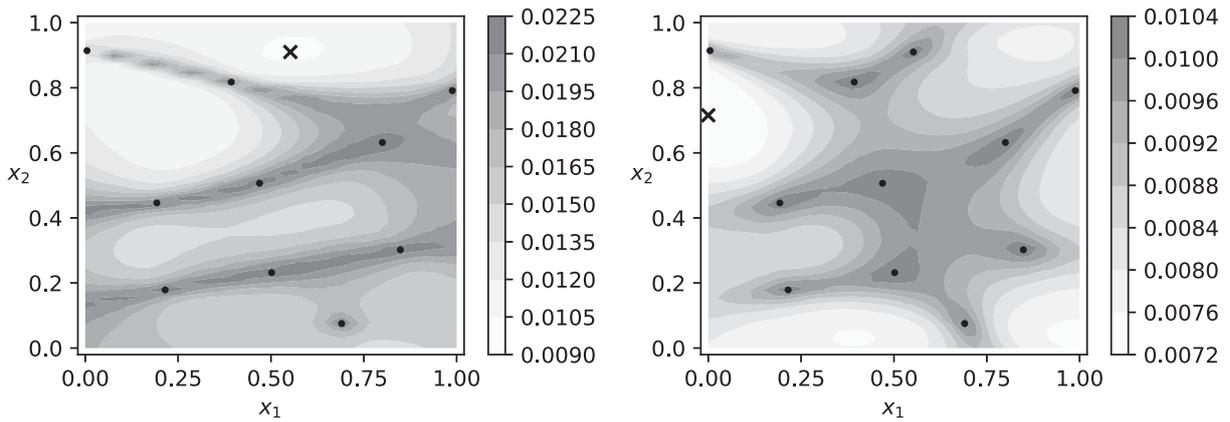


Figure 2: Gaussian process regression: mean prediction variance σ_m^2 (Algorithm 2) from the grid-based method with respect to inputs x_1 and x_2 before adding the first sample (left) and the second sample (right) of the first sample set. Existing samples are marked as dots, added samples are marked as crosses.

In case of polynomial chaos expansion, the distance-based criterion (eq. 11) and S-optimal sampling (eq. 10) are applied to select new samples. In all cases, particle swarm optimization is used to find such points in the input space. The obtained models \mathcal{M} are validated after each sample set.

For comparison, new samples are generated by standard Monte Carlo sampling using the same number of samples per set $n_i = 2, i = 1 \dots 5$.

In Figures 1, 2, 3 and 4 the locations of $n_1 = 2$ added samples in the first sample set are shown for Gaussian process regression and polynomial chaos expansion for chosen sampling methods, respectively. The same space-filling design is used to allow for better comparison. In Figure 1, the contour plot indicates prediction variance σ^2 of model $\mathcal{M}_{\text{guess}}^{\text{GPR}}$ that is used for sample selection. Points with maximum prediction variance after each step are selected as new samples. In Figure 2, the contour plot shows the expected mean prediction variance σ_m^2 (see

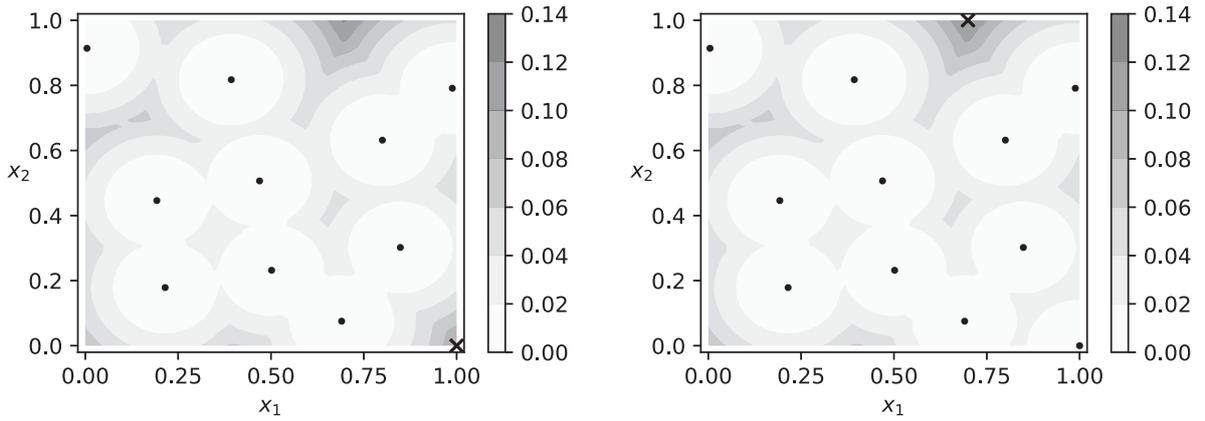


Figure 3: Polynomial chaos expansion: distance-based criterion (eq. 11) for new samples with respect to inputs x_1 and x_2 before adding the first sample (left) and the second sample (right) of the first sample set. Existing samples are marked as dots, added samples are marked as crosses.

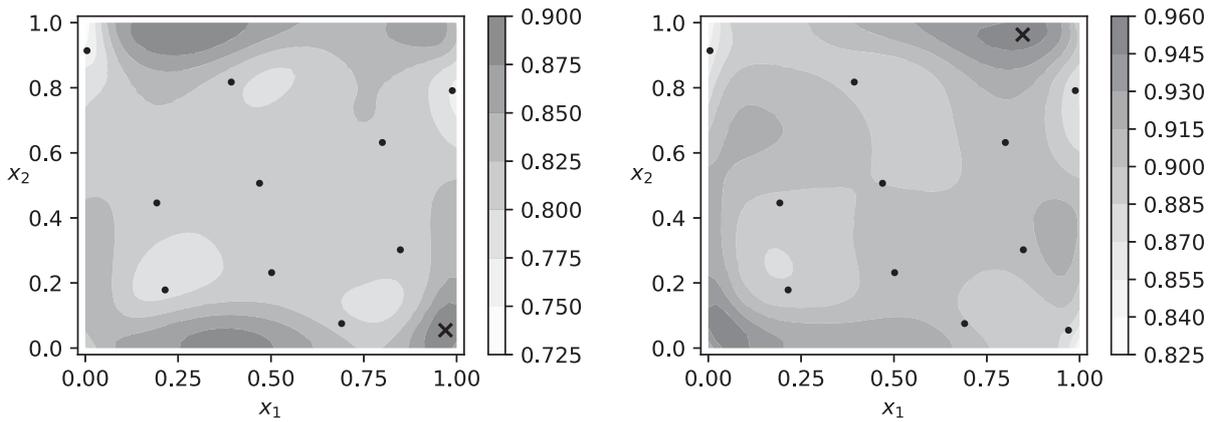


Figure 4: Polynomial chaos expansion: S-value (eq. 10) with respect to inputs x_1 and x_2 before adding the first sample (left) and the second sample (right) of the first sample set. Existing samples are marked as dots, added samples are marked as crosses.

Algorithm 2) that would be expected to result from a new sample at respective points in the input space. Points with the lowest values of σ_m^2 are selected as new samples. In Figure 3, the contour plot indicates the euclidean distance of points in the input space to the closest existing sample. New samples are chosen that maximize this distance. In Figure 4, the contour plot indicates the S-value (eq. 10) that is maximized to select new samples.

In all demonstrated cases, the selection of new samples shows to ensure good space-filling properties. For Gaussian process regression, it can be recognized that new samples are less likely to occur on the boundary, if the proposed grid-based selection method is applied.

In Figure 5 the validation error $\varepsilon_{\text{RMSE}}$ (eq. 8) is illustrated after each sample set for all investigated sampling methods. The methods are compared to the case where new samples are selected according to standard Monte Carlo sampling. It is distinct that sample selection methods (Algorithm 1) are superior compared to Monte Carlo sampling. In the considered test case, Gaussian

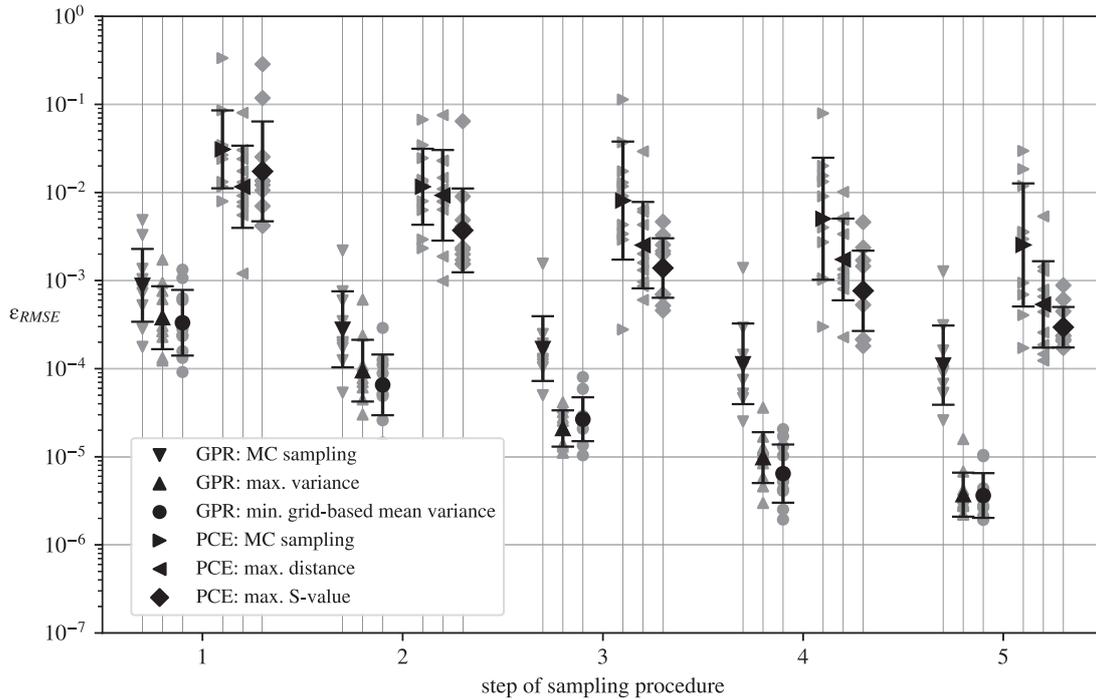


Figure 5: Validation error $\varepsilon_{\text{RMSE}}$ for different sampling methods (vertical lines) after each of five sample sets (horizontal axis). Grey markers show the results for single experiments beginning with new space-filling designs, while black markers show their mean values with standard deviations.

process regression yields better results than polynomial chaos expansion. This is due to the fact that the number of polynomial basis functions is very limited regarding the small number of samples in order to avoid over-fitting. In this work, the maximum number of basis functions for least-angle regression is updated after each sample set to half of the current sample size. In addition, the maximum polynomial degree is increased during the sampling procedure.

For Gaussian process regression, the proposed grid-based method yields only vaguely smaller validation errors compared to the conventional sample selection method for Gaussian process regression. However, based on the sample selection technique it is presumed that the global prediction variance reduction over the whole input space is improved through the proposed grid-based method.

For polynomial chaos expansion, the S-optimality criterion yields slightly smaller errors compared to the distance-based measure. However, it is emphasized, that the polynomial basis is updated after each sample set. Thus, the optimality criterion may change in such a way that the chosen samples are not optimal anymore with regard to the new basis. The distance-based measure has shown to be more robust in very sparse designs (i. e. less than ten samples) and thus for a small number of basis functions.

5 CONCLUSIONS

Studies on a simple example have shown that proposed multi-point sequential sampling methods are very promising compared to standard Monte Carlo sampling and should be taken

into consideration, especially if computationally intensive models are investigated and the necessary sample size for the desired surrogate model quality is not known in advance. For sparse designs such as in the considered test case, Gaussian process regression appears to be a superior regression tool compared to polynomial chaos expansion. This is due to great limitations of polynomial chaos expansion regarding the number and order of basis functions for sparse designs.

Even though no significant improvement was obtained by using the proposed grid-based sampling technique, it may be worthy to further investigate this method if surrogate model quality demands are high. The additional computational effort may still be small if computationally expensive models are investigated. However, it becomes apparent that the conventional sample selection method for Gaussian process regression, namely to search for points in the input space with maximum prediction variance, is a sufficient and computationally affordable tool in general. Instead of using a fixed grid, other possibilities may be reasonable. For example, a Latin hypercube design may be used as grid to introduce randomness. The Latin hypercube design may then be randomly updated after each added sample, so that biases originating from fixed grid points may be prevented.

The distance-based measure has shown to be a robust and successful tool that can be applied for other surrogate models as it only considers the input space. If further improvement is desired for polynomial chaos expansion, optimality criteria such as S-optimality may be applied.

REFERENCES

- [1] B. Settles. Active learning literature survey. Computer sciences technical report 1648. *University of Wisconsin-Madison*, 2009.
- [2] Y. Eldar, M. Lindenbaum, M. Porat, Y. Y. Zeevi. The farthest point strategy for progressive image sampling. *IEEE Transactions on Image Processing*, **6**, 1305–1315, 1997.
- [3] P. Z. G. Qian, Nested Latin hypercube designs, *Biometrika*, **96**, 957–970, 2009.
- [4] S. Seo, M. Wallat, T. Graepel, K. Obermayer. Gaussian process regression: active data selection and test point rejection. *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*, **3**, 241–246, 2000.
- [5] D. Ginsbourger, R. L. Riche, L. Carraro. A Multi-points Criterion for Deterministic Parallel Global Optimization based on Gaussian Processes. hal-00260579, 2008.
- [6] S. Marelli, B. Sudret. An active-learning algorithm that combines sparse polynomial chaos expansions and bootstrap for structural reliability analysis. *Structural Safety*, **75**, 67–74, 2018.
- [7] R.T. Haftka, D. Villanueva, A. Chaudhuri. Parallel surrogate-assisted global optimization with expensive functions – a survey. *Structural and Multidisciplinary Optimization*, **54**, 3–13, 2016.
- [8] A. J. Booker, J. E. Dennis, P. D. Frank et al. A rigorous framework for optimization of expensive functions by surrogates. *Structural Optimization* **17**, 1–13, 1999.

- [9] N. Lüthen, S. Marelli, B. Sudret. Sparse polynomial chaos expansions: Literature survey and benchmark. arXiv preprint arXiv:2002.01290, 2020.
- [10] C. E. Rasmussen, C. K. I. Williams. Gaussian Processes for Machine Learning. The MIT Press. 2005.
- [11] G. Blatman, Adaptive sparse polynomial chaos expansions for uncertainty propagation and sensitivity analysis, Université Blaise Pascal, Clermont-Ferrand, France, 2009.
- [12] G. Blatman, B. Sudret. Adaptive sparse polynomial chaos expansion based on least angle regression, *Journal of Computational Physics*, **230**, 2345–2367, 2011.
- [13] M.E. Johnson, L.M. Moore, D. Ylvisaker. Minimax and maximin distance designs. *Journal of Statistical Planning and Inference*, **26**, 131–148, 1990.
- [14] J. Kennedy, R. Eberhart. Particle swarm optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*, **4**, 1942–1948, 1995.

CALIBRATION OF MATERIAL MODEL PARAMETERS USING MIXED-EFFECTS MODELS

Clément LABOULFIE¹, Mathieu BALESSENT², Loïc BREVAULT², Sebastien DA
VEIGA³, François-Xavier IRISARRI¹, Rodolphe LE RICHE⁴, Jean-François MAIRE¹

¹DMAS, ONERA, Université Paris Saclay
F-92322, Châtillon FRANCE
e-mail: clement.laboulfie, francois-xavier.irisarri, jean-francois.maire@onera.fr

² DTIS, ONERA, Université Paris Saclay
F-91123, Palaiseau FRANCE
e-mail: mathieu.balesdent, loic.brevault@onera.fr

³ Safran TECH
Rue des jeunes Bois, 78117 Châteaufort FRANCE
sebastien.da-veiga@safrangroup.com

⁴ CNRS LIMOS
Mines Saint-Étienne and UCA, 158 cours Fauriel, 42100 Saint-Étienne, FRANCE
leriche@emse.fr

Keywords: Model Calibration, Uncertainty Quantification, Mixed-effects models, Composite

Abstract. *The quantification of model parameter uncertainty is a long-standing issue in model calibration. Classical techniques provide methods to handle some type of uncertainties (e.g. experimental noise or model bias). However, usual calibration techniques are not designed to take into account the variability between the different individuals. This is not a problem if the individual variability is negligible but it is an important issue if the individual variability is significant. The mixed-effects models provide a statistical framework to calibrate the parameters of a model taking into account the individual variability. The objective of this paper is to introduce the mixed-effects in material science. The ONERA Damage model (ODM) is considered, first with synthetic data, then with thirteen experimental strain-stress curves of a ceramic matrix composite material. The robustness of the mixed-effects approach regarding the variability and the number of specimen is investigated. Model choices such as the correlation between ODM parameters and other settings are discussed. The ability of mixed-effects models to characterize the material variability and to provide accurate estimates of the parameters associated to each specimen is illustrated.*

1 GENERAL INTRODUCTION

Model calibration is an active research topic which is common to all scientific domains such as hydrology [1], economics [2], biology [3] and mechanics [4]. The goal of calibration may be described as follows: for a given set of experimental observations and a model whose predictions are controlled by parameters, find the model parameters values that provide the best adequation between the model responses and the observations. Most of the time, the observations are noisy, the models may not be able to reproduce perfectly the observations (model bias), the specimens are subject to variability (due to the production process). As a consequence, the inferred model parameters are uncertain. Characterizing the uncertainty of the calibrated model parameters is essential before carrying out further analyses (uncertainty propagation, sensitivity analysis, *etc.*).

In the field of mechanics, for a given experiment (a tensile test for instance) and a given material, international standards impose to repeat the tests on different specimens of this material ([5] in aeronautics) to quantify the effects of material variability on the mechanical properties (*e.g.*, the Young's modulus, the ultimate tensile strength). As a consequence, databases used to characterize materials are often composed of the results of an experiment on different specimens of the same material. Taking into account experimental variability in the calibration process is necessary when the model parameters are sensitive to them.

A large number of calibration procedures exist which are commonly divided into the frequentist [6, 7] and the Bayesian methods [8, 9]. The frequentist approach consists in minimizing over the model parameters a misfit function [6] which quantifies a difference between the model output and the data. The least-square criterion and other L_p -norms [7, 10, 11] are usual misfit criteria. In [12], other criteria such as the weighted least squares are proposed. The results of the calibration depend on the chosen criterion and on the optimization algorithm. Another type of criterion commonly used is the maximum likelihood estimator (MLE) [9, 13, 14]. For a given statistical hypothesis over the misfit (*e.g.*, the misfit follows a Gaussian distribution), the likelihood measures how well the model output matches the data. The frequentist framework also provides methods to handle the calibration of multiple experiments assuming a single value for the input model parameters [15]. This scenario can arise if the database is composed of repetitions of a tensile test. In this case, the criterion to be minimized is a vector containing the scalar misfit criteria associated to each repetition. To solve such multi-objective optimization problem, either the problem is transformed into a single objective problem (for instance by a weighted sum of the objectives) or a multi-objective algorithm is used [16]. In the multi-objective formulation, the result of the calibration is a set of parameters values called the Pareto frontier which contains all the tradeoffs between the different misfit criteria. This set is made of the parameters which are consistent in the Pareto sense with respect to the different repetitions. In the frequentist framework, irrespectively of the number of misfit criteria, model parameters are assumed to be deterministic. However, because of the presence of experimental noises and specimens variability, the model parameters should be considered as uncertain. The frequentist framework provides methods (such as the asymptotic theory [2, 6, 11] and bootstrap [9, 17, 18]) to quantify model parameter uncertainties but they are mainly dedicated to characterize the experimental noise, not the material variability.

In the Bayesian framework [6, 9], the parameters are considered as random variables. The aim of the Bayesian inference is to get a description of their probability density function (PDF).

It relies on an assumed prior density and on a likelihood function. The prior density sums up the available knowledge before calibration. The choice of the prior may be difficult and has an impact on the calibration results. As in the frequentist framework, the likelihood function expresses the goodness of fit of a given set of model parameters. The result of Bayesian inference is a PDF of the model parameters called the posterior distribution or a set of samples of it. This PDF combines information from the prior density, the likelihood and the available data [6, 8]. Most of the time, the posterior distribution cannot be computed analytically. In case no analytical expression of the posterior distribution is available, Monte Carlo Markov Chain (MCMC) methods allow to generate samples from this posterior density [19, 20]. The MCMCs build a random walk in the parameters space which selects samples of the posterior density[19]. The properties of the random walk ensure that the approximate posterior distribution converges to the exact posterior distribution as the number of iterations goes to infinity. Among the MCMC algorithms, one of the most popular algorithms is the Metropolis-Hastings algorithm [21, 22]. In practice, it may be challenging to define the settings of MCMC algorithms and to ensure an appropriate convergence to the exact posterior distribution.

Both frequentist and Bayesian frameworks rely on the likelihood function which expresses the goodness of fit of the model response to the data given a set of model parameters. In the material engineering literature [23, 24], both approaches make the same hypothesis towards material variability. Indeed, in the derivation of the likelihood function, it is assumed that all the specimens can be described by a unique set of model parameters. If the material variability can be neglected, this assumption is correct and allows to calibrate a single specimen. Under such an hypothesis, material model calibration has been carried out using both frequentist and Bayesian approaches. For instance, Avril *et al.* [23] gave examples of mechanically suited criteria to be minimized in a frequentist approach like the Finite Element Method Updating approach or the Constitutive Equation Gap Method. Within the frequentist framework again, Chongshuai *et al.* [25] calibrated a visco-elastic model and Solanki *et al.* a non-linear damage model [26]. Bayesian inference has long been applied to mechanical problems. Isenberg presented in [24] the calibration of elastic properties. Gogu *et al.* compared results from both frequentist and Bayesian approaches [27]. Gogu *et al.* later proposed the identification of elastic properties using Finite Element Model for composite materials [28]. Non-linear models can also be calibrated thanks to Bayesian inference. Liu & Au [29] calibrated a non-linear hysteretic model, Rizzi *et al.* [30] proposed the identification of a finite element plasticity model and Rappel *et al.* calibrated visco-elastic models [31]. Rappel *et al.* [4] also proposed a tutorial to Bayesian inference for different mechanical models among which elastic linear, linear elasticity-perfect plasticity models and viscoelastic models.

When material variability is significant, the specimens should no longer be described by a unique set of model parameters. It is possible to calibrate individually each specimen, but the parameter vector inferred on one specimen is not necessarily consistent with the observations of the other specimens. In fact, the simple likelihood function is not designed to take into account material variability. The result of this calibration consists of a set of parameter vectors, one for each calibrated specimen. Yet, these parameter vectors are somehow related as all of them characterize the same material on which has been carried out the same experiment. Rizzi *et al.* [30] proposed a way to take into account material variability in the calibration process. They made the hypothesis that for all the available specimens, the model parameters vary between a lower and upper bound (defined by expert knowledge through mathematical

constraints). This is equivalent to assuming that, for all the specimens, the model parameters are uniformly distributed between those bounds and thus can be approximated with a first order Legendre uniform polynomial chaos expansion (PCE). The objective was to find the coefficients of the PCE for each model parameter. This approach is useful as the calibration process tries to take explicitly into account the material variability. However, it does not estimate the parameters of every specimen which are necessary for physical interpretations. To sum up, none of the previous methods allows to calibrate a material model parameter taking into account both experiment repetitions and material variability.

The aim of this paper is to propose a calibration method compliant with specimens repetition (defined in the following as a population) in the presence of material variability. The approach relies on mixed-effects models [32]. We seek to demonstrate the ability of the method to both describe the individual variability and to provide estimates of the individual parameters. In Section 2.1, the mixed-effects approach is presented. Section 2.3 gives two different methods to estimate the likelihood function. Then, this new methodology is applied for the first time to a simplified damage model derived from a material model for ceramic matrix composite materials (called ONERA Damage Model (ODM) [33, 34, 35]) with four parameters to be calibrated. In Section 3.3, the method is tested on virtual data to study the calibration methodology settings and the consequences of the mixed-effects. Then, in Section 3.4, thirteen individual replicate tensile experiments (that each contains a dozen to a hundred observations) are processed with the method. The specimens are made of CERASEP A400 [36, 37], a ceramic-matrix composite material.

2 POPULATION-BASED APPROACH AND MIXED-EFFECTS MODELS

The mixed-effects models and the population-based approaches are now described. The models are first presented under a general scope to illustrate how this point of view can help to analyze the material variability (in section 2.1). Then, the models are detailed and specified for the calibration of a material.

2.1 Introduction to the population-based and the mixed-effects models

Population-based models attempt to describe the variability of physical phenomena observed in a population of individuals. This type of approaches finds its origin in pharmacometrics [38, 39] where it is important to quantitatively describe interactions between drugs, diseases and patients. For instance, during a drug test, every subject is given the same amount of drug but his response heavily depends on his genome [32, 39]. The mixed-effects notion comes from the fact that there are “fixed” effects that are shared by the entire population of individuals and “random” effects that are specific to each individual of the population. For a given model of interest, the specimen parameters (also referred as the individual parameters) can be decomposed as

$$\text{individual parameters} = \text{fixed effects} + \text{random effects} \quad (1)$$

In a mechanical context, if the studied parameter is the Young’s modulus, Eq.(1) states that the Young’s modulus of each specimen is the combination of a reference value (for instance given by the producer) and of a deviation due to the production process. Depending on the relationship between the input parameters and the responses, linear [40, 41, 42] and non-linear [43] mixed-effects models have been developed.

In Figure 1, the differences between the classical and the population-based approach are highlighted. On the left of the Figure, in the classical approach, the underlying hypothesis is that

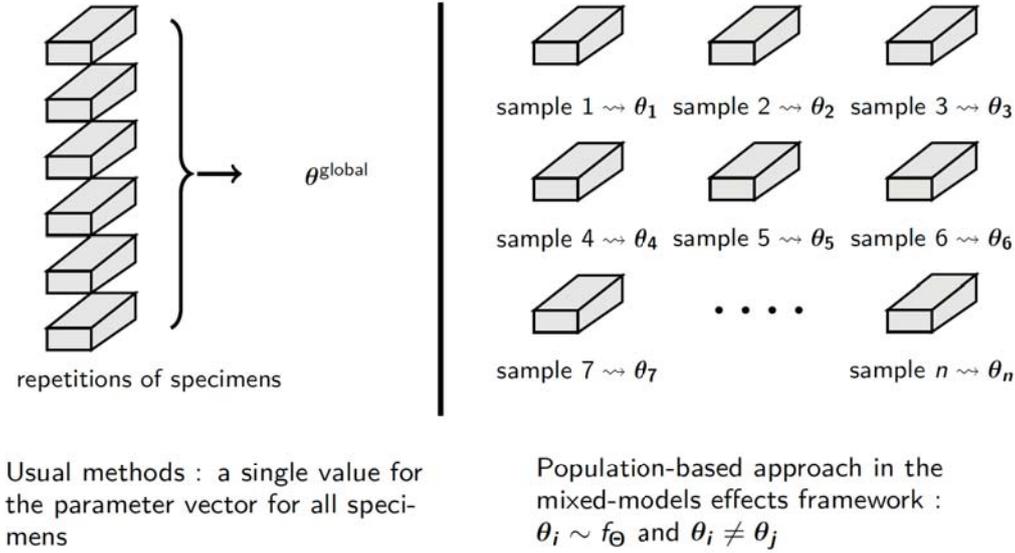


Figure 1: Comparisons of the assumptions made with respect to the individual variability in the classical approach (left) and in the population-based approach (right).

all the specimens can be described by a unique set of parameters. In the population-based approach (on the right), each sample is assigned a set of parameters. In the mixed-effects models, it is assumed that there exists an underlying probability distribution (noted f_{Θ}) whose outcomes are the individual parameters set $\theta_1, \theta_2, \dots, \theta_n$ with n the number of individuals [32, 44]. This is an improvement from the classical approach as it establishes a relation between the individual parameters set. In this framework, both the underlying probability distribution of the model parameters and the value of the individual parameters are determined. In addition, it remains possible to include other sources of uncertainty (experimental noise, model bias, *etc.*) on the individual parameters.

Thanks to its ability to describe individual variability, the population-based approach is used in cases where this variability is important. To the best of our knowledge, this approach has never been applied to material models to account for the variability introduced by the repetition of mechanical tests over a population of specimens.

2.2 Formalization

We now describe the mixed-effects approach of [32, 44] in the context of tensile tests performed on different specimens. All the specimens are samples from the same material, for example SiC-SiC composites CERASEP A400. They share the same dimensions and features. There are n specimens and $i \in \llbracket 1, n \rrbracket$ is the corresponding index. The number of observations the i^{th} specimen (or individual) is N_i and j stands for the index of the j^{th} measure (with $j \in \llbracket 1, N_i \rrbracket$). The j^{th} output measure of the i^{th} individual is labeled as y_{ij} and t_{ij} stands for the j^{th} input measure of the i^{th} individual. The random vector of the output measures of the i^{th} individual is written Y_i and its outcome $\mathbf{y}_i = (y_{ij})_{j \in \llbracket 1, N_i \rrbracket}$. The set of parameters of the i^{th} individual is denoted $\theta_i = (\theta_i^1, \theta_i^2, \dots, \theta_i^d) \in \mathbb{R}^d$, with d the number of model parameters to be calibrated. The model of the material is noted $\mathcal{F}(\cdot)$. PDFs will be noted by the generic letter f .

2.2.1 Mixed-effects models approach

The mixed-effects framework [32, 44] assumes that there exists a probability distribution f_{Θ} whose outcomes are the individual parameters:

$$\forall i \in \llbracket 1, n \rrbracket, \theta_i \underset{\text{i.i.d.}}{\sim} f_{\Theta} \quad (2)$$

where i.i.d. stands for independent and identically distributed. Both f_{Θ} and the θ_i are unknown and the aim is to determining them. In addition, if f_{Θ} is parametric (Gaussian distribution for instance), Π stands for its parameters and $f_{\Theta} = f_{\Theta, \Pi}$. Identifying $f_{\Theta, \Pi}$ is tantamount to determining Π . Given Π and $\theta_i \sim f_{\Theta, \Pi} \forall i \in \llbracket 1, n \rrbracket$, the model output y_i can be written as:

$$\forall i \in \llbracket 1, n \rrbracket, y_i \underset{\text{i.i.d.}}{\sim} \mathcal{F}(\cdot, \theta_i) + \xi \quad (3)$$

In Eq. (3), ξ stands for the random vector of the errors. It represents the experimental noise and the model bias. Without any other hypothesis, the outcomes of ξ labeled $(\xi_{ij})_{(i,j) \in \llbracket 1, n \rrbracket \times \llbracket 1, N_i \rrbracket}$ are different for each individual and for each observation. The global mixed-effects models for the j^{th} output measure of the i^{th} individual y_{ij} reads:

$$y_{ij} = \mathcal{F}(t_{ij}, \theta_i) + \xi_{ij} \quad (4)$$

Classically, several additional hypotheses are assumed. First, $f_{\Theta, \Pi}$ is chosen to be a multivariate Gaussian distribution (of dimension d):

$$f_{\Theta, \Pi} = \mathcal{N}(\boldsymbol{\mu}, \Sigma) \quad (5)$$

with $\boldsymbol{\mu} \in \mathbb{R}^d$ the mean vector and $\Sigma \in \mathcal{M}_d(\mathbb{R})$ the covariance matrix. This choice has to be made taking into account expert knowledge, experimental and physical problem characteristics. As a consequence, individual parameters can be written as follows

$$\theta_i = \boldsymbol{\mu} + \mathbf{b}_i \quad (6)$$

with $\mathbf{b}_i \underset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \Sigma)$. Comparing Eqs.(1) and (6), $\boldsymbol{\mu}$ stands for the fixed effects (the same for the whole population) and \mathbf{b}_i the random effects (different for each individual). The second hypothesis is that for each individual and each measure, the error term is a Gaussian white noise (no bias, no correlation):

$$\xi_{ij} \underset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \omega_{ij}) \quad (7)$$

with ω_{ij} the variance of the noise of the j^{th} output measure of the i^{th} individual. The noise further is supposed to be homoscedastic, that is to say $\omega_{ij} = \omega_i \forall (i, j) \in \llbracket 1, n \rrbracket \times \llbracket 1, N_i \rrbracket$. Finally, the vector of parameters to be calibrated is denoted Ψ :

$$\Psi = (\boldsymbol{\mu}, \Sigma, \Omega) \in \mathbb{R}^{n+d+\frac{d(d+1)}{2}}$$

with $\Omega = \text{diag}(\omega_1, \dots, \omega_n)$. The mixed-effects models seek Ψ and gives an estimate of the individual parameters $(\theta_i)_{i \in \llbracket 1, n \rrbracket}$.

2.2.2 Likelihood of mixed-effects models

The calibration is often achieved by maximizing the likelihood of Ψ , $\mathcal{L}(\Psi) = f(\mathbf{y}_1, \dots, \mathbf{y}_n | \Psi)$, even if other methods can be found to estimate Ψ [45].

The first step consists in writing the PDF of output measurements for a given set of individual parameters and error term. Combining the Eqs.(4) and (7), this density is expressed as:

$$f(\mathbf{y}_i | \boldsymbol{\theta}_i, \omega_i) = \frac{1}{(\omega_i \sqrt{2\pi})^{N_i}} \prod_{j=1}^{N_i} e^{-\frac{1}{2} \left(\frac{y_{ij} - \mathcal{F}(t_{ij}, \boldsymbol{\theta}_i)}{\omega_i} \right)^2} \quad (8)$$

The individual parameters are distributed according to a Gaussian PDF (Eq.(5)). For a given Ψ , their density can be written as follows:

$$f(\boldsymbol{\theta}_i | \Psi) = \frac{1}{\sqrt{|\Sigma|} (2\pi)^d} e^{-\frac{1}{2} (\boldsymbol{\theta}_i - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{\theta}_i - \boldsymbol{\mu})} \quad (9)$$

Following the use of the conditional probability rule, the PDF of the individual parameters $\boldsymbol{\theta}_i$ and model output \mathbf{y}_i for a given Ψ , that is to say $f(\mathbf{y}_i, \boldsymbol{\theta}_i | \Psi)$, is:

$$f(\mathbf{y}_i, \boldsymbol{\theta}_i | \Psi) = f(\mathbf{y}_i | \boldsymbol{\theta}_i, \Psi) f(\boldsymbol{\theta}_i | \Psi) \quad (10)$$

Using Eqs.(8), (9) and (10), $f(\mathbf{y}_i, \boldsymbol{\theta}_i | \Psi)$ can be written as :

$$f(\mathbf{y}_i, \boldsymbol{\theta}_i | \Psi) = \frac{1}{\omega_i^{N_i} \sqrt{|\Sigma|} (2\pi)^{N_i+d}} e^{-\frac{1}{2} \left((\boldsymbol{\theta}_i - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{\theta}_i - \boldsymbol{\mu}) + \sum_{j=1}^{N_i} \left(\frac{y_{ij} - \mathcal{F}(t_{ij}, \boldsymbol{\theta}_i)}{\omega_i} \right)^2 \right)} \quad (11)$$

Because the $\boldsymbol{\theta}_i$'s are not observed, The likelihood of the i^{th} individual $\mathcal{L}_i(\Psi)$ is the integral of $f(\mathbf{y}_i, \boldsymbol{\theta}_i | \Psi)$ with respect to all possible $\boldsymbol{\theta}_i$ over \mathbb{R}^d :

$$\mathcal{L}_i(\Psi) = \int_{\mathbb{R}^d} f(\mathbf{y}_i, \boldsymbol{\theta}_i | \Psi) d\boldsymbol{\theta}_i \quad (12)$$

Under the assumption of independent individuals, the likelihood of Ψ reads as the product of all the individual likelihoods,

$$\mathcal{L}(\Psi) = \prod_{i=1}^n \mathcal{L}_i(\Psi) \quad (13)$$

The maximum likelihood estimator $\hat{\Psi}$ is obtained as the result of the following maximization problem over the set of all the possible parameters Ξ :

$$\hat{\Psi} = \arg \max_{\Psi \in \Xi} \mathcal{L}(\Psi) \quad (14)$$

In practice, the log-likelihood is computed to ease numerical optimization.

Mixed-effects models approach consists in calibrating a (multivariate) probability distribution. To do so, a minimum number of samples is required. However, in the material field, the number of repetitions of a given experiment is small (in the order of 10 for some tests, see [5] for tensile tests). This has to be confronted with the number of parameters to be calibrated which is here in $\mathcal{O}(n + d^2)$. As a consequence, the chosen modeling must be consistent with the available number of repetitions.

2.3 Computing the likelihood

The evaluation of the likelihood function requires to compute the individual likelihoods, which imply to estimate the multi-dimensional integrals of Eq.(12). A fundamental method to compute the integral is the Monte-Carlo method [46]. To compute the integral of a function $g(\cdot)$ with respect to any density f , the Monte-Carlo method works as follows: samples are generated with respect to the PDF f (potentially with MCMC methods); the integral is approximated as the empirical mean of the function $g(\cdot)$ calculated at each sample. The main issue is that the sampling density $f(\boldsymbol{\theta}_i|\Psi)$ does not necessarily generate model parameters that result in a proper adequation of the model responses and the observations. Therefore, the likelihood function often collapses to 0. In order to generate model parameters that better suit the observations, an importance sampling scheme (written MCMC-IS) [32, 47] is implemented and described next. It will be compared to another usual methods to compute multidimensional integrals, the Laplace approximation [48]. This approach is based on an approximation of the integrand. It does not involve any sampling technique but requires to perform an auxiliary minimization.

2.3.1 Computation of the likelihood by importance sampling

Importance sampling scheme belongs to the same family of integration schemes as the classical Monte-Carlo [9]. An auxiliary PDF is used to generate samples in place of the initial density function. The idea is to generate model parameters associated to the model responses that are more consistent with the available observations. As a result, the likelihood function does not collapse to 0, a necessary condition for its maximization. The integral of Eq.(12) is rewritten as follows:

$$\int_{\mathbb{R}^d} f(\mathbf{y}_i|\boldsymbol{\theta}_i, \Psi)f(\boldsymbol{\theta}_i|\Psi)d\boldsymbol{\theta}_i = \int_{\mathbb{R}^d} f(\mathbf{y}_i|\boldsymbol{\theta}_i, \Psi) \frac{f(\boldsymbol{\theta}_i|\Psi)}{\pi_i(\boldsymbol{\theta}_i|\Psi)} \pi_i(\boldsymbol{\theta}_i|\Psi)d\boldsymbol{\theta}_i \quad (15)$$

with $\pi_i(\boldsymbol{\theta}_i|\Psi)$ the importance sampling density. The chosen importance sampling density is $\pi_i(\boldsymbol{\theta}_i|\Psi) = f(\boldsymbol{\theta}_i|\mathbf{y}_i, \Psi)$ which allows to generate model parameters conditioned on both Ψ and the available observations \mathbf{y}_i . To compute the integral in Eq.(15), it is necessary to generate samples from $\pi_i(\boldsymbol{\theta}_i|\Psi)$ which can be done with MCMC methods since this density is known up to a normalization constant (with respect to $\boldsymbol{\theta}_i$). Indeed, the conditional probability rule gives

$$f(\boldsymbol{\theta}_i|\mathbf{y}_i, \Psi) = \frac{f(\mathbf{y}_i, \boldsymbol{\theta}_i|\Psi)}{f(\mathbf{y}_i|\Psi)} \quad (16)$$

and the numerator is known from Eq.(11). It can be noticed that the denominator which appear in Eq.(16) is in fact the individual likelihood $\mathcal{L}_i(\Psi)$ defined in Eq.(12). Computing the integral requires to evaluate $f(\mathbf{y}_i|\boldsymbol{\theta}_i, \Psi)$ (which is known in closed form with Eq.(8)) and the importance sampling ratio $\frac{f(\boldsymbol{\theta}_i|\Psi)}{\pi_i(\boldsymbol{\theta}_i|\Psi)}$ (which demands to evaluate $f(\boldsymbol{\theta}_i|\mathbf{y}_i, \Psi)$). Samples can be generated thanks to MCMC technics but it remains necessary to evaluate the PDF value $f(\boldsymbol{\theta}_i|\mathbf{y}_i, \Psi)$. To provide an estimation of $f(\boldsymbol{\theta}_i|\mathbf{y}_i, \Psi)$, it is possible to use Kernel Density Estimation methods [49]. However, with such methods, the accuracy of the PDF estimation decreases when the dimension of $\boldsymbol{\theta}_i$ increases. Therefore, it is instead decided to approximate this density by

$$\pi_i(\boldsymbol{\theta}_i|\Psi) = \mathcal{N}(m(\boldsymbol{\theta}_i, \mathbf{y}_i, \Psi), C^2(\boldsymbol{\theta}_i, \mathbf{y}_i, \Psi)) \quad (17)$$

with $m(\boldsymbol{\theta}_i, \mathbf{y}_i, \Psi) = \mathbb{E}(\boldsymbol{\theta}_i|\mathbf{y}_i, \Psi)$ the empirical mean of the MCMC samples of $f(\boldsymbol{\theta}_i|\mathbf{y}_i, \Psi)$ and $C^2(\boldsymbol{\theta}_i, \mathbf{y}_i, \Psi) = \mathbb{V}(\boldsymbol{\theta}_i|\mathbf{y}_i, \Psi)$, the empirical covariance matrix with only diagonal terms in

$\mathcal{M}_d(\mathbb{R})$. Finally, for M i.i.d. samples generated with respect to $\pi_i(\boldsymbol{\theta}_i|\Psi)$ as defined in Eq.(17) (labeled $\tilde{\boldsymbol{\theta}}_i^k, k \in \llbracket 1, M \rrbracket$), the integral of Eq.(15) is approximated as follows:

$$\int_{\mathbb{R}^d} f(\mathbf{y}_i|\boldsymbol{\theta}_i, \Psi) f(\boldsymbol{\theta}_i|\Psi) d\boldsymbol{\theta}_i \approx \frac{1}{M} \sum_{k=1}^M f(\mathbf{y}_i|\tilde{\boldsymbol{\theta}}_i^k, \Psi) \frac{f(\tilde{\boldsymbol{\theta}}_i^k|\Psi)}{\pi_i(\tilde{\boldsymbol{\theta}}_i^k|\Psi)}$$

This computation is carried out for each individual likelihood (typically the likelihood associated to each specimen). The most computationally demanding part of the likelihood estimation is the generation of MCMC samples which requires thousands of repeated material model evaluations.

2.3.2 Computation of the likelihood through the Laplace approximation

The Laplace approximation [50] applies to integrals of the type

$$A = \int_{\mathbb{R}^d} e^{h(\mathbf{x})} d\mathbf{x}$$

with $h(\cdot)$ a function which complies with some constraints:

1. $h(\cdot)$ admits a global maximum \mathbf{x}_0 that belongs to the integration interval,
2. $h(\cdot)$ is a twice-differentiable function,
3. its Hessian matrix computed at $\mathbf{x} = \mathbf{x}_0$ is a symmetric definite negative matrix.

The main idea is to state that only points close to \mathbf{x}_0 significantly contribute to the integral. The different calculations that allow to establish the equations of the Laplace approximation are presented below. The Taylor expansion of $h(\cdot)$ at \mathbf{x}_0 can be written as :

$$h(\mathbf{x}) = h(\mathbf{x}_0) + (\mathbf{x} - \mathbf{x}_0)^T \nabla h(\mathbf{x}_0) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_0)^T \mathcal{H}(h)(\mathbf{x}_0) (\mathbf{x} - \mathbf{x}_0) + o(\|\mathbf{x} - \mathbf{x}_0\|^2) \quad (18)$$

with:

- $\nabla h(\mathbf{x}_0) = (\frac{\partial h}{\partial x_i}(\mathbf{x}_0))_{i \in \llbracket 1, d \rrbracket}$ the gradient vector of $h(\cdot)$ at \mathbf{x}_0 ,
- $\mathcal{H}(h)(\mathbf{x}_0) = (\frac{\partial^2 h}{\partial x_i \partial x_j}(\mathbf{x}_0))_{(i,j) \in \llbracket 1, d \rrbracket^2}$ the Hessian matrix of $h(\cdot)$ at \mathbf{x}_0 .

As \mathbf{x}_0 is the global maximum, the gradient vanishes and the substitution of $h(\mathbf{x})$ in Eq.(2.3.2) by its approximation determined in Eq.(18) gives:

$$\int_{\mathbb{R}^d} e^{h(\mathbf{x})} d\mathbf{x} \approx \int_{\mathbb{R}^d} e^{h(\mathbf{x}_0) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_0)^T \mathcal{H}(h)(\mathbf{x}_0) (\mathbf{x} - \mathbf{x}_0)} d\mathbf{x} = e^{h(\mathbf{x}_0)} \int_{\mathbb{R}^d} e^{\frac{1}{2} (\mathbf{x} - \mathbf{x}_0)^T \mathcal{H}(h)(\mathbf{x}_0) (\mathbf{x} - \mathbf{x}_0)} d\mathbf{x} \quad (19)$$

$\mathcal{H}(h)(\mathbf{x}_0)$ is symmetric definite negative so $-\mathcal{H}(h)(\mathbf{x}_0)$ is symmetric definite positive. As a result, in Eq.(19), the integrand is a Gaussian PDF of mean \mathbf{x}_0 et and covariance matrix $-\mathcal{H}(h)(\mathbf{x}_0)^{-1}$. As a PDF always integrates to 1,

$$\int_{\mathbb{R}^d} e^{-\frac{1}{2} (\mathbf{x} - \mathbf{x}_0)^T M^{-1} (\mathbf{x} - \mathbf{x}_0)} d\mathbf{x} = (2\pi)^{\frac{d}{2}} \sqrt{|M|} \quad (20)$$

Applying Eqs.(19) and (20), the Laplace approximation of A is obtained as

$$A \approx e^{h(\mathbf{x}_0)} \frac{(2\pi)^{\frac{d}{2}}}{\sqrt{|-\mathcal{H}(h)(\mathbf{x}_0)|}} \quad (21)$$

For a given $\Psi = (\boldsymbol{\mu}, \Sigma, \Omega)$, the individual likelihood reads as [48]

$$\mathcal{L}_i(\Psi) = \int_{\mathbb{R}^d} f(\mathbf{y}_i|\boldsymbol{\theta}_i, \Psi) f(\boldsymbol{\theta}_i|\Psi) d\boldsymbol{\theta}_i = \int_{\mathbb{R}^d} \frac{1}{(\omega_i \sqrt{2\pi})^{N_i} \sqrt{|\Sigma|(2\pi^d)}} e^{-\frac{g_i(\boldsymbol{\mu}, \Delta_i, \mathbf{y}_i, \mathbf{b}_i)}{2\omega_i^2}} d\mathbf{b}_i \quad (22)$$

with $\boldsymbol{\theta}_i = \boldsymbol{\mu} + \mathbf{b}_i$ where $\boldsymbol{\mu}$ stands for the mean (*i.e.* the fixed effects) and \mathbf{b}_i are the individual deviations (*i.e.* the random effects). Δ_i is the result of the Cholesky decomposition of $\omega_i^2 \Sigma^{-1}$ (so $\omega_i^2 \Sigma^{-1} = \Delta_i \Delta_i^T$). The function $g_i(\cdot)$ is defined by

$$g_i(\boldsymbol{\mu}, \Delta_i, \mathbf{y}_i, \mathbf{b}_i) = \|\mathbf{y}_i - \mathcal{F}(\mathbf{t}_i, \boldsymbol{\mu} + \mathbf{b}_i)\|^2 + \|\Delta_i \mathbf{b}_i\|^2 \quad (23)$$

The Laplace method is applied in two steps:

1. Search for the individual parameters (or rather the deviations), $\hat{\mathbf{b}}_i$, minimizing $g_i(\cdot)$ (23),
2. Computation of the Laplace approximation with formula in Eq.(21).

The Hessian matrix of $g_i(\cdot)$ at $\hat{\mathbf{b}}_i$ is [48]:

$$\mathcal{H}(g_i)(\hat{\mathbf{b}}_i) = \frac{\partial^2 \mathcal{F}(\boldsymbol{\mu}, \mathbf{y}_i, \hat{\mathbf{b}}_i)}{\partial \mathbf{b}_i \partial \mathbf{b}_i^T} (\mathbf{y}_i - \mathcal{F}(\boldsymbol{\mu}, \mathbf{y}_i, \hat{\mathbf{b}}_i)) + \frac{\partial \mathcal{F}(\boldsymbol{\mu}, \mathbf{y}_i, \hat{\mathbf{b}}_i)}{\partial \mathbf{b}_i} \frac{\partial \mathcal{F}(\boldsymbol{\mu}, \mathbf{y}_i, \hat{\mathbf{b}}_i)}{\partial \mathbf{b}_i} + \Delta_i^T \Delta_i \quad (24)$$

In practice, $\frac{\partial^2 \mathcal{F}(\boldsymbol{\mu}, \mathbf{y}_i, \hat{\mathbf{b}}_i)}{\partial \mathbf{b}_i \partial \mathbf{b}_i^T} (\mathbf{y}_i - \mathcal{F}(\boldsymbol{\mu}, \mathbf{y}_i, \hat{\mathbf{b}}_i))$ can be neglected if the model $\mathcal{F}(\cdot)$ is close enough to the experiment \mathbf{y}_i [51]. The term $\frac{\partial \mathcal{F}(\boldsymbol{\mu}, \mathbf{y}_i, \hat{\mathbf{b}}_i)}{\partial \mathbf{b}_i}$ is evaluated using a finite difference scheme. As a result, the negative log-likelihood is finally expressed as:

$$\begin{aligned} -\ln(\mathcal{L}(\Psi)) &= -\sum_{i=1}^n \ln(\mathcal{L}_i(\Psi)) = -\sum_{i=1}^n \ln \left(\int_{\mathbb{R}^d} f(\mathbf{y}_i|\boldsymbol{\theta}_i, \Psi) f(\boldsymbol{\theta}_i|\Psi) d\boldsymbol{\theta}_i \right) \\ &\approx \frac{n}{2} \ln(|\Sigma|) + \sum_{i=1}^n \left(\frac{1}{2} \ln(-|\mathcal{H}(g_i)(\hat{\mathbf{b}}_i)|) + \frac{g_i(\boldsymbol{\mu}, \Delta_i, \mathbf{y}_i, \hat{\mathbf{b}}_i)}{2\omega_i^2} + N_i \ln(\omega_i \sqrt{2\pi}) \right) \end{aligned} \quad (25)$$

In both approaches (MCMC-IS and Laplace), the computation of the different individual likelihoods is independent. As a result, to reduce the computational time, the calculation of the individual likelihoods is performed in parallel.

2.4 Maximizing the likelihood

Now that the expression of the likelihood for a mixed-effects model has been established, the model parameters can be estimated by maximizing it, or equivalently (although numerically better conditioned), by minimizing minus the log-likelihood,

$$\hat{\Psi} = \arg \min_{\Psi \in \Xi} -\ln(\mathcal{L}(\Psi)) \quad (26)$$

Remember that computing the likelihood is numerically challenging as it is necessary to estimate a multi-dimensional integral with respect to the unobserved variables Eq.(12). To solve the minimization problem (Eq. (26)), the first possibility is to rely on gradient-based algorithms which need either to compute the gradient or an approximation of the gradient of the objective function. Here, the gradient of $-\ln(\mathcal{L}(\Psi))$ with respect to Ψ is difficult to compute analytically. The application of classical differentiation formulas shows that it is necessary to differentiate the individual likelihoods which is a tricky task as it requires to differentiate the integral of Eq. (12). Indeed, differentiating the integrals is difficult as both the integrand and the integration variable depend of Ψ via Eq. (2). A second possibility is to use gradient-free optimization algorithms such as evolutionary algorithms but they tend to require too many likelihood estimations. As a result, neither gradient-based algorithms nor evolutionary algorithms are used in this paper to solve the problem defined in Eq.(26).

Others methods based on likelihood maximization include the Stochastic Approximation Expectation Maximization algorithm (SAEM) [47]. The algorithm is an adaptation of the classical Expectation Maximization algorithm which uses a stochastic approximation of the likelihood [52]. The algorithm builds a sequence of estimate of Ψ , $(\Psi_k)_{k \in \mathbb{N}}$ which converges to the exact optimal value of Ψ under regularity assumptions on the likelihood [47]. This algorithm is implemented for instance in the toolbox SAEMIX [53].

In the current work, to speed-up the convergence of the optimization, a Bayesian optimization scheme is chosen [54]. Bayesian optimization works as follows. The search starts by computing the objective function over a design of experiment (DoE) a latin hypercube sampling [55] for instance. Next, a surrogate model of the objective function is constructed with a Gaussian process [56]. The Gaussian process allows the definition of an infill criterion (here the expected improvement [57]) which is maximized in the search space Ξ to determine the location where the next exact likelihood function should be evaluated. The procedure is repeated until a stopping criterion is met, here a maximum number of likelihood estimations.

2.5 Estimating the individual parameters

The mixed-effects models approach allows to infer the individual parameters. With the two methods (Laplace approximation and importance sampling scheme) chosen to compute the likelihood, individual parameters are by-product of the likelihood function calculation at $\hat{\Psi}$. In the importance sampling scheme, individual parameters are predicted as the mean of the PDF $\pi_i(\theta_i|\Psi)$, *i.e.*, the empirical mean of $\theta_i|y_i, \Psi$. Note in passing that MCMC samples can be used to propagate uncertainties through the model. With the Laplace approximation, individual parameters are computed through the minimization of function $g_i(\cdot)$ at $\hat{\Psi}$ and $\hat{\theta}_i = \mu + \hat{b}_i$. These individual parameters can be interpreted as the mode of $f(\theta_i|y_i, \hat{\Psi})$.

3 APPLICATIONS

In this section, the mixed-effects methodology is applied to the calibration of a material model of ceramic-matrix composite materials called the ONERA Damage Model (ODM) and presented in [33, 34, 35]. The ODM model is presented in Section 3.1. In Section ??, the choices regarding the statistical model. In Section 3.3, the approach is tested on synthetic experimental data in order to carry out different sensitivity analyses of the method. Eventually, in Section 3.4, real experimental data are calibrated.

3.1 The ONERA Damage Model

A simplified uni-axial version of the ODM model is now presented. The stress (in MPa) will be labeled σ and the strain (without units) ε . The strain-stress relation is

$$\sigma(t) = E^{\text{eff}}(d_s(t))\varepsilon(t) - E_0\varepsilon_r(t) \quad (27)$$

with E_0 the Young's modulus, E^{eff} the effective Young's modulus which takes into account the loss in stiffness of the material caused by the damage d_s , ε_r is the residual strain and t the time. The residual strain is the strain left in the absence of load. The effective Young's modulus reads

$$E^{\text{eff}}(d_s(t)) = \frac{E_0}{1 + \eta(t)d_s(t)} \quad (28)$$

The damage desactivation index, η , represents the fact that damage does not impact the stiffness for compressive loads. It is given by $\eta(t) = h(\varepsilon(t))$ with h the Heaviside function. The damage d_s is computed from the thermodynamical force y . The damage equations are the following

$$\begin{cases} y(t) = \frac{1}{2}E_0\langle\varepsilon(t)\rangle_+^2 & (29) \\ g_s(y(t)) = \frac{\sqrt{y_{\max}(t)} - \sqrt{y_{0s}}}{\sqrt{y_{cs}}} \text{ with } y_{\max}(t) = \sup_{\tau \in [0,t]} y(\tau) & (30) \\ d_s(t) = d_c(1 - e^{-(g_s(y(t)))_+^p}) & (31) \end{cases}$$

$\langle x \rangle_+$ stands for the positive part of x (*i.e.* 0 if $x < 0$ and x if $x > 0$). Eq.(29) defines the thermodynamical driving force. The positive part means that damage is created only under tensile stress. In Eq.(30), the parameter y_{0s} is the damage threshold, y_{cs} the damage evolution celerity, and p a shape parameter called the damage evolution exponent. The damage threshold indicates the beginning of damage. Indeed, in Eqs.(29) and (30), for thermodynamical forces smaller than y_0^s , the damage does not increase and the model sums up to $\sigma = E_0\varepsilon$ using Eq.(28). To this damage threshold can be associated a strain damage threshold given by $\varepsilon_0^s = \sqrt{\frac{2y_{0s}}{E_0}}$. The parameter d_c stands for the damage saturation. The damage evolution exponent controls the regularity of the transition between the linear and non linear parts of the curve. If $p > 1$, the derivative of the stress with respect to the strain (known as the tangent matrix), $\frac{\partial \sigma}{\partial \varepsilon}$, is continuous on the whole curve. However, if $p < 1$, the tangent matrix is not continuous at $\varepsilon = \varepsilon_0^s$ and it can create numerical issues in finite element calculations. Nevertheless, contrary to the case $p > 1$, the damage threshold is much easier to calibrate if $p < 1$ as it is directly related to a kink on strain-stress curves.

The residual strain evolves according to the following equation:

$$\frac{\partial \varepsilon_r}{\partial t} = \chi \eta \frac{\partial d_s}{\partial t} \left(\frac{E^{\text{eff}}}{E_0} \right)^2 \varepsilon \quad (32)$$

In Eq.(32), only χ has to be identified. The model parameters are summed up in Table 1.

3.2 Settings of the statistical model

We now seek to apply the mixed-effects models methodology on strain-stress curves which can be modeled by the ODM model. The first step consists in applying the methodology to virtual data to investigate its ability to calibrate stress-strain curves and describe the material

Table 1: ODM Model parameters.

Young's modulus	damage threshold	damage evolution celerity	damage saturation	damage evolution exponent	residual strain
E_0	y_{0s}	y_{cs}	d_c	p	χ

variability. Since the data is generated directly from the model, one knows the exact model parameters distribution and the individual parameters values, and we can confront them to the calibration results. It is also possible to evaluate the robustness of the calibration technique to the number of available individuals. More formally, for a given mean μ and covariance matrix Σ (this set of parameters will be denoted Ψ_{exact}), it is possible to sample the individual parameters which are labeled $\theta_{i,\text{exact}}$. For each of these individual parameters, the ODM model outputs $\sigma_{i,\text{calc}}$ are computed, which stand for the different specimens. Using the mixed-effects approach, a calibrated distribution characterized by the estimated distribution parameters (labeled $\hat{\Psi}$) and the corresponding individual parameters (denoted $\hat{\theta}_i$) is obtained. One objective consists in checking that the calibrated distribution is consistent with the exact distribution, the virtual data and the exact individual parameters. Other items will be investigated such as the impact of the number of individuals on the calibration process and how the results of the calibration depend on the involved specimens.

Throughout this work, the following indicators will be studied:

- The Kullback-Leibler divergence [58] between the exact f_{exact} and the approximated distribution $\hat{f}_{\text{calibrated}}$: $\text{KL}(f_{\text{exact}}, \hat{f}_{\text{calibrated}}) = \int_{\mathbb{R}^d} f_{\text{exact}}(\theta | \Psi_{\text{exact}}) \ln \left(\frac{f_{\text{exact}}(\theta | \Psi_{\text{exact}})}{\hat{f}_{\text{calibrated}}(\theta | \hat{\Psi})} \right) d\theta$
- Error in the parameters space : for all the individual parameters set, it reads as $\frac{1}{n} \sum_{i=1}^n \frac{|\theta_{i,\text{exact}} - \hat{\theta}_i|}{\theta_{i,\text{exact}}}$ (division is elementwise, with no target value equal to 0)
- Error of distribution parameters : $\frac{|\Psi_{\text{exact}} - \hat{\Psi}|}{\Psi_{\text{exact}}}$ (division is elementwise, with no target value equal to 0)
- Error in model space : for all the experiments, it can be written as $\frac{1}{n} \sum_{i=1}^n \frac{1}{N_i} \|\sigma_{i,\text{calc}} - \mathcal{F}(\mathbf{t}_i, \hat{\theta}_i)\|_2$

Only the last criterion can be used to study the results obtained with experimental data because in this case neither the exact model parameters distribution nor the exact individual parameters are known.

Several Assumptions are made here. First, only monotonic loadings are considered so that, from the original 6 parameters of the ODM model, the residual strain component can be neglected. Consequently, χ is set to 0 because this parameter is only involved in the computation of the residual strain component. In addition, the material model implies strong correlations such as those between y_c^s and p or y_0^s and p . To avoid correlations at the beginning, it is decided to set p to 1.2, a usual value for this kind of materials. As a consequence, all the parameters are considered as independent and the covariance Σ is a diagonal matrix made of the variance terms. This simplifying assumption is discussed in the perspectives of the paper. It is also decided to assign to each specimen the same error term

$$\forall (i, j) \in \llbracket 1, n \rrbracket \times \llbracket 1, N_i \rrbracket \omega_{ij} = \omega$$

Table 2: Exact values of the parameters distribution.

	E_0 [MPa]	y_{0s} [MPa]	y_{cs} [MPa]	d_c
Mean	1.89×10^5	2.50×10^{-2}	2.24	3.77
Standard deviation	1.79×10^4	8.00×10^{-3}	0.439	0.893

As a result, 8 to 9 parameters have to be calibrated (depending on if ω is calibrated or fixed which stand for 4 mean and 4 variance parameters).

For the computation of the likelihood with the importance sampling scheme, 10^4 MCMC samples are generated and the first 500 samples burnt. The thinning is set to 20. Once all these samples are discarded, it remains 4750 of the 10^4 initial samples. The MCMC sampler used can be found in [59]. The algorithm used to minimize the function $g_i(\cdot)$ (Eq.(23)) is the CMA-ES algorithm [60] with a budget of 10^3 iterations and a population size of 50.

The algorithm that maximizes the likelihood function is the EGO algorithm from the GPy library [61]. The initial DoE is a latin hypercube sampling with 45 points to which 160 points are added during the optimization. Variables are normalized between 0 and 1. Optimization bounds are discussed in the hereafter.

3.3 Calibrating with virtual data

3.3.1 Generating virtual data

The targeted mean and standard deviations of the parameters are chosen at values which are consistent with usual observations as presented in Table 2. 20 i.i.d samples from $f_{\Psi_{\text{exact}}}$ are generated. For each of these samples and given a strain profile (the strain rate $\dot{\epsilon} = \frac{\partial \epsilon}{\partial t}$ is equal to $3.00 \times 10^{-4} \text{ s}^{-1}$), the corresponding model outputs $(\sigma_{i,\text{calc}})_{i \in \llbracket 1, 20 \rrbracket}$ are computed. An heteroscedastic noise is added to the experimental data:

$$\sigma_{i,\text{noisy}} = \sigma_{i,\text{calc}} \times (1 + \beta\zeta) \text{ with } \zeta \sim \mathcal{N}(0, 1) \text{ and } \beta = 0.02 \quad (33)$$

Exact and noisy data are depicted in Figure 2. The bounds on both means and standard deviations used for the optimization are shown in Tables 3 and 4. Within these bounds, the search space of the model parameter distribution is large and the different distributions exhibit significant variety. This can be seen in Figure 3 where some marginals for different means and variances of E_0 are represented.

3.3.2 Calibration with a fixed error

This section is dedicated to the presentation of the results of the calibration of the ODM model with the virtual data presented in Figure 2. The results from both approaches (Laplace and MCMC-IS) are presented. The standard deviation of the error ω is fixed to 1 MPa. The 10 specimens involved in the calibration process are presented in Figure 2. The presented calibration process is run with the different settings introduced in Section 3.2.

To analyze the results, it is possible to look at the calibrated parameters distribution characterized by its mean and standard deviation. These values are presented in Table 5. The first comment is that the calibrated distributions are rather close to the exact distributions, except

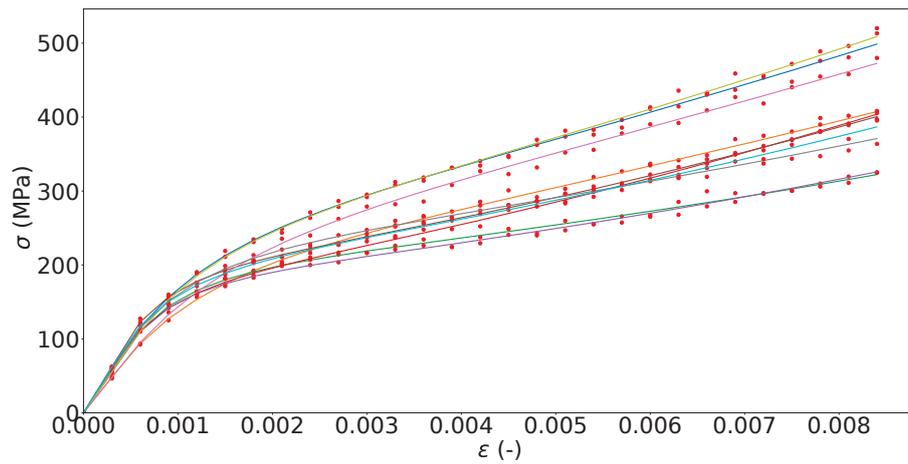


Figure 2: Model with exact individual parameters (lines) and noisy data (dots).

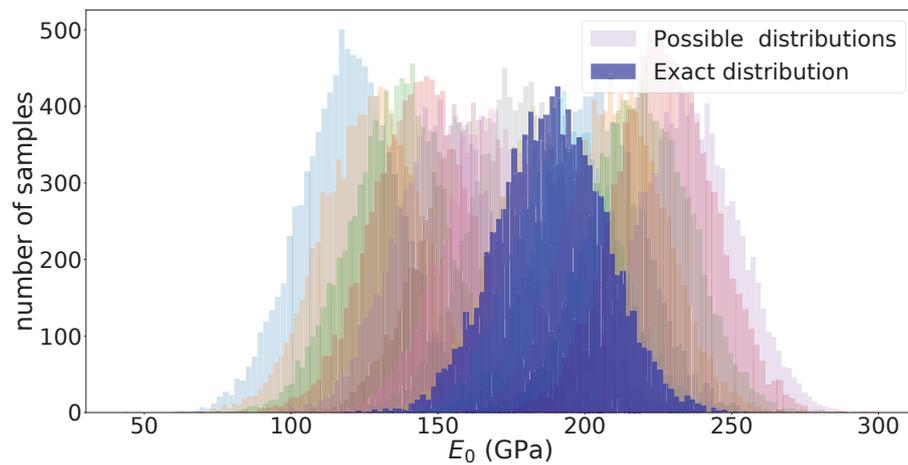


Figure 3: Possible marginal distributions for the parameter E_0 .

Table 3: Bounds for the mean parameters μ .

	E_0 [MPa]	y_{0s} [MPa]	y_{cs} [MPa]	d_c
Lower bound	1.20×10^5	5.00×10^{-3}	1.10	1.30
Upper bound	2.35×10^5	5.00×10^{-2}	3.50	6.00

Table 4: Bounds for the standard deviation parameters, Σ .

	σ_{E_0} [MPa]	$\sigma_{y_{0s}}$ [MPa]	$\sigma_{y_{cs}}$ [MPa]	σ_{d_c}	ω [MPa]
Lower bound	1.40×10^4	3.00×10^{-3}	0.250	0.600	0.381
Upper bound	2.00×10^4	1.00×10^{-2}	0.500	1.08	12.5

for y_{0s} . The corresponding marginals are displayed in Figures 5a, 5c, 5d. Both approaches seem to provide a different result as as two order of magnitude separate their Kullback-Leibler divergences (see Table 7). However, omitting the y_{0s} makes the KL divergence fall to 0.81 for the MCMC approach and 0.33 for the Laplace approach. The two approaches provide similar results except for the damage threshold. The marginal distributions associated to the Young's modulus and the damage evolution celerity are well calibrated. The distribution associated to the Young's modulus is generally the best calibrated parameter distribution as it appears to be the most sensitive parameter in the ODM model, followed by y_{cs} and d_c . The differences between the different coordinates are further discussed later.

On Figure 6, the resulting calibrated individual parameters value for each specimen is displayed. It is important to notice that the values of the individual model parameters are all different allowing to express the variability of the material specimen. This is a key aspect of the proposed approach compared to existing techniques discussed in the introduction. Considering the results about the individual model parameters, the mean relative errors (over the 10 individuals) between the exact and the calibrated individual parameters is presented in Table 8. The relative error on the Young's modulus, which is from both a numerical and mechanical points of view the most sensitive parameters, is the smallest with respect the other mean errors. The damage saturation d_c and the damage evolution celerity y_{cs} are also properly calibrated in both approaches. However, as it was possible to expect, the mean of relative errors on y_{0s} is the highest in each approach. It is also possible to check that the data used in the calibration process is properly calibrated. For that, the model outputs corresponding to the estimated parameters are plotted in Figure 7a in which it is possible to notice that the model is properly calibrated as the model responses are in adequation with the virtual data. This can be assessed by checking the averaged distance in the model space given in Table 9. For each specimen, the difference between the model responses with the exact individual parameters and with the calibrated individual parameters are illustrated in Figure 7b. It illustrates the efficient calibration of the ndividual model parameters for all the considered specimens. Even if the marginal distributions for the model parameters are assumed to be independent, in the MCMC-IS approach, the generated model parameter samples by the importance density using MCMC (Figure 8) exhibit the correlations between the parameter coordinates in order to generate model responses in adequation with the available virtual data.

Neither the marginals nor the individual parameters of the damage threshold y_{0s} are well calibrated. One of the hypotheses which can explain this fact is the lack of sensitivity of the damage threshold in the objective function. To check this assumption, it is possible to carry out a raw sensitivity analysis by studying the lengthscales of the Gaussian process involved in the optimization process. This interpretation comes from the fact that the lengthscale acts as a wavelength of the Gaussian process which is here a surrogate model of the mixed-effects likelihood function. If the wavelength along one axis of the Gaussian process is small, it means that the range of variations of the objective function along this direction is small, which may be interpreted as a limited sensitivity of the objective function along this coordinate. On the contrary, if the wavelength is large, the variations of the objective function along this direction exhibit large variations, which means the objective function varies significantly along this axis and the corresponding parameter is sensitive. For the Laplace calibration, the order of magnitude of the lengthscales (standard and normalized with respect to the optimization bounds) are presented in Table 6. Standard numerical values give $\ell(E_0) \gg \ell(d_c), \ell(y_{cs}) > \ell(y_{0s})$ as expected. Normalized lengthscales only indicate that $\ell(E_0) \gg \ell(d_c), \ell(y_{cs}), \ell(y_{0s})$, which is still consistent with physical knowledge. The low sensitivity of y_{0s} in the likelihood function,

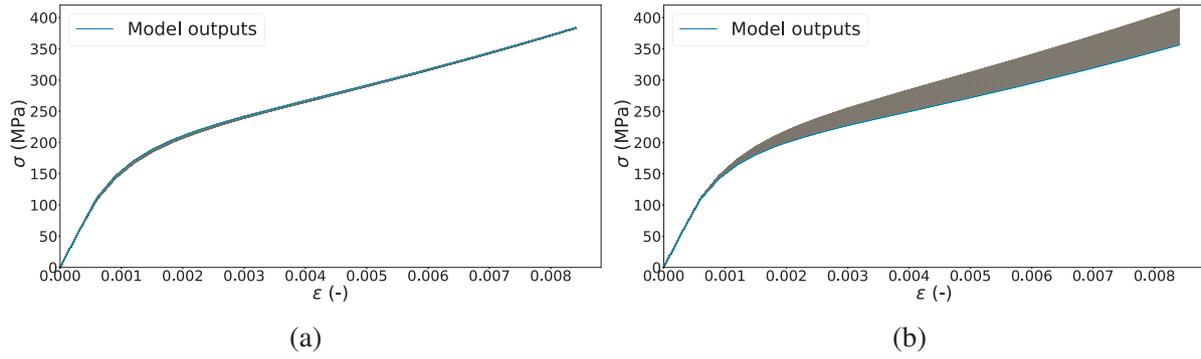


Figure 4: On the left (a), model outputs for different damage thresholds varying between $[0.9 \times \mu_{y_{0s}}, 1.1 \times \mu_{y_{0s}}]$ ($\mu_{y_{0s}}$ and other parameters correspond to the mean parameters presented in Table 1). On the right (b), model outputs for different damage thresholds varying between $[0.9 \times \mu_{y_{cs}}, 1.1 \times \mu_{y_{cs}}]$ ($\mu_{y_{cs}}$ and other parameters correspond to the mean parameters presented in Table 1).

which explains the differences encountered along the y_{0s} (Figure 5b) axis can come from the choice of p . In Section 3.1, it was stated that y_{0s} was complicated to calibrate for $p > 1$ with only strain-stress curves, which means that previous remarks are consistent with expert knowledge. To investigate the sensitivity of the model output towards the damage threshold, another simple solution can consist in computing the model outputs for different damage threshold and the other parameters fixed. On Figure 4a, it is possible to notice that the damage threshold, for a variation of 20% around the corresponding mean value $\mu_{y_{0s}}$, does not influence much the model output as for all strain values, the difference between the extreme stress curves is beneath 5 MPa. The relative variation of the stress is below 5% which can be considered as non significant. Yet, the model responses exhibit greater sensitivity for the same relative variation of the damage evolution celerity y_{cs} (see Figure 4b). The same observations can be made on the other model parameters. This ascertains the results obtained with the analyses on the lengthscales. In practice, y_{0s} is usually calibrated using acoustic data which are not available here. Damage starts when the first fibers of the composite material break. It produces acoustic events which can be recorded and help to indicate when the first failure (and the start of damage) happens.

In this section, the ability of the mixed-effects models methodology to calibrate the ODM model has been illustrated. This framework allows to characterize the material variability and its impact on the model parameters. In addition, to provide information on the distribution of the model parameters, it allows to determine reliable estimates of the individual model parameters. Furthermore, both computational methods of the likelihood (*i.e.* MCMC-IS and Laplace) give similar results in terms of calibrated model parameter distributions and individual parameter values. Given that the computation of the likelihood with the Laplace method is much faster to compute (for 10 specimens of 29 measurements each on a multi-threaded code with 4 CPU on a standard machine with processor Intel Core I5, MCMC-IS approach takes 2 minutes to compute the likelihood while the Laplace approach takes about 4 seconds), all further analyses will be conducted using the Laplace approximation method for the computation of the likelihood.

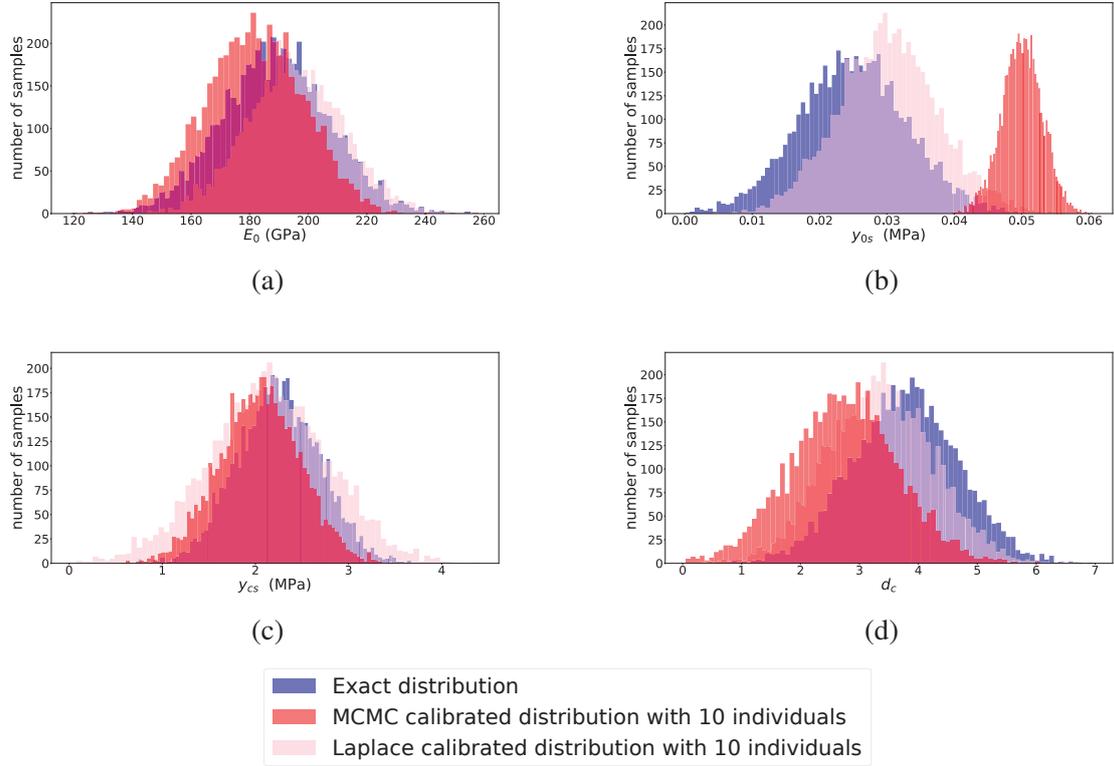


Figure 5: Exact and calibrated marginals along the E_0 axis (a), the y_{0s} axis (b), the y_{cs} axis (c), and the d_c axis (d).

Table 5: Calibrated means and standard deviations with fixed error term. The relative error between the exact Ψ and the calibrated Ψ in % is indicated between brackets.

	E_0 [MPa]	y_{0s} [MPa]	y_{cs} [MPa]	d_c
Exact means	1.88×10^5	0.0250	2.24	3.77
Laplace calibrated means	$1.95 \times 10^5(4)$	0.0301(20)	2.15(4)	3.36(10)
MCMC-IS calibrated means	$1.81 \times 10^5(4)$	0.0500(100)	2.05(8)	2.74(27)
Exact standard deviations	1.79×10^4	8.00×10^{-3}	0,439	0,893
Laplace calibrated standard deviations	$1.62 \times 10^4(9)$	$7.31 \times 10^{-3}(9)$	0.647(47)	0.871(2)
MCMC-IS calibrated standard deviations	$1.67 \times 10^4(7)$	$3.00 \times 10^{-3}(62)$	0.431(1)	0.910(2)

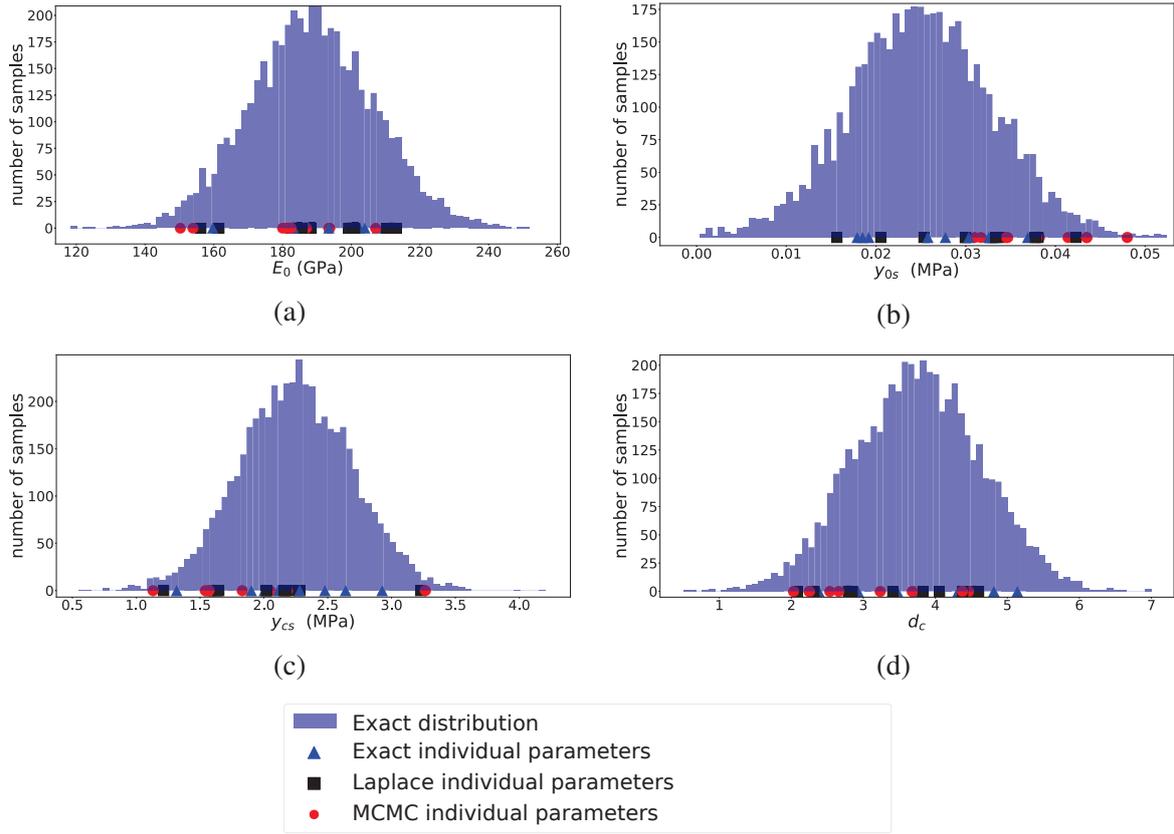


Figure 6: Exact marginals, exact and calibrated individual parameters along the E_0 axis (a), the y_{0s} axis (b), the y_{cs} axis (c), and the d_c axis (d).

Table 6: Order of magnitude of the Gaussian process lengthscale along each direction at the last iteration for the Laplace calibration process.

	$\ell(E_0)$	$\ell(y_{0s})$	$\ell(y_{cs})$	$\ell(d_c)$
standard lengthscale ℓ	10^9	10^{-2}	10^0	10^0
normalized lengthscale ℓ_r	10^4	10^{-1}	10^{-1}	10^{-1}

Table 7: Kullback-Leibler divergences for the virtual data case, the error term being fixed.

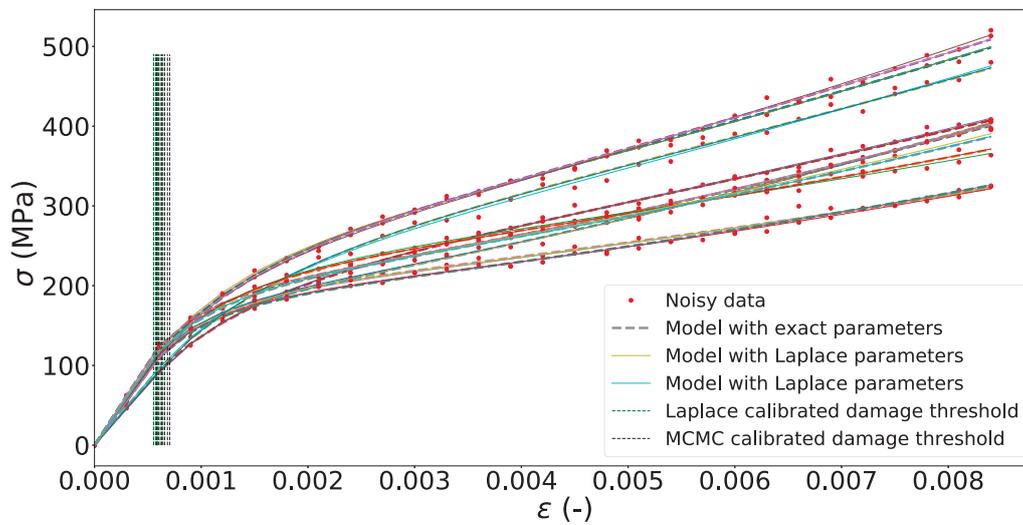
	MCMC-IS	Laplace
Kullback-Leibler divergence	37.3	0, 53

Table 8: Mean relative errors for the 10 calibrated individual parameters compared to the exact parameter values.

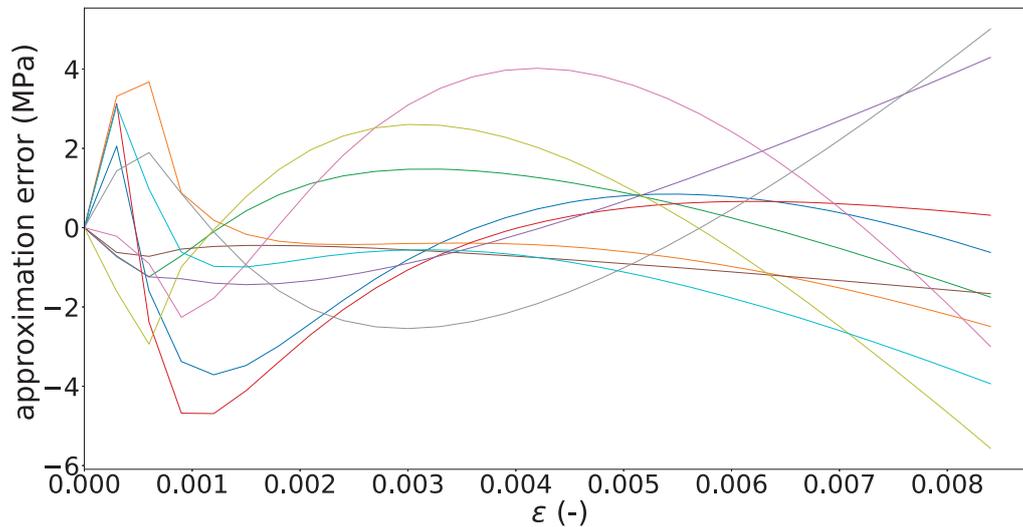
	E_0 [MPa]	y_{0s} [MPa]	y_{cs} [MPa]	d_c
Laplace relative error (%)	1.59	23.5	10.6	3.25
MCMC-IS relative error (%)	3.04	43.15	13.4	6.04

Table 9: Mean approximation error for the 10 model outputs corresponding to the calibrated individual parameters compared to model outputs corresponding the exact parameter values.

	Laplace	MCMC-IS
Distance in model space (MPa)	0,311	0,348



(a)



(b)

Figure 7: In (a), exact data used for calibration vs model responses for the estimated individual parameters. In (b), differences between model with exact parameters and with Laplace calibrated parameters.

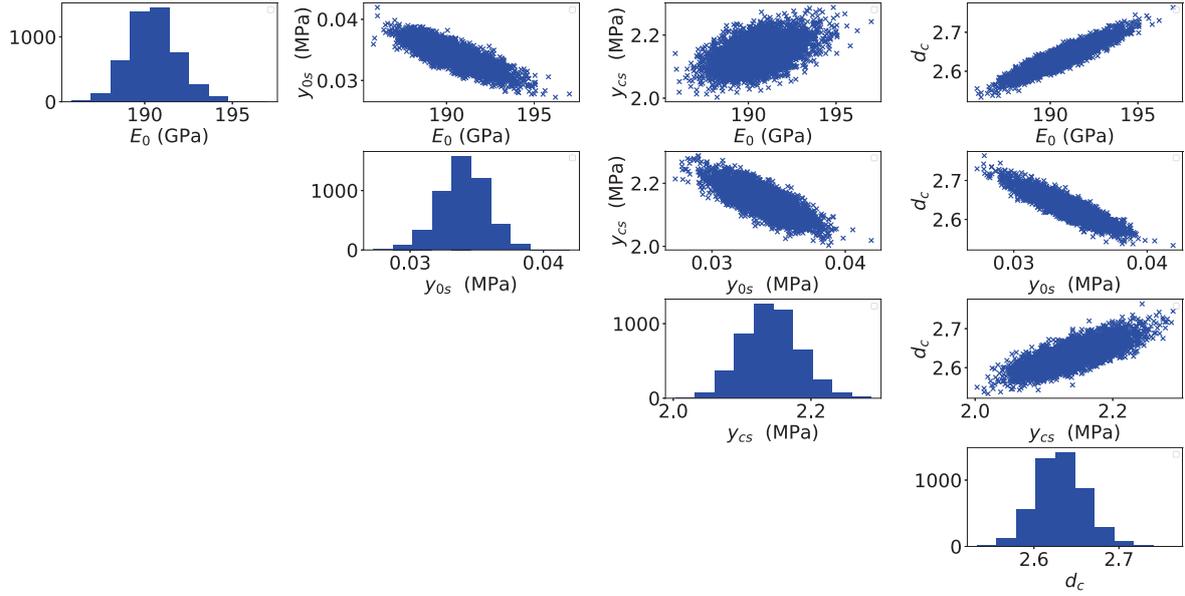


Figure 8: Pairplot of the MCMC samples of $\hat{f}(\theta|y_0, \hat{\Psi})$.

3.3.3 Calibration with an optimized error term

In this Section, the variance ω of the error term is not considered fixed anymore and it is calibrated along with the other parameters in Ψ . Moreover, the objective of this section is to study the influence of the number of available individuals in the calibration method.

The number of available observations plays a key role in calibration. In general, a high number of observations allows to reduce the uncertainty associated to the calibrated parameters. However, in mechanics, the number of specimens is limited due to time and cost considerations and the usual number of specimens varies from 5 to 10 in the best cases. As a result, to study the influence of the number of individuals on the calibration results, it is decided to calibrate the model with respectively 5, 10 and 20 individuals (all depicted in Figure 2) to take into account representative constraints on the number of available specimens. The upper bound on the variance of the error term is set to 10 MPa and the lower bound to 0.1 MPa.

The presented calibration approach using the Laplace technique for the likelihood estimation is carried out for 5, 10 and 20 individuals generated with the same exact model parameters distribution. To analyze the obtained results, it is possible to plot the marginals along the E_0 axis for 5, 10 and 20 individuals (see Figure 9). In this figure, it is possible to notice the marginal along the y_{cs} axis is better inferred as the number of specimens increases. This phenomenon can also be noticed for the marginal along the d_c axis (see Figure 17). However, the marginal along the E_0 axis is downgraded from 10 to 20 individuals (see Figure 16). As the marginal distributions are jointly calibrated, it is necessary to measure the improvement of the calibration using Kullback-Leibler divergences for 5, 10 and 20 individuals (y_{0s} is omitted in the KL calculation, see discussion in the previous section). The KL divergence decreases with the increase of the number of individuals (Table 10), therefore the distribution of the model parameters is better calibrated as the number of available specimens increases. In addition, on Figure 11b, it can be noticed that the damage threshold is always the parameter with the highest associated error: in other words, this parameter is complicated to calibrate regardless of the number

Table 10: Kullback-Leibler divergences for 5, 10 and 20 individuals (without consideration of y_{0s} marginal).

Number of specimens	5	10	20
Kullback-Leibler divergence	0.82	0.42	0.32

Table 11: Mean error (for all the deformation values) for 5, 10 and 20 individuals between the calibrated model responses and the model responses with the exact parameters.

Number of specimens	5	10	20
Mean approximation error (MPa)	0.226	0.327	0.357

of specimens. To illustrate the impact of the model parameter distribution convergence on the model responses, it is possible for each calibrated distribution (for 5, 10 and 20 individuals) and for the exact distribution to generate parameter samples (2000 in the present case). Then, for each parameter sample, the ODM model is computed. Eventually, the 5% and 95% quantiles in the model response space are estimated (Figure 10).

From Figure 10, it can be seen that the estimated 5% quantiles for the model responses using the calibrated distribution and the exact distribution converge one to another as the number of individuals increases. Similar analysis can be done for the 95% quantiles. It allows to see that the convergence of the calibrated model parameters distribution toward the exact distribution as the number of individuals increases, enables the generation of model responses that cover the entire set of possibilities (and not just a subset as for 5 individuals). This is of prime importance if this calibrated model parameter distribution is then used for uncertainty propagation in structure analysis for instance. It is therefore important to keep in mind that with a small number of individuals (around 5 specimens), the calibrated model parameter distribution might not be enough representative of the entire set of possible outcomes (due to the lack of individual diversity with only 5 specimens). Therefore, with a limited number of specimens, the calibrated model parameter distribution has to be used with attention.

Eventually, it is important to notice that the resulting model parameter values for each individual is closed to the exact parameter values even when only 5 individuals are considered.

In this section, the sensitivity to the number of individuals has been studied. It was has been that as the number of specimens increases, the calibrated marginals get closer to the exact marginals, and the same thing is observed in the model space. Finally, it was also noticed even with few specimens, the mixed-effects methods calibrates well the involved specimens.

3.3.4 Robustness of the model calibration with respect to different available individuals

Calibration in the mixed-effects models framework relies on specimens to determine the model parameter distribution and later estimate the individual parameters values. The objective of this section is to evaluate the robustness of the calibration using mixed-effects models with respect to the available specimens. In other words, whatever the specimens are, the population-

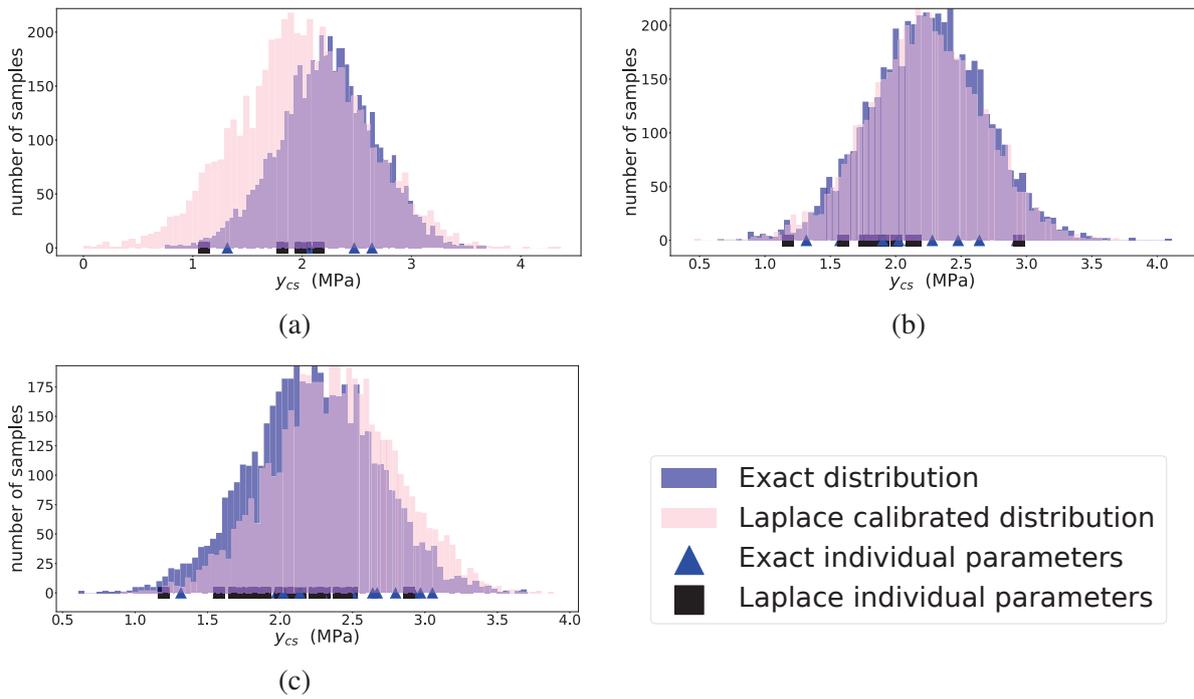


Figure 9: Exact and calibrated marginals along the y_{cs} axis for 5 individuals (a), 10 individuals (b) and 20 individuals (c).

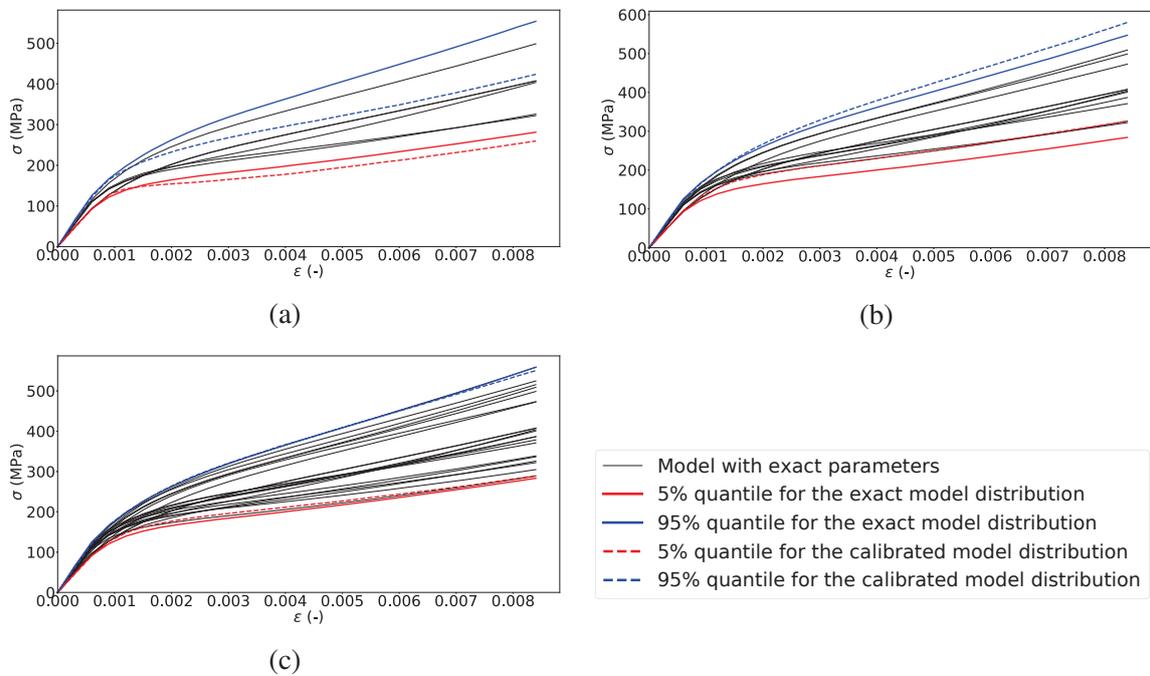


Figure 10: Uncertainty propagation from the distribution calibrated with 5 individuals (a), 10 individuals (b) and 20 individuals (c).

based approach is able to approximate correctly the model parameter distribution and the individual parameters values. To check this assumption, the calibration process is repeated 20 times with different sets of specimens. For each of these repetitions, 10 different individual parameters values set are generated from the same exact distribution $f_{\Psi_{\text{exact}}}$, the corresponding model outputs are computed and noise is added as in Eq.(33). The individual parameters are different from one repetition to the other. All the repetitions are calibrated with mixed-effects models methodology using the same settings (*i.e.* initial DoE, number of iterations, bounds, *etc.*).

For each repetition, the Kullback-Leibler divergence is computed (see Figure 11a). The advantage of the Kullback-Leibler divergence is that this quantity takes into account all the different variations of the joint distribution. It is also possible to compute the mean error between the estimated individual parameters and the exact individual parameters for the 10 considered individuals and that for the 20 repetitions (see the boxplots 11b).

The objective is to check that calibration using mixed-effects models approach does not depend of involved specimens. With Figure 11a and Figure 11b, it is possible to notice that for all the repetitions, the range of variation of the Kullback-Leibler divergences and the mean relative errors are rather small. This tends to indicate that for the distribution parameters, the calibration results do not depend too much to the involved specimens. It is also possible to consider the calibrated variances of the error term. The different calibrated standard deviations of the error terms with 10 specimens are shown in Figure 11c. For all the repetitions but one, the standard deviation of the error term varies between 4.53 to 6.41 MPa. As the hypothesis made on the noise in the likelihood function is different from the noise which was added to the virtual data, there is not any reference value. Nevertheless, it can be noticed that calibrated standard deviations are within a rather small range, which tends to confirm that the calibration of this parameter is rather independent of the specimens used for calibration.

Up to now, virtual data have been used in the calibration process in order to illustrate the approach with a pedagogical point of view. It allowed to compared to obtained calibration results with respect to a known model parameters distribution and to the known individual parameters values. With this virtual data, the sensitivity to the number of individuals and their repartitions have been analyzed.

In the next section, a test case using experimental data is carried out, using the same calibration process.

3.4 ODM model calibration with experimental data

We now apply the mixed-effects models to calibrate the ODM model with 13 tensile tests performed on CERASEP A400, a woven composite material. The available tests are plotted in Figure 12. The methodology settings (*i.e.* initial DoE, number of iterations, kernel of the Gaussian process) are the same as in the previous sections. The calibration bounds are given in Tables 12 and 13.

Notice in Figure 12 that all curves are less disparate than the virtual data generated in Section 3.3.1. Material failure happens at different strains and thus the ultimate tensile strength varies. Moreover, the transition between the linear and non-linear regimes is also subjected to variability. These observations constitute an interesting test case for a population-based approach such as the mixed-effects models. Finally, the experiments have different number of observations

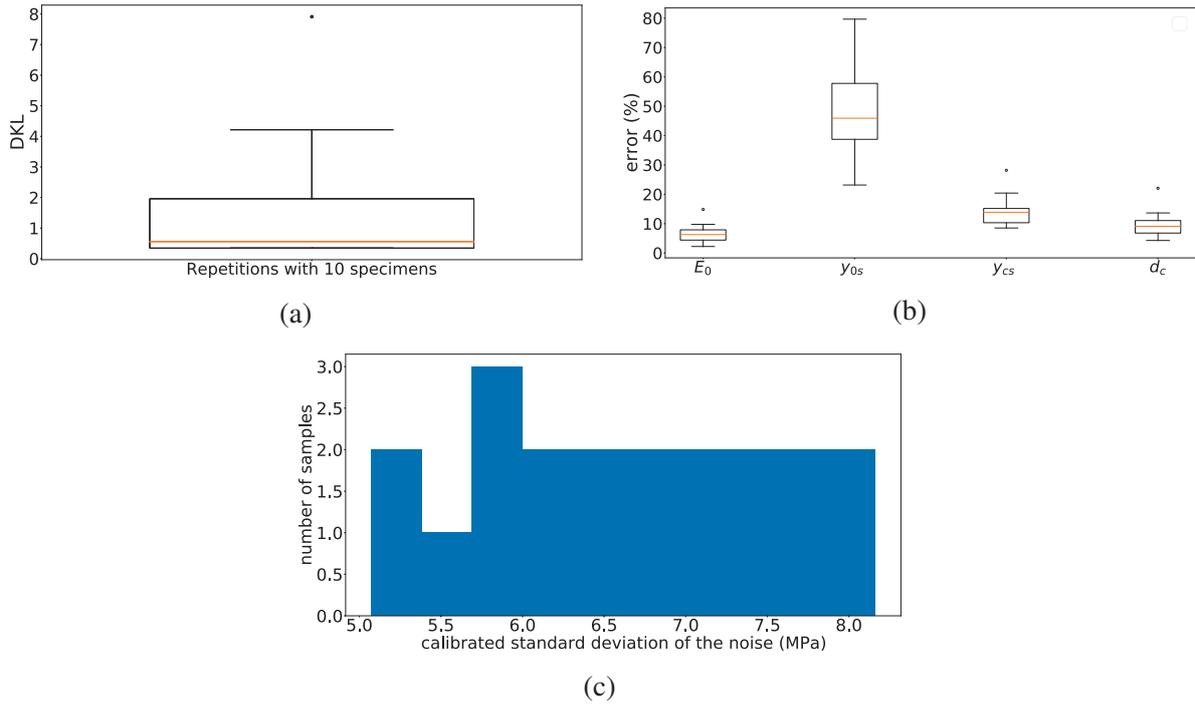


Figure 11: With 10 specimens: (a) boxplot of Kullback-Leibler divergence for all repetitions, (b) boxplot of the error of the individual parameters for all repetitions, (c) histogram of the calibrated standard deviation of the error term.

points (from 20 to 700). To make sure all individuals have the same weight in the likelihood function, it is decided to subsample those with more than 20 observation points. Experiments are subsampled periodically and always include the last point.

Figure 13 shows the calibrated marginals. The marginal along E_0 seems well calibrated in the sense that all the individual parameters belong to the interval $[\mu_{E_0} - 3\sigma_{E_0}, \mu_{E_0} + 3\sigma_{E_0}]$ which concentrates 99% of possible values for a Gaussian random variable. Along the y_{0s} axis, the same situation as in Section 3.3 is encountered as all the individual parameters are concentrated around the mean. The individual parameters are also concentrated around the mean for y_{cs} so that the variance seems overestimated. An explanation could be that the Bayesian optimization has not converged yet to the maximum likelihood estimator of Ψ . Along the d_c axis, the individual parameters seem consistent with the marginal even if without the point whose damage saturation is between 3 and 3.5, the variance of d_c would have been considered as overestimated. More classically, the adequation between the model responses for the estimated individual parameters and the experimental data is displayed in Figure 14 for two individuals (others can be found in appendix, Figures 19 and 20). In both plots, the estimated individual properly matches the experimental data (the same applies to the other experiments, see Figures 19 and 20)).

Eventually, it is possible to propagate the uncertainties of the calibrated distribution to the model parameters and to compare the results to the experimental data. In Figure 15, 1000 samples from the calibrated distribution are drawn and the corresponding model outputs computed. Notice that the experimental data are included in the bundle of model outputs, which shows

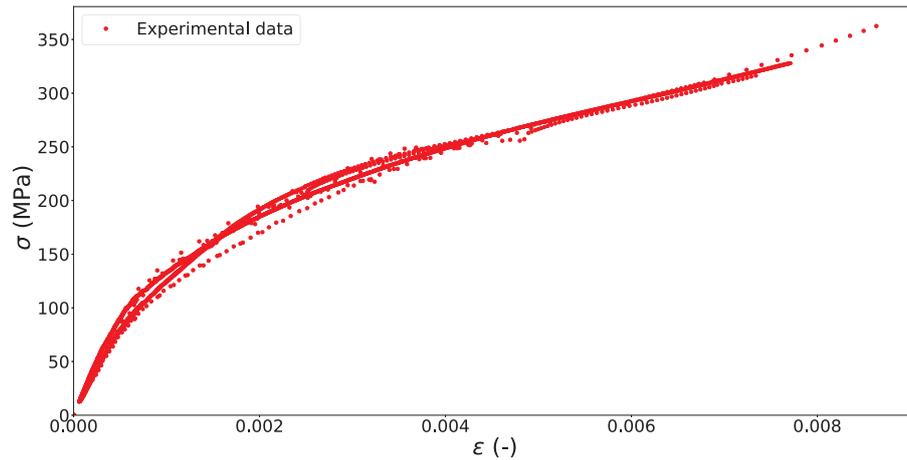


Figure 12: Experimental strain-stress curves of the CERASEP A400 material.

that the calibrated distribution conservatively learns the observed variability. The variability of the inferred distribution is more important than the one observed on the experimental curves. This overestimated variance of model parameters may be caused by the fact that the Bayesian optimization has not converged yet with in the given number of iterations. Taking into account correlations between model parameters in the calibration process might help to reduce the estimated variances.

4 CONCLUSIONS & PERSPECTIVES

In this paper, a population-based mixed-effects model was applied to calibrate a material model and to characterize material variability. First, the mixed-effects model has been described with an emphasis put on the numerical estimation of the likelihood. Then, this methodology has been applied to virtual data to compare the results with known model parameters distribution and individual parameters values. It has allowed to analyze the effects of the number of specimens and the specimens variability. Finally, the mixed-effects approach has been applied to experimental data obtained with a woven ceramic matrix composite material.

Our investigation show that mixed-effects models offer a promising framework to calibrate material parameters while taking into account specimen variability. Even if the accuracy of the calibrated parameter distribution seems sensitive to the number of specimens, the method typically provides accurate estimates of the individual parameters and captures well the observed variability. The number of specimens should also help to choose the level of details of the probabilistic model. For instance, calibrating correlations with a limited set of individuals yields estimates which may not be trustworthy.

In terms of future work, two major directions should be investigated. Firstly, it is important to include in the calibration process the correlations between the model parameters. A study of the relationship between the number of available individuals and the accuracy of the calibration correlations is essential. Secondly, improvements in the mixed-effects models are required in order to account for different types of data (for instance acoustic measures in addition to tensile-stress observations). In doing so, the likelihood will be more sensitive to this parameter and as a matter, the marginal along the y_{0s} will be better calibrated and should make the estimation of the damage threshold more accurate.

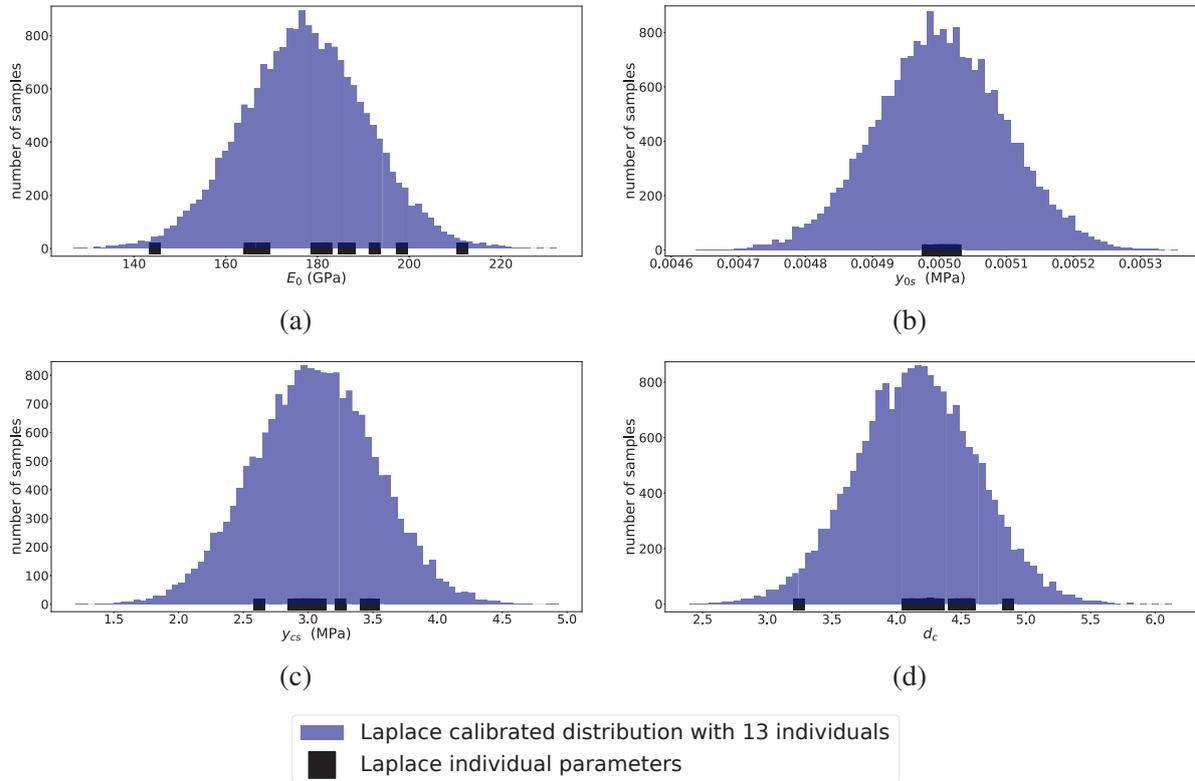


Figure 13: Calibrated model (histograms) and individual (squares) parameters along the E_0 (a), the y_{0s} (b), the y_{cs} (c) and the d_c axes (d) using the 13 CERASEP A400 specimens.

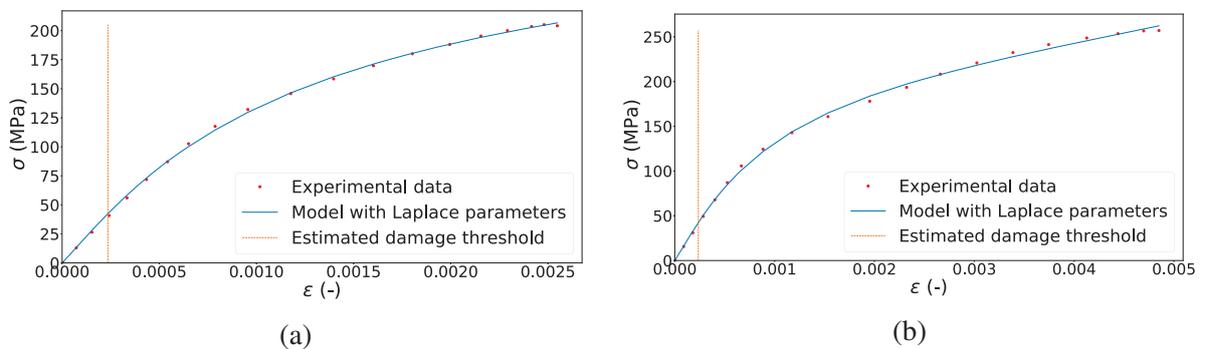


Figure 14: Model outputs vs. experimental data for the second (a) and the third (b) experiments.

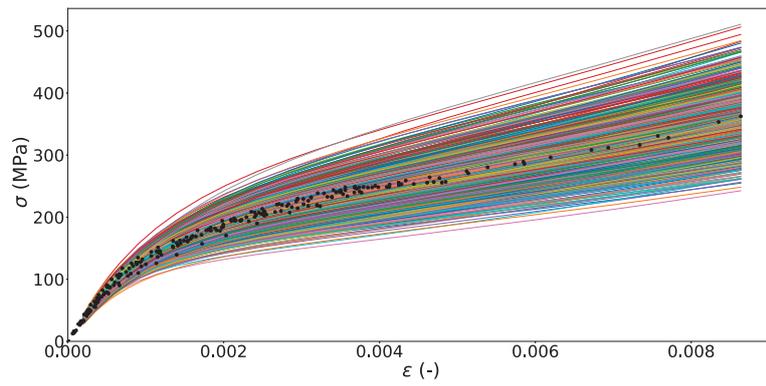


Figure 15: Uncertainty propagation from the calibrated distribution.

A Calibrated model parameter marginals of y_{cs} and d_c with the Laplace approach using virtual data

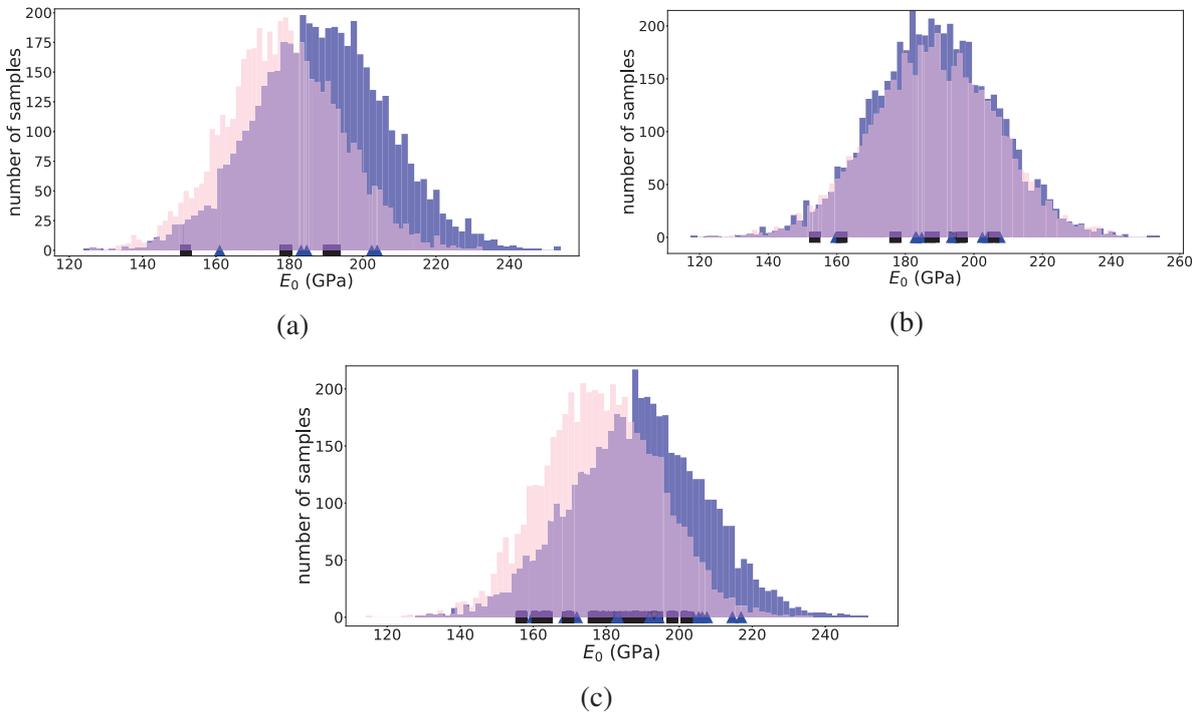


Figure 16: Exact and calibrated marginals along the E_0 axis for 5 individuals (a), 10 individuals (b) and 20 individuals (c).

B Calibrated model parameter marginals of d_c with the Laplace approach using virtual data

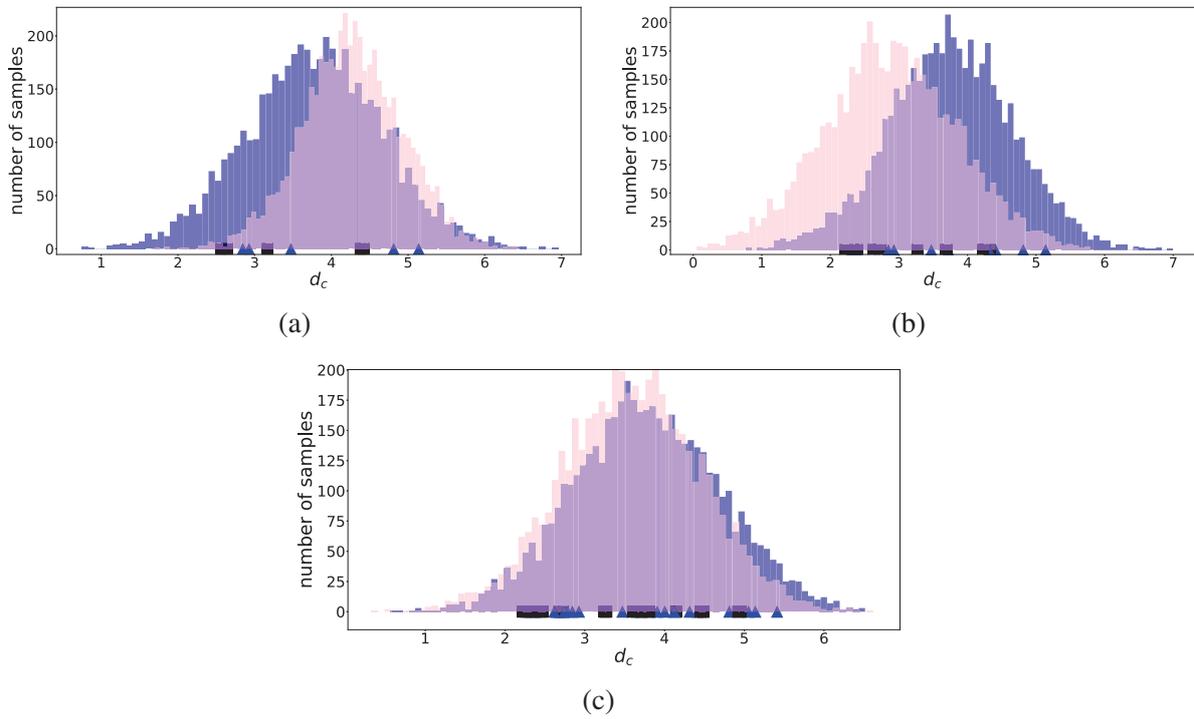


Figure 17: Exact and calibrated marginals along the d_c axis with 5 individuals (a), 10 individuals (b) and 20 individuals (c).

C Bounds of optimization for the Laplace approach

Table 12: Bounds for mean parameters.

	E_0 [MPa]	y_{0s} [MPa]	y_{cs} [MPa]	d_c
Lower bound	1.40×10^5	5.00×10^{-3}	2.8	2.75
Upper bound	2.2×10^5	5.00×10^{-2}	5.50	8.00

Table 13: Bounds for standard deviation parameters.

	σ_{E_0} [MPa]	$\sigma_{y_{0s}}$ [MPa]	$\sigma_{y_{cs}}$ [MPa]	σ_{d_c}	σ_{ω} [MPa]
Lower bound	8.00×10^3	5.00×10^{-5}	0.100	0.005	0.500
Upper bound	1.80×10^4	1.00×10^{-4}	1.2	0.9	2.00

D Exact and noisy data

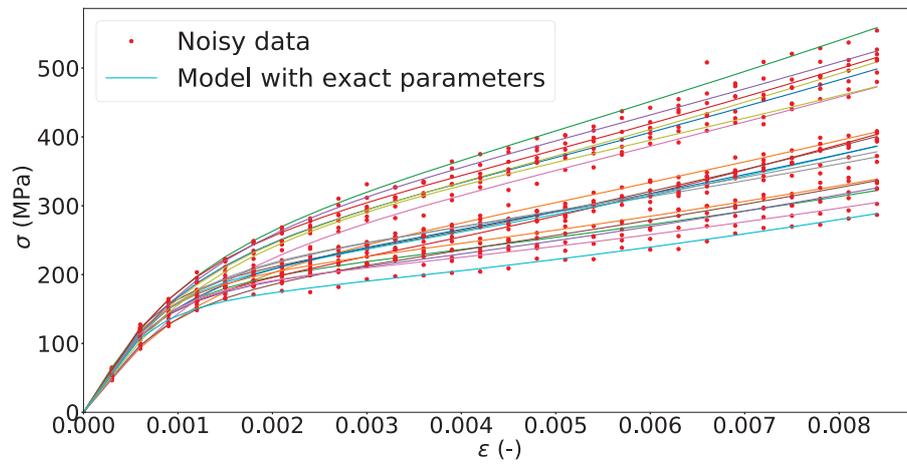


Figure 18: Model with exact individual parameters and noisy data.

E Results of calibration for all experiments from CERASEP A400 material

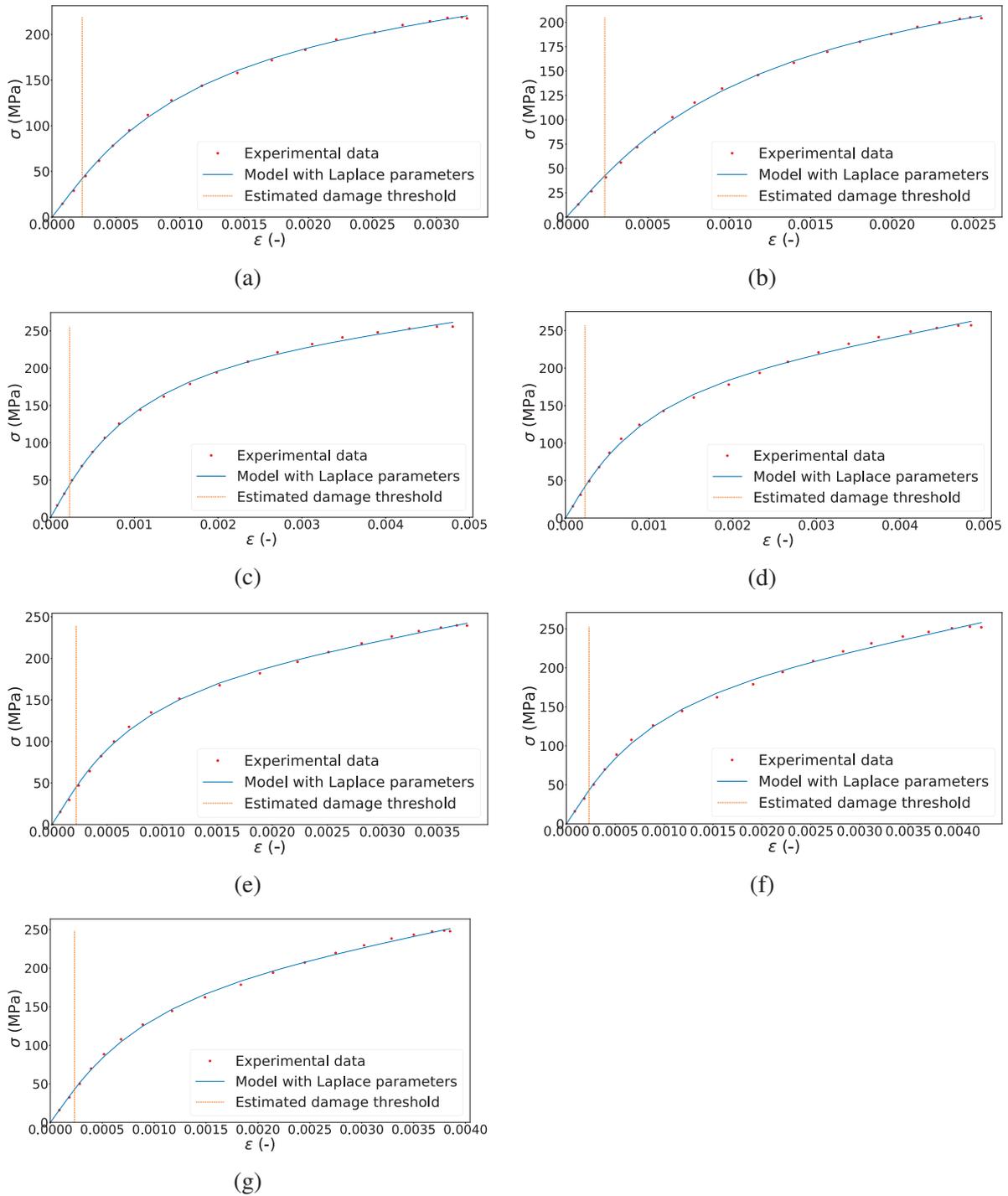


Figure 19: Model output vs experimental data for the first experiment (a), the second experiment (b), third experiment (c), fourth experiment (d), fifth experiment (e), sixth experiment (f) and the seventh experiment (g).

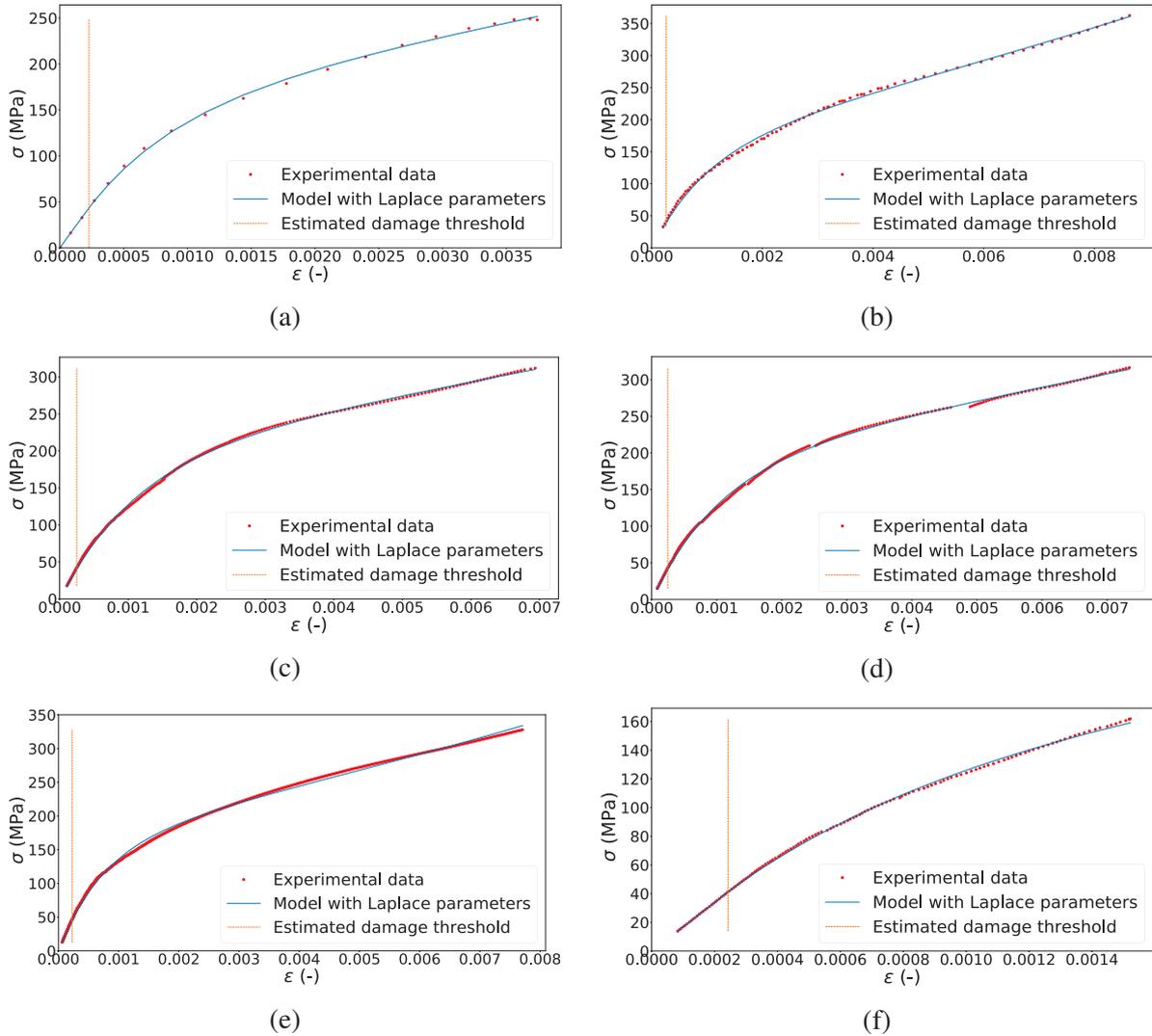


Figure 20: Model output vs experimental data for the eighth experiment (a), the ninth experiment (b), the tenth experiment (c), the eleventh experiment (d), the twelfth experiment (e), and the thirteenth experiment (f).

REFERENCES

- [1] A. Efstratiadis and D. Koutsoyiannis, “One decade of multi-objective calibration approaches in hydrological modelling: a review.,” *Hydrological Sciences Journal*, vol. 55, 2010.
- [2] A. Anestis and C. René, *Régression non linéaire et applications. (in french)*. Économie et statistiques avancées., Paris: Economica, 1992.
- [3] E. Anane, D. C. Lopez C, T. Barz, G. Sin, K. V. Gernaey, P. Neubauer, and M. N. Cruz Bournazou, “Output uncertainty of dynamic growth models: effect of uncertain parameter estimates on model reliability.,” *Biochemical Engineering Journal*, vol. 150, p. 107247, 2019.
- [4] H. Rappel, L. Beex, J. Hale, L. Noels, and S. Bordas, “A tutorial on bayesian inference to identify material parameters in solid mechanics.,” *Archives of Computational Methods in Engineering*, 2019.
- [5] U. Defense, T. P. Company, M. S. Corporation, A. S. for Testing, and Materials, *Composite Materials Handbook-MIL 17, Volume III: Materials Usage, Design, and Analysis*. The Composite Materials Handbook-MIL 17, Taylor & Francis, 1999.
- [6] M. Gallagher and J. Doherty, “Parameter estimation and uncertainty analysis for a watershed model.,” *Environmental Modelling & Software*, vol. 22, no. 7, pp. 1000–1020, 2007.
- [7] A. Tarantola, *Inverse problem theory and methods for model parameter estimation*. Society for Industrial and Applied Mathematics, 1 ed., 2005.
- [8] M. C. Kennedy and A. O’Hagan, “Bayesian calibration of computer models.,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 3, pp. 425–464, 2001.
- [9] L. Wasserman, *All of Statistics: A Concise Course in Statistical Inference*. Springer Publishing Company, Incorporated, 2010.
- [10] M. Merriman, *A List of Writings Relating to the Method of Least Squares: With Historical and Critical Notes*. Transactions of the Connecticut Academy of Arts and Sciences, Academy, 1877.
- [11] H. T. Banks, S. Hu, and W. C. Thompson, *Modeling and Inverse Problems in the Presence of Uncertainty*. Chapman and Hall/CRC, 2014.
- [12] P. Janssen and P. Heuberger, “Calibration of process-oriented models.,” *Ecological Modelling*, vol. 83, no. 1, pp. 55–66, 1995.
- [13] R. Fisher and K. Pearson, *On an Absolute Criterion for Fitting Frequency Curves*. Messenger of mathematics, 1911.
- [14] G. Young, “Mathematical statistics: An introduction to likelihood based inference.,” *International Statistical Review*, vol. 87, pp. 178–179, 2019.
- [15] J. A. Vrugt, H. V. Gupta, L. A. Bastidas, W. Bouten, and S. Sorooshian, “Effective and efficient algorithm for multiobjective optimization of hydrologic models: multiobjective optimization of hydrologic models.,” vol. 39, no. 8, 2003.

- [16] Y. Collette and P. Siarry, *Optimisation multiobjectif. (in french)*, vol. 44 of *Algorithmes (Paris)*. Eyrolles, 2002.
- [17] R. T. Bradley Efron, *An introduction to bootstrap*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability, Chapman & Hall, 1 ed., 1994.
- [18] B. Efron, “Bootstrap methods: Another look at the jackknife.,” *The Annals of Statistics*, vol. 7, no. 1, pp. 1–26, 1979.
- [19] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*. Springer-Verlag, 2005.
- [20] J. S. Liu, *Monte Carlo Strategies in Scientific Computing*, vol. 44. 2002.
- [21] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, “Equation of state calculations by fast computing machines.,” *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.
- [22] W. K. Hastings, “Monte Carlo sampling methods using Markov Chains and their applications.,” *Biometrika*, vol. 57, p. 14, 1970.
- [23] S. Avril, M. Bonnet, A. S. Bretelle, M. Grédiac, F. Hild, P. Ienny, F. Latourte, D. Lemosse, S. Pagano, E. Pagnacco, and F. Pierron, “Overview of identification methods of mechanical parameters based on full-field measurements.,” *Experimental Mechanics*, vol. 48, pp. 381–402, 2008.
- [24] J. Isenberg, “Progressing from least squares to bayesian estimation.,” *Proc. of ASME Design Engineering Technical Conferences*, 1979.
- [25] W. Chongshuai, H. Yiqian, and Y. Haitian, “A SBFEM and sensitivity analysis based algorithm for solving inverse viscoelastic problems.,” vol. 106, pp. 588–598, 2019.
- [26] K. Solanki, M. Horstemeyer, W. Steele, Y. Hammi, and J. Jordon, “Calibration, validation, and verification including uncertainty of a physically motivated internal state variable plasticity and damage model.,” *International Journal of Solids and Structures*, vol. 47, no. 2, pp. 186–203, 2010.
- [27] C. Gogu, R. Haftka, R. Le Riche, J. Molimard, and A. Vautrin, “Introduction to the bayesian approach applied to elastic constants identification.,” *AIAA Journal*, vol. 48, no. 5, pp. 893–903, 2010.
- [28] C. Gogu, W. Yin, R. Haftka, P. Ifju, J. Molimard, R. L. Riche, and A. Vautrin, “Approche bayésienne pour gérer les incertitudes dans l’identification à partir de mesures de champ.(in french).,” p. 6, 2011.
- [29] P. Liu and S.-K. Au, “Bayesian parameter identification of hysteretic behavior of composite walls.,” *Probabilistic Engineering Mechanics*, vol. 34, p. 101109, 2013.
- [30] F. Rizzi, M. Khalil, R. E. Jones, J. A. Templeton, J. T. Ostien, and B. L. Boyce, “Bayesian modeling of inconsistent plastic response due to material variability.,” p. 18, 2019.
- [31] H. Rappel, L. A. A. Beex, and S. P. A. Bordas, “Bayesian inference to identify parameters in viscoelasticity.,” vol. 22, no. 2, pp. 221–258, 2015.

- [32] M. Lavielle, *Mixed Effects Models for the Population Approach: Models, Tasks, Methods and Tools*. 2014.
- [33] J.-F. Maire, P. M. Lesne, and R. Girard, “An explicit behavioural damage model for the design of components in ceramic matrix composites.,” *Key Engineering Materials*, vol. 127-131, pp. 1053–1060, 1997.
- [34] L. Marcin, J.-F. Maire, N. Carrre, and E. Martin, “Development of a macroscopic damage model for woven ceramic matrix composites.,” *International Journal of Damage Mechanics*, vol. 20, no. 6, pp. 939–957, 2011.
- [35] C. B. Ramdane, “étude et modélisation du comportement mécanique de CMC oxyde/oxyde. (in french).,” *Thèse de doctorat de l’ Université de Bordeaux I*, 2014.
- [36] R. Naslain, “Design, preparation and properties of non-oxide cmcs for application in engines and nuclear reactor: An overview,” *Composites Science and Technology*, vol. 64, pp. 155–170, 2004.
- [37] J. Viricelle, P. Goursat, and D. Bahloul-Hourlier, “Oxidation behaviour of a multi-layered ceramic-matrix composite (sic)f/c/(sibc)m.,” *Composites Science and Technology*, vol. 61, no. 4, pp. 607–614, 2001.
- [38] R. A. Fisher, “The correlation between relatives on the supposition of mendelian inheritance.,” 1919.
- [39] R. Drikvandi, “Nonlinear mixed-effects models for pharmacokinetic data analysis: assessment of the random-effects distribution.,” *Journal of Pharmacokinetics and Pharmacodynamics*, vol. 44, 2017.
- [40] C. R. Henderson, O. Kempthorne, S. R. Searle, and C. M. von Krosigk, “The estimation of environmental and genetic trends from records subject to culling.,” *Biometrics*, vol. 15, no. 2, pp. 192–218, 1959.
- [41] C. R. Henderson, “Best linear unbiased estimation and prediction under a selection model.,” *Biometrics*, vol. 31, no. 2, pp. 423–447, 1975.
- [42] N. M. Laird and J. H. Ware, “Random-effects models for longitudinal data.,” *Biometrics*, vol. 38, no. 4, pp. 963–974, 1982.
- [43] M. J. Lindstrom and D. M. Bates, “Nonlinear mixed effects models for repeated measures data.,” *Biometrics*, vol. 46, no. 3, pp. 673–687, 1990.
- [44] E. Demidenko, *Mixed models. Theory and applications with R. 2nd ed.* 2013.
- [45] M. Davidian and D. Giltinan, *Nonlinear Models for Repeated Measurement Data*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability, Taylor & Francis, 1995.
- [46] N. Metropolis and S. Ulam, “The Monte Carlo method.,” *Journal of the American Statistical Association*, vol. 44, no. 247, pp. 335–341, 1949.

- [47] E. Kuhn and M. Lavielle, “Coupling a stochastic approximation version of em with a mcmc procedure.,” *Probability and Statistics*, vol. 8, 2004.
- [48] J. Pinheiro and D. Bates, *Mixed-Effect Models in S and S-plus.*, vol. 96. 2002.
- [49] M. Wand and M. Jones, *Kernel Smoothing*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability, Taylor & Francis, 1994.
- [50] A. Azevedo-Filho and R. Shachter, “Laplace’s method approximations for probabilistic inference in belief networks with continuous variables.,” pp. 28–36, 1994.
- [51] D. Bates, D. Hamilton, and D. Watts, “Calculation of intrinsic and parameter-effects curvatures for nonlinear regression models.,” *Communications in Statistics - Simulation and Computation*, vol. 12, pp. 469–477, 1983.
- [52] G. Stegmann, R. Jacobucci, J. Harring, and K. Grimm, “Nonlinear mixed-effects modeling programs in R.,” *Structural Equation Modeling: A Multidisciplinary Journal*, vol. 25, pp. 1–6, 2017.
- [53] E. Comets, A. P. Lavenu, and M. Lavielle, “Parameter estimation in nonlinear mixed effect models using saemix, an R implementation of the SAEM algorithm.,” *Journal of Statistical Software*, vol. 80, no. 3, 2017.
- [54] D. Packwood, *Bayesian Optimization for Materials Science*. SpringerBriefs in the Mathematics of Materials 3, Springer Singapore, 1 ed., 2017.
- [55] 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.,” *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979.
- [56] C. Rasmussen, O. Bousquet, U. Luxburg, and G. Rtsch, “Gaussian processes in machine learning.,” *Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2 - 14, 2003, Tbingen, Germany, August 4 - 16, 2003, Revised Lectures, 63-71 (2004)*, vol. 3176, 2004.
- [57] D. Jones, M. Schonlau, and W. Welch, “Efficient global optimization of expensive black-box functions.,” *Journal of Global Optimization*, vol. 13, pp. 455–492, 1998.
- [58] S. Kullback and R. A. Leibler, “On information and sufficiency.,” *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79 – 86, 1951.
- [59] D. Foreman-Mackey, D. W. Hogg, D. Lang, and J. Goodman, “emcee: The mcmc hammer.,” vol. 125, no. 925, p. 306, 2013.
- [60] N. Hansen, S. Müller, and P. Koumoutsakos, “Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES).,” *Evolutionary computation*, vol. 11, pp. 1–18, 2003.
- [61] GPy, “A Gaussian process framework in python,” 2012.

ADAPTIVE SEQUENTIAL SAMPLING FOR POLYNOMIAL CHAOS EXPANSION

Lukáš Novák¹, Miroslav Vořechovský¹, and Václav Sadílek¹

¹Brno University of Technology
Veveří 331/95, Brno 60200, Czech Republic
e-mail: {novak.l, vorechovsky.m, sadilek.v}@fce.vutbr.cz

Keywords: Polynomial Chaos Expansion, Adaptive Sampling, Sequential Sampling.

Abstract. *The paper presents a sampling strategy created specifically for surrogate modeling via polynomial chaos expansion. The proposed method combines adaptivity of surrogate model and sequential sampling enabling one-by-one extension of an experimental design. The iteration process of sequential sampling selects from a large pool of candidate points by trying to cover the design domain proportionally to their local variance contribution. The criterion for the sample selection balances between exploitation of the surrogate model and exploration of the design domain. The obtained numerical results confirm its superiority over standard non-sequential approaches in terms of surrogate model accuracy and estimation of the output variance.*

1 INTRODUCTION

Uncertainty quantification of mathematical model of physical system $Y = g(X)$ is attracting an increasing attention in the last decades. Since the quantity of interest (QoI) Y can be output of very computationally demanding model such as complex engineering structures solved by finite element method, it is often necessary to create an approximation of original mathematical model which significantly computationally cheaper. Moreover, it is beneficial to choose approximation which enables direct post-processing in order to obtain statistical moments and sensitivity indices without additional computational demands. Therefore, the paper is focused on a popular method: Polynomial Chaos Expansion (PCE). Although PCE is very accurate surrogate model, its accuracy is highly dependent on design of experiments (DOE). The paper presents a novel approach for sequential extension of the experimental design based on adaptively refined PCE. The proposed approach significantly reduces the number of samples in experimental design to achieve a good surrogate model and thus it reduces the necessary number of evaluations of the original mathematical model.

2 POLYNOMIAL CHAOS EXPANSION

Evaluation of mathematical model of QoI is often highly computationally demanding and thus it is necessary to create an efficient approximation. PCE is a method of representing the output variable Y as a function g^{PCE} of an another random variable ξ called the germ with given distribution

$$Y = g(X) \approx g^{PCE}(\xi), \quad (1)$$

and representing the function $g(X)$ via polynomial expansion. A set of polynomials, orthogonal with respect to the probability distribution of the germ, are used as a basis of the Hilbert space of all real-valued random variables of finite variance. The orthogonality condition for all $j \neq k$ is given by the inner product of the Hilbert space defined for any two functions ψ_j and ψ_k with respect to the weight function p_ξ (probability density function of ξ) as:

$$\langle \psi_j, \psi_k \rangle = \int \psi_j(\xi) \psi_k(\xi) p_\xi(\xi) d\xi = 0. \quad (2)$$

Orthogonal polynomials ψ corresponding to a selected probability distributions p_ξ can be chosen according to Wiener-Askey scheme [12]. For further processing, it is common to use normalized polynomials, where the inner product is equal to the Kronecker delta δ_{jk} , i.e. $\langle \psi_j, \psi_k \rangle = \delta_{jk}$, where $\delta_{jk} = 1$ if and only if $j = k$, and $\delta_{jk} = 0$ otherwise.

In the case of \mathbf{X} and $\boldsymbol{\xi}$ being vectors containing M random variables, the polynomial $\Psi(\boldsymbol{\xi})$ is multivariate and it is built up as a tensor product of univariate orthogonal polynomials. The quantity of interest (QoI), i.e. the response of the mathematical model $Y = g(\mathbf{X})$, can then be represented, according to Ghanem and Spanos [5], as

$$Y = g(\mathbf{X}) = \sum_{\boldsymbol{\alpha} \in \mathbb{N}^M} \beta_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}), \quad (3)$$

where $\boldsymbol{\alpha} \in \mathbb{N}^M$ is a set of integers called the *multi-index*, $\beta_{\boldsymbol{\alpha}}$ are deterministic coefficients and $\Psi_{\boldsymbol{\alpha}}$ are multivariate orthogonal polynomials.

For practical computation, PCE expressed in Eq. (3) must be truncated to a finite number of terms P . The truncation is commonly achieved by retaining only terms whose total degree $|\boldsymbol{\alpha}|$

is less than or equal to a given p . Therefore, the truncated set of PCE terms is then defined as

$$\mathcal{A}^{M,p} = \left\{ \boldsymbol{\alpha} \in \mathbb{N}^M : |\boldsymbol{\alpha}| = \sum_{i=1}^M \alpha_i \leq p \right\}. \quad (4)$$

Additional reduction of the truncated set was proposed by Blatman and Sudret [2] as a ‘‘hyperbolic’’ truncation scheme. Such an approach leads to a dramatic reduction in the cardinality of the truncated set for high total polynomial orders p .

From a statistical point of view, truncated PCE is a simple linear regression model with intercept. Therefore, it is possible to use ordinary least square (OLS) regression to minimize the error ε . In order to use OLS for $\boldsymbol{\beta}$ estimation, it is necessary to first sample n_{sim} realizations of the input random vector \mathbf{X} and the corresponding results of the original mathematical model \mathcal{Y} , together called the experimental design (ED). Then, the vector of deterministic coefficients $\boldsymbol{\beta}$ is calculated using data matrix $\boldsymbol{\Psi}$ as

$$\boldsymbol{\beta} = (\boldsymbol{\Psi}^T \boldsymbol{\Psi})^{-1} \boldsymbol{\Psi}^T \mathcal{Y}. \quad (5)$$

The number of terms P is dependent on the number of input random variables M and the maximum total degree of polynomials p . Therefore, in case of a large stochastic model, the problem can become computationally highly demanding. The solution can utilize advanced model selection algorithms such as Least Angle Regression (LAR) [3] to find an optimal set of PCE terms as proposed by Blatman and Sudret [2]. Note that, similar techniques such as orthogonal matching pursuit [11] or Bayesian compressive sensing [7] achieve similar numerical results. The sparse set of basis functions obtained by any adaptive algorithm is further denoted for the sake of clarity as \mathcal{A} .

3 ADAPTIVE SEQUENTIAL SAMPLING

The accuracy of PCE is unfortunately highly dependent on given experimental design similarly as in case of any surrogate model. Although there are many sampling schemes suitable for PCE, the recent study [4] shows an advantage of sequential approach. Therefore this paper is focused on iterative selection of the new sampling points according to specific criteria created particularly for PCE. Note that there are two different strategies for sequential sampling. The first is to enrich the initial ED according to a space-filling criterion (exploration) without assuming any knowledge of the mathematical model or PCE form. The second strategy works with the structure of the PCE in order to identify an optimal sample. Unfortunately, in situations when the initial screening overlooks a globally important region, the exploitation criterion may continue refinement of some other, locally important region that was detected, and there is a risk of never discovering a globally important region. Therefore, it is important to include a balance between both criteria in search for the best candidate. Note that, such approach was employed already in a different context [10]: a criterion motivated by the Koksma-Hlawka inequality [8] was proposed and coupled with stratified sampling in order to improve the efficiency of statistical integration. Beside sequential sampling, the ideal algorithm should be able to adaptively reconstruct the PCE using model selection algorithms in order to identify a sparse set of basis functions \mathcal{A} in each iteration.

We propose an adaptive sequential sampling strategy accompanied by a criterion designed for non-intrusive PCE. Once a pool of candidates containing n_{pool} realizations of the random vector $\boldsymbol{\xi}$ generated by any sampling technique is available, it is necessary to construct a criterion

Θ for the selection of the best candidate balancing between the exploitation and exploration of the design domain:

$$\Theta(\boldsymbol{\xi}^{(c)}) \equiv \sqrt{\sigma_{\mathcal{A}}^2(\boldsymbol{\xi}^{(c)}) \cdot \sigma_{\mathcal{A}}^2(\boldsymbol{\xi}^{(s)})} l_{c,s}^M. \quad (6)$$

The criterion is a product of two terms: the exploitation term and the exploration term. The exploration aspect is maintained by accounting for the distance $l_{c,s}$ between a candidate $\boldsymbol{\xi}^{(c)}$ and its nearest neighboring point from the existing ED, $\boldsymbol{\xi}^{(s)}$. For the distance term we select the Euclidean distance between the candidate and its nearest neighbor as

$$l_{c,s} = \sqrt{\sum_{i=1}^M |\xi_i^{(c)} - \xi_i^{(s)}|^2}. \quad (7)$$

The exploitation in candidate selection is motivated by our desire to uniformly cover local contributions to the total variance, σ_Y^2 . The variance can be thought of as an integral of local contributions $\sigma_{\mathcal{A}}^2(\boldsymbol{\xi})$ over the design domain indexed by coordinates $\boldsymbol{\xi}$. Once the PCE has been established at any given stage of the algorithm, the local variance is computationally cheap to evaluate for any location $\boldsymbol{\xi}$ as

$$\sigma_{\mathcal{A}}^2(\boldsymbol{\xi}) = \left[\sum_{\substack{\alpha \in \mathcal{A} \\ \alpha \neq 0}} \beta_{\alpha} \Psi_{\alpha}(\boldsymbol{\xi}) \right]^2 p_{\xi}(\boldsymbol{\xi}). \quad (8)$$

In the criterion we take the geometric mean of two local variance contributions representing average variance contribution of the region between the candidate and its nearest neighbor. When this geometric mean is multiplied by the M th power of the distance between the two points, $l_{c,s}^M$, the volume (variance contribution) of an area between them is estimated. The proposed criterion maintains a balance between exploration and exploitation, since a candidate which is close to an existing point can only be selected if the corresponding variance density is significant. Similarly, when a region with low contribution is being detected by the PCE, candidates from such regions are ignored. Maximization of the proposed criterion leads to the best candidate, which is added to active ED. The pool of candidates can be generated by commonly known LHS and the proposed criterion is employed for the selection of the best candidate at every iteration of sequential sampling.

4 NUMERICAL EXAMPLE

The pilot numerical study is represented by Ishigami function [6]. The function is strongly nonlinear, non-monotonic and presents strong interactions. We set the coefficients as in [9]. Let $\mathbf{X} \sim \mathcal{U}[-\pi, \pi]^3$ and the mathematical model

$$Y = \sin(X_1) + 7 \sin^2(X_2) + 0.1 X_3^4 \sin(X_1). \quad [\sigma_Y \approx 13.844587940] \quad (9)$$

The setup of PCE was as follows: PCE is solved by non-intrusive OLS, a sparse set of the basis functions \mathcal{A} is obtained by LAR with maximum total polynomial order $p = 10$ and $p = 20$. The initial ED for the PCE construction before the first step of the proposed iterative algorithm is generated by LHS and it contains an initial screening design with $n_{\text{sim}} = 10$ realizations of the input random vector. The results are compared in terms of the relative error in variance of QoI

$$\epsilon = \frac{|\sigma - \sigma_Y|}{\sigma_Y}, \quad (10)$$

defined as the absolute deviation of the estimated variance σ from the exact value σ_Y divided by the exact variance; and commonly used leave-one-out error of PCE approximation Q^2 [1]. The calculations were repeated 100 times and the orders of errors (\log_{10}) are depicted in Fig.1. Solid lines represent mean values and the scatters represent $\pm \sigma$ confidence intervals.

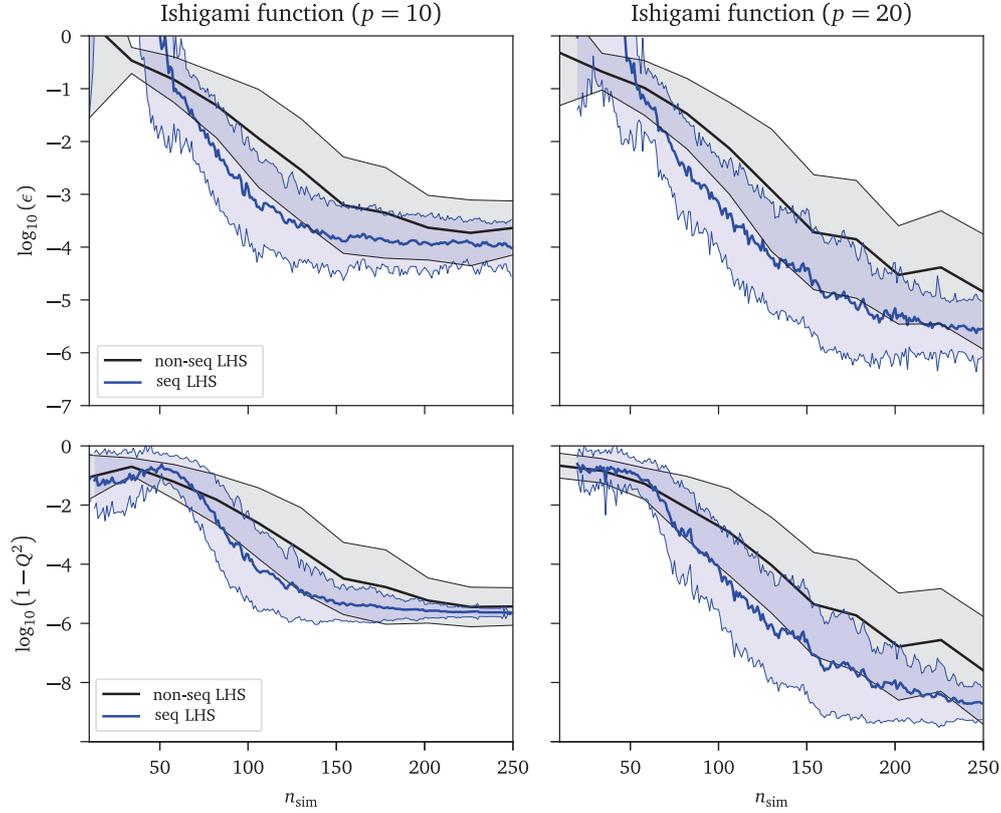


Figure 1: Obtained results for the Ishigami function. The first two row represents the accuracy measured by ϵ and the second row shows Leave-one-out error Q^2 .

5 DISCUSSION AND CONCLUSION

The paper presented innovative approach of adaptive sequential sampling for polynomial chaos expansion. The proposed method combines adaptivity of a PCE and sequential sampling. The sequential sampling is based on one-by-one extension of existing experimental design by a selection of the best candidate from large pool. The best sample candidate is identified by proposed criterion consisting of two parts: average local variance between a candidate “c” and its nearest neighbor “s” and the Euclidean distance between them. Both parts of the criterion together maintain the balance between exploration of the design domain and exploitation of current form of PCE. The presented method can be easily coupled with any existing sampling method such as LHS, which was employed in numerical example. From obtained results, one can see significant improvement in accuracy of PCE using sequential sampling in comparison to standard non-sequential LHS. The improvement is especially clearly visible for mid-size ED. Moreover, the benefit of the sequential sampling is higher in case of $p = 20$ as can be seen in Fig.1 (right) since the maximum polynomial order does not limit the convergence of PCE. Further work will be focused on combination of the sequential approach with advanced sampling schemes.

ACKNOWLEDGMENT

The authors acknowledge financial support provided by the Ministry of Education, Youth and Sports of the Czech Republic under project no. LTAUSA19058. Additionally the first author is supported by BUT internal grant project FAST-J-21-7209.

REFERENCES

- [1] Géraud Blatman and Bruno Sudret. Sparse polynomial chaos expansions and adaptive stochastic finite elements using a regression approach. *Comptes Rendus Mécanique*, 336(6):518–523, 2008.
- [2] Géraud Blatman and Bruno Sudret. Adaptive sparse polynomial chaos expansion based on least angle regression. *Journal of Computational Physics*, 230(6):2345–2367, 2011.
- [3] Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *The Annals of Statistics*, 32(2):407–451, 2004.
- [4] Noura Fajraoui, Stefano Marelli, and Bruno Sudret. Sequential design of experiment for sparse polynomial chaos expansions. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):1061–1085, 2017.
- [5] Roger G. Ghanem and Pol D. Spanos. *Stochastic Finite Elements: A Spectral Approach*. Springer New York, 1991.
- [6] T. Ishigami and T. Homma. An importance quantification technique in uncertainty analysis for computer models. In *Proceedings. First International Symposium on Uncertainty Modeling and Analysis*. IEEE Comput. Soc. Press, 1990.
- [7] Shihao Ji, Ya Xue, and Lawrence Carin. Bayesian compressive sensing. *IEEE Transactions on Signal Processing*, 56(6):2346–2356, 2008.
- [8] J. F. Koksma. Een algemeene stelling uit de theorie der gelijkmatige verdeling modulo 1. *Mathematica B*, 11:7–11, 1942/1943.
- [9] Amandine Marrel, Bertrand Iooss, Béatrice Laurent, and Olivier Roustant. Calculations of sobol indices for the gaussian process metamodel. *Reliability Engineering & System Safety*, 94(3):742–751, 2009.
- [10] Michael D. Shields. Adaptive Monte Carlo analysis for strongly nonlinear stochastic systems. *Reliability Engineering & System Safety*, 175:207–224, July 2018.
- [11] Joel A. Tropp and Anna C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 53(12):4655–4666, 2007.
- [12] Dongbin Xiu and George Em Karniadakis. The Wiener–Askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24(2):619–644, 2002.

AN EXPERIMENTAL STUDY OF VARIABILITY IN DAMPING, FREQUENCY RESPONSE AND MODAL DATA

Asish Kumar Panda¹, Subodh V. Modak²

¹ Indian Institute of Technology Delhi
Hauz Khas, New Delhi 110016
e-mail: Asish.Kumar.Panda@mech.iitd.ac.in

² Indian Institute of Technology Delhi
Hauz Khas, New Delhi 110016
e-mail: svmodak@mech.iitd.ac.in

Abstract

Accurate modelling of the damping in structures, along with the mass and stiffness properties, is important for an accurate prediction of the dynamic response. Also important is modeling of the variability in damping, along with the variability the mass and stiffness properties, from sample to sample if the variability of the dynamic response is to be accurately predicted. The present work is a part of the ongoing efforts in this direction. The objective of this paper is two-fold. The first is to study the variability of the damping factors of various modes of the test structure over its several samples. The second objective is to study the variability when the test structures are made up of different materials. An experimental study is conducted on beam samples of three different materials, Mild steel, Aluminum and Acrylic. Variability in frequency response functions (FRFs), modal data including variability of damping factors is quantified. The study offers some important insights into importance of modeling of damping uncertainty for making accurate structural dynamic predictions.

Keywords: Frequency Response Functions, Variability, Damping Factors, Uncertainty, Experimental Study

1 INTRODUCTION

Finite Element (FE) Models are widely used for analysis of engineering structures. A finite element model that accurately represents the dynamic behavior of a system is very useful for structural dynamic design and analysis, damage detection, structural health monitoring and vibration control [1]. Despite the high sophistication of FE modeling, prediction of dynamic characteristics using FE models often shows considerable discrepancies with respect to the experimental measurements. These discrepancies may arise due to modeling inaccuracies associated with material properties, boundary conditions, joints, damping and due to the idealization and simplifications made in the modeling [1,2]. Measurement noise may also contribute to these discrepancies. The conventional approach to FE model updating is typically deterministic in nature [3–6], which means that the experimental data is obtained from a single test piece and therefore the measured data does not have any variability. It is observed that the engineering structures in practice, all conforming to the same nominal design have a variability associated with them. In view of this, the deterministic FE model cannot accurately represent the dynamics of a population of nominally identical structures. The variability in the measured dynamic characteristics occurs due to inherent variation in the material properties, geometry and manufacturing from specimen to specimen. Variability may also arise due to measurement noise [7,8], environmental effect, damage [9], disassembly and reassembly of the same structure [10] and material degradation over a period of time. It is therefore desired that the FE model is also able to predict the variability of the dynamic characteristics across the samples. This has led to development of stochastic approaches to model updating where both the deterministic as well stochastic uncertainties in the model are identified. Most current methods of Stochastic FE model updating identify uncertainties in parameters associated with the mass and stiffness matrices [8,11,12]. This allows predicting variability of natural frequencies and mode shapes.

However, the question arises about the extent of the variability of dynamic response over the samples of the test structure. The variability of the dynamic response depends on the variability of the damping loss factors. In the light of this, one of the objectives of this paper is to study the extent of variability of damping factor that occurs across several samples of a given structure. The second objective is to study the variability of dynamic characteristics in samples of materials with different levels of damping. This is carried out for beams of three different materials, Mild steel (MS), Aluminum (Al) and Acrylic.

If the variability of the damping loss factors is significant then this requires modeling variabilities not only in the mass and stiffness matrices but also the damping matrix of FE model so as to enable predicting the variability of the dynamic response.

2 EXPERIMENTAL STUDY OF VARIABILITY OF DYNAMIC CHARACTERISTICS

The experiment is carried with three different beam materials, Mild steel, Aluminum and Acrylic. Different samples of beam of each material are tested under free-free condition. FRFs are measured at 16 test points in the transverse direction on each beam sample. An accelerometer is mounted at test point 7 to measure transverse acceleration, while the excitation is given using a modal hammer by hitting at different test points to measure FRFs over a frequency range of 0-1000 Hz. Fig. 1 shows the experimental setup used to measure FRFs. The beam is suspended vertically to ensure that the mass of accelerometer and the stiffness of the string used for suspension does not affect the beam's dynamic characteristics in the transvers direction.

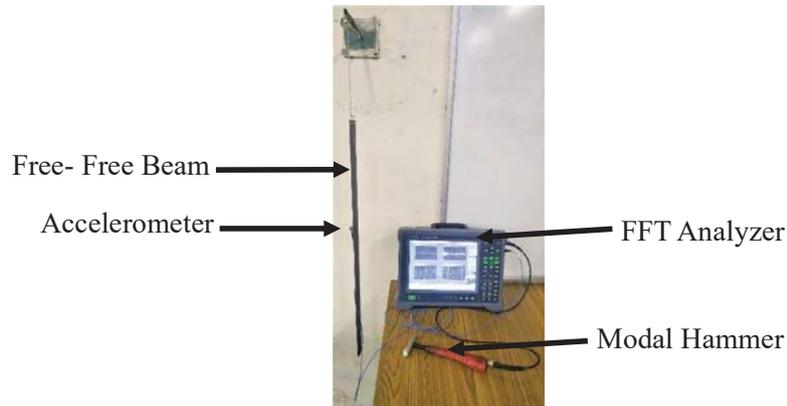
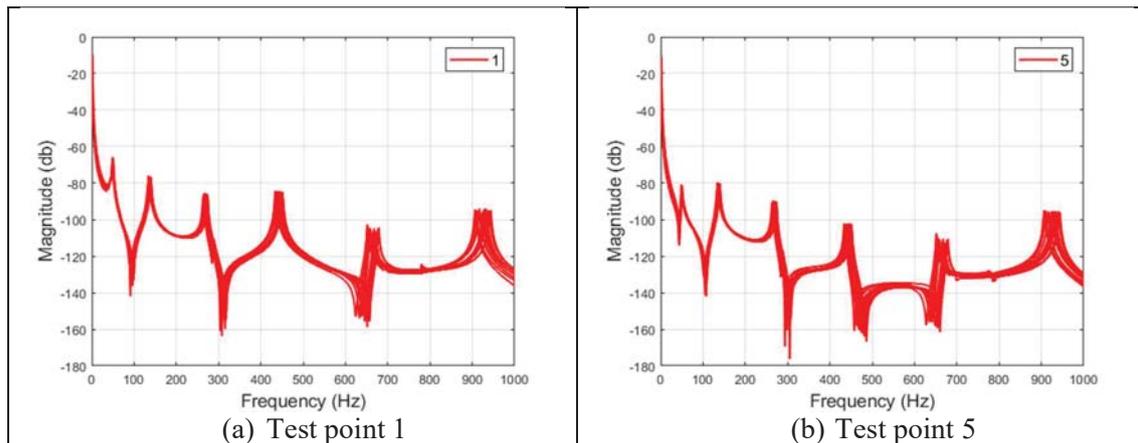


Fig. 1 experimental setup

2.1 Study on Mild steel beam samples

The nominal dimensions of the Mild steel beam samples are (750×31.5×5.3) mm and the mass of the beam is 972 gm. Fig. 2 shows overlays of the experimental FRFs of various samples at test point 1, 5, 7 and 10, respectively. It can be seen that the variability in experimental FRFs is increasing as we go up the frequency range. The modal analysis of measured FRFs on the beam samples is carried out. Figs. 3 and 4 show histograms of variability in natural frequencies and loss factors with the fitted normal distributions, respectively, of the MS beam samples. Increasing the number of samples will increase the accuracy of the distributions. Table 1 shows values of mean, standard deviation and coefficient of variation (COV) of natural frequency and loss factor for the first six modes of vibration of the beam samples. It is seen that the standard deviation of natural frequency is increasing with the mode number but the variation in natural frequency as a percentage of the mean value is nearly same for all the modes listed in the table. It is also found that the natural frequencies for different modes vary on an average by 2.6%, while the loss factors vary on an average by 9.81% over different samples. It is also noted that the COVs of loss factor of higher modes are higher.



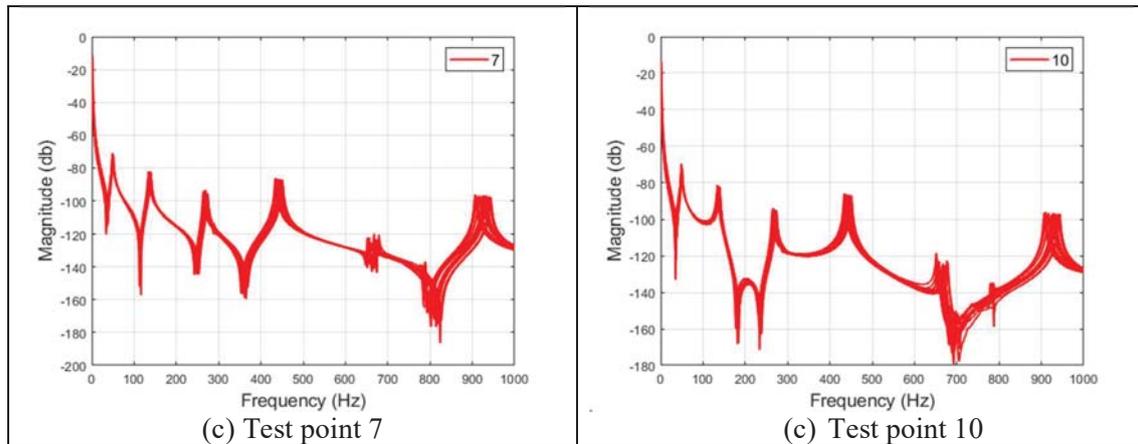
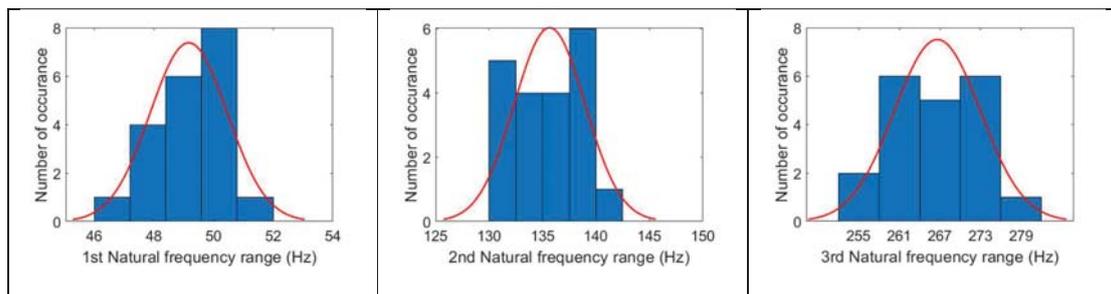


Fig. 2. Overlays of the experimental FRFs measured on various beam samples at test points a) 1 b) 5 c) 7 d) 10

Table 1. Mean, standard deviation and coefficient of variation (COV) of natural frequencies and loss factors

Mode Number	Natural Frequency (Hz)			Loss Factor (%)		
	Mean value (f_{mean})	Standard deviation (f_{std})	Coefficient of variance ($f_{cov} = \frac{f_{std}}{f_{mean}} \times 100$)	Mean value (L_{mean})	Standard deviation (L_{std})	Coefficient of variance ($L_{cov} = \frac{L_{std}}{L_{mean}} \times 100$)
1	49.17	1.29	2.64	0.0421	0.0032	7.72
2	135.66	3.30	2.44	0.0163	0.0007	4.14
3	266.60	6.37	2.39	0.0084	0.0004	4.92
4	438.66	10.60	2.42	0.0055	0.0003	5.35
5	658.84	15.60	2.37	0.0039	0.0008	19.33
6	915.77	21.89	2.39	0.0037	0.0007	17.89



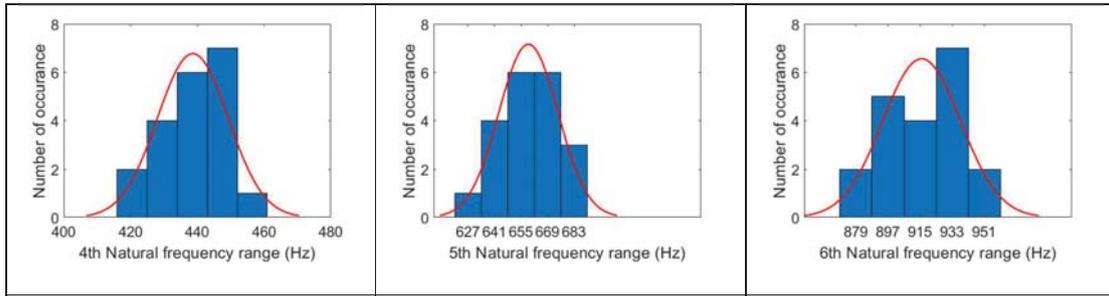


Fig. 3 Histograms of variability in natural frequencies of the MS beam samples

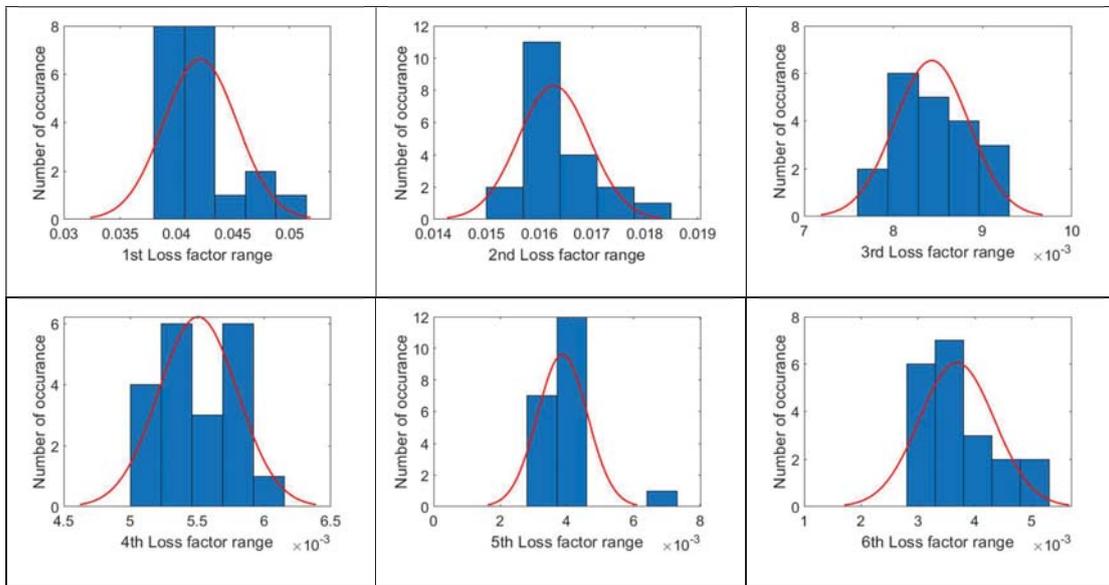


Fig. 4 Histograms of variability in loss factors of the MS beam samples

2.2 Study on Aluminum beam samples

In this section, 20 different Aluminum beam samples with nominal dimensions (750×29.6×5.8) mm are used for the study. The nominal mass of the beam samples is 467.8 gm. Fig. 5 shows overlays of the experimental FRFs of various samples at test points 2, 5, 7 and 11, respectively. It is seen that, like MS beam, the variability in experimental FRFs increases as we go up the frequency range. The modal analysis of measure FRFs on the beam samples is carried out. Figs. 6 and 7 show histograms of variability in natural frequencies and loss factors with the fitted normal distributions, respectively, of the Al beam samples. Table 2 shows values of mean, standard deviation and coefficient of variation (COV) of natural frequency and loss factor for the first six modes of vibration of the beam samples. It is seen that, like MS beam the standard deviation of natural frequency is increasing with the mode number but the variation in natural frequency as a percentage of the mean value is nearly same for all the modes listed in the table. It is also found that the natural frequencies for different modes vary on an average by 1.5%, while the loss factors vary on an average by 14.32% over different samples. It is also noted that the COVs of loss factor of higher modes are higher.

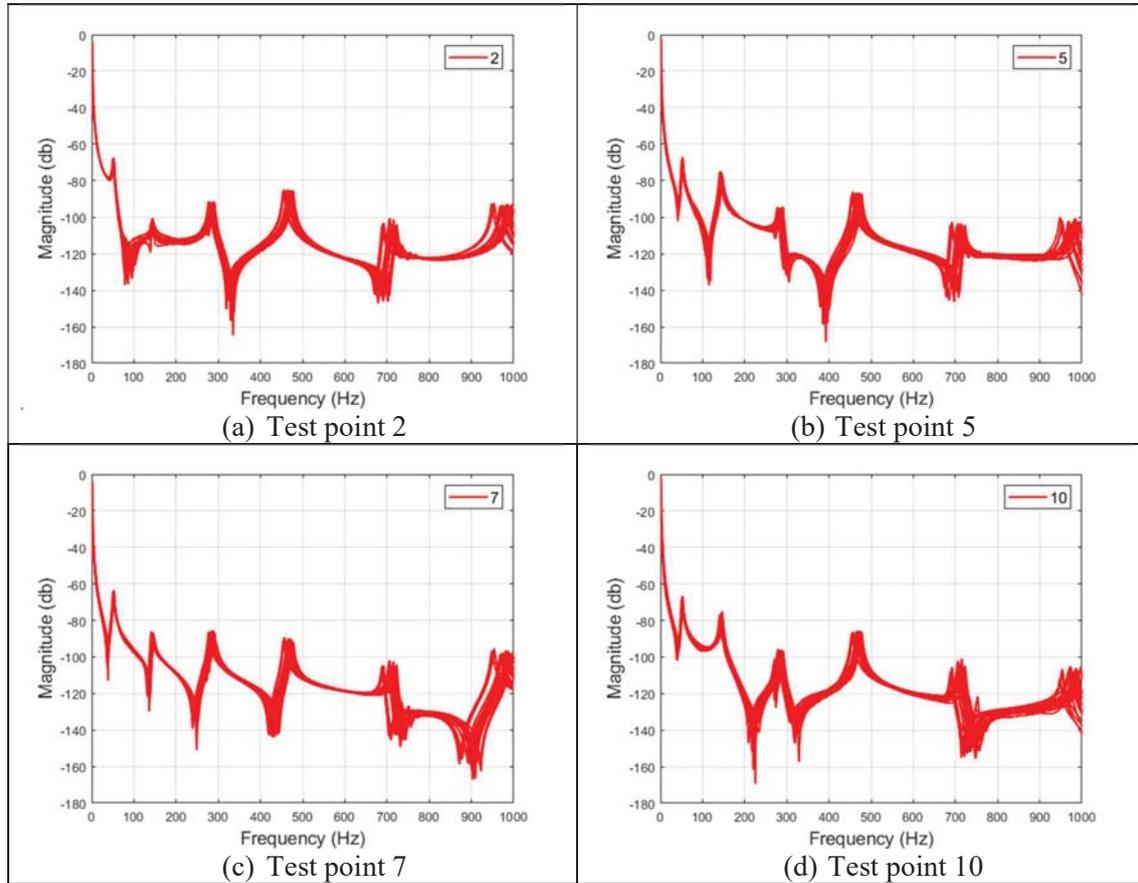


Fig. 5 Overlays of the experimental FRFs measured on various Al beam samples at test points a) 2 b) 5 c) 7 d) 10

Table 2. Mean, standard deviation and coefficient of variation (COV) of natural frequencies and loss factors (Al beams)

Mode Number	Natural Frequency (Hz)			Loss Factor (%)		
	Mean value (f_{mean})	Standard deviation (f_{std})	Coefficient of variance ($f_{cov} = \frac{f_{std}}{f_{mean}} \times 100$)	Mean value (L_{mean})	Standard deviation (L_{std})	Coefficient of variance ($L_{cov} = \frac{L_{std}}{L_{mean}} \times 100$)
1	52.03	0.75	1.44	0.0444	0.00236	5.31
2	144.38	2.13	1.47	0.0026	0.00265	9.82
3	285.80	4.49	1.57	0.0207	0.00570	27.56
4	467.21	7.14	1.53	0.0090	0.0001	11.21
5	707.38	10.48	1.48	0.0058	0.00121	20.94
6	975.65	14.79	1.52	0.006	0.000664	11.05

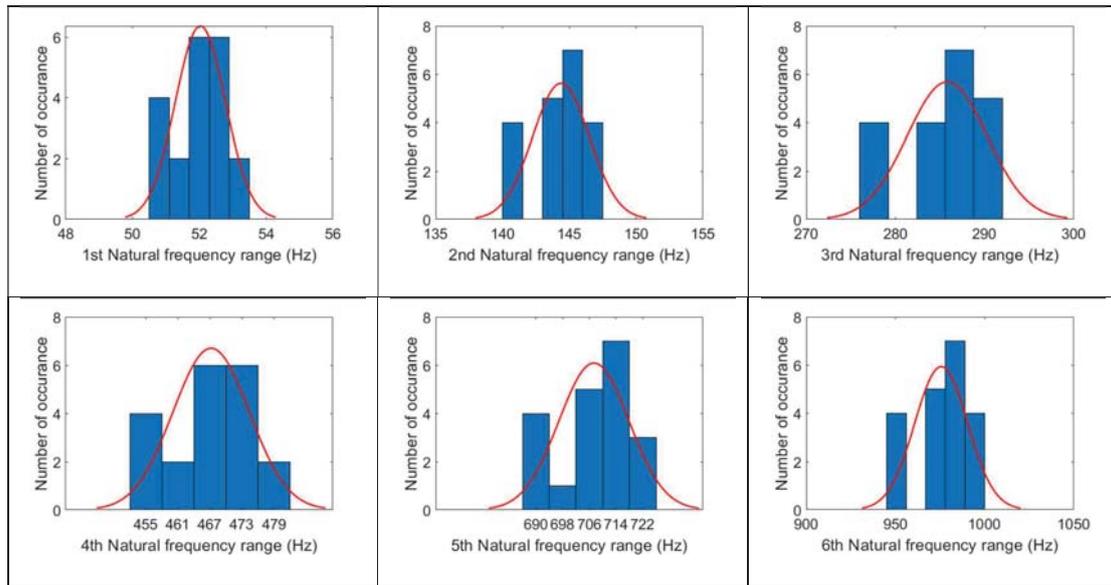


Fig. 6 Histograms of variability in natural frequencies (first six modes) of the Al beam samples

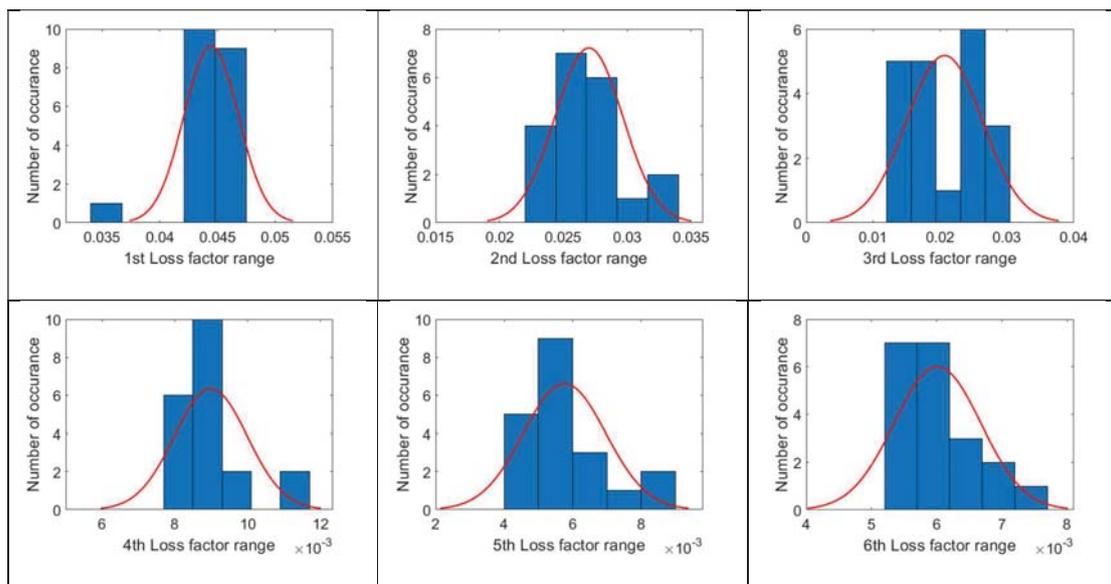


Fig. 7 Histograms of variability in loss factors (first six modes) of the Al beam samples

2.3 Study on Acrylic beam samples

In this section, 20 different Acrylic beam samples with nominal dimensions (750×33.4×3.7) mm are used for the study. The nominal mass of the beam samples is 97 gm. Fig. 8 shows overlays of the experimental FRFs of various samples at test points 3, 5, 7 and 11, respectively. It is seen that, like MS and Al beams, the variability in experimental FRFs increases as we go up the frequency range. The modal analysis of measure FRFs on the beam samples is carried

out. Figs. 9 and 10 show histograms of variability in natural frequencies and loss factors with the fitted normal distributions, respectively, of the Acrylic beam samples. Table 3 shows values of mean, standard deviation and coefficient of variation (COV) of natural frequency and loss factor for the first six modes of vibration of the beam samples. It is seen that, like other two beams, the standard deviation of natural frequency is in-creasing with the mode number but the variation in natural frequency as a percentage of the mean value is nearly same for all the modes listed in the table. It is also found that the natural frequencies for different modes vary on an average by 0.93%, while the loss factors vary on an average by 27.56% over different samples. It is also noted that the COVs of loss factor of higher modes are higher.

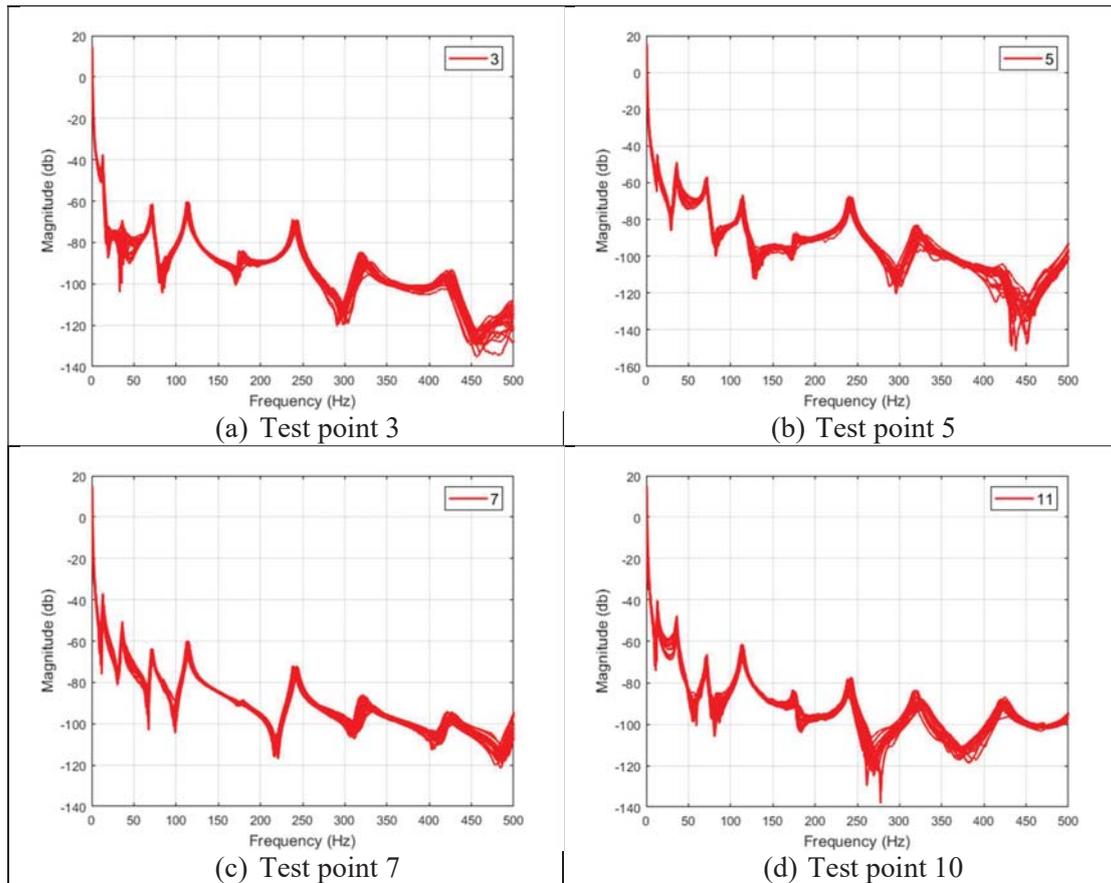


Fig. 8 Overlays of the experimental FRFs measured on various Acrylic beam samples at test points a) 1 b) 5 c) 7 d) 10

Table 3. Mean, standard deviation and coefficient of variation (COV) of natural frequencies and loss factors (Acrylic beams)

Mode Number	Natural Frequency (Hz)			Loss Factor (%)		
	Mean value (f_{mean})	Standard deviation (f_{std})	Coefficient of variance ($f_{cov} = \frac{f_{std}}{f_{mean}} \times 100$)	Mean value (L_{mean})	Standard deviation (L_{std})	Coefficient of variance ($L_{cov} = \frac{L_{std}}{L_{mean}} \times 100$)

1	13.23	0.13	1.03	0.0980	0.01312	13.39
2	35.90	0.40	1.13	0.0616	0.01744	28.32
3	71.41	0.66	0.93	0.0352	0.00810	23.02
4	113.83	1.03	0.90	0.0303	0.00799	26.38
5	179.93	1.16	0.66	0.0332	0.01455	43.84
6	241.80	1.53	0.63	0.0206	0.00545	26.39

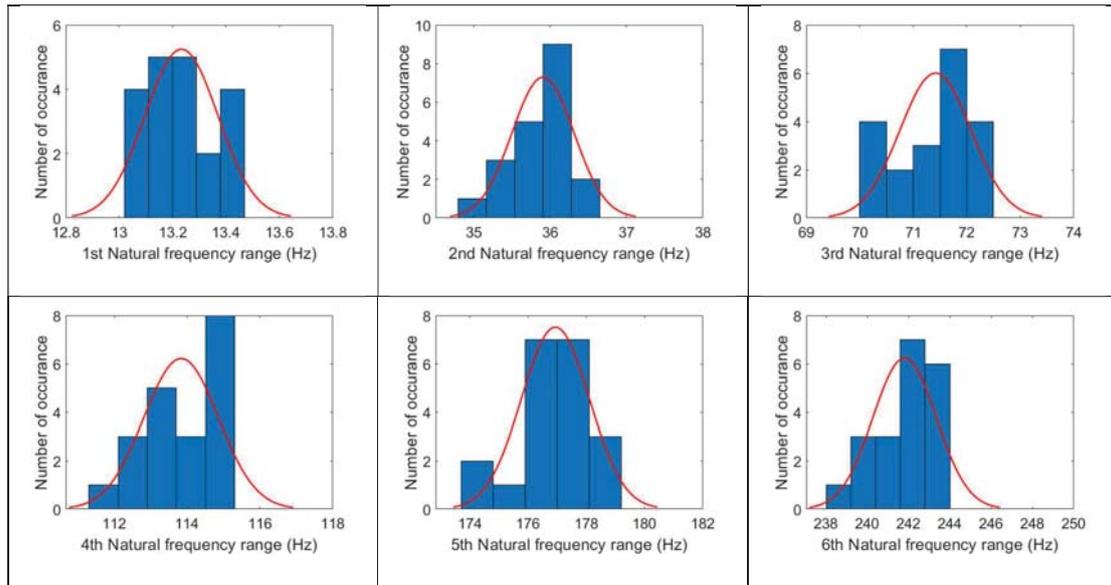


Fig. 9 Histograms of variability in natural frequencies (first six modes) of the Acrylic beam samples

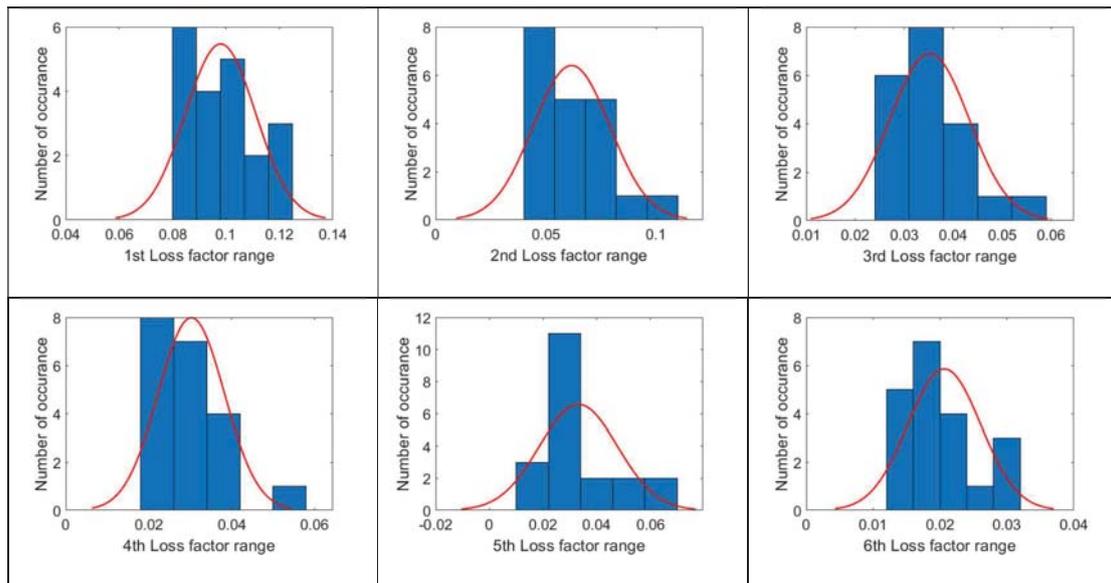


Fig. 10 Histograms of variability in loss factors (first six modes) of the Acrylic beam samples

Table 4. Relative comparison of variability (COV of natural frequencies and loss factors)

Material Type	Variability of natural Frequency		Variability of loss factor	
	Average COV value of six modes (%)	Maximum COV value of six modes (%)	Average COV value of six modes (%)	Maximum COV value of six modes (%)
MS	2.6	2.64	9.81	19.33
Al	1.5	1.57	14.32	27.56
Acrylic	0.93	1.13	27.56	43.84

2.4 Comparison of Variability of dynamic characteristic of MS, Al and Acrylic beams

Table 4 shows a comparison of variabilities of natural frequencies and loss factors of three different beam materials. The table shows average values of COV of natural frequencies and loss factors of six modes. It can be seen that average value of COV of natural frequencies decreases, whereas average value of COV loss factors increases, moving from low damping material (MS) to high damping material (Acrylic). Maximum COV values of natural frequencies are closer to the corresponding average values showing that relative variations of COVs of natural frequencies over different mode of same material are small. Maximum COV values of loss factors has a considerable difference as compared to corresponding average values showing that relative variations of COVs of loss factors over different mode of same material varies a lot. These variation are more in higher modes as compared to lower modes.

3 CONCLUSION

- The paper presents an experimental study of variability in damping, FRFs and modal data. The investigations are carried out using three experimental beam structures made up of MS, Al and Acrylic. They represent structures with low, medium and high damping.
- From the study, it is observed that different samples of structures show the variability of not only the natural frequencies and FRFs but also the modal damping factors. Thus, there is a need to develop methods for stochastic model updating which can also identify variability damping matrix parameters.
- The study also reveals that the test structures are made up of materials with higher damping show higher variability of damping loss factors and lower variability of natural frequencies as compared to structures made up of material with lower damping.
- It is therefore concluded that modeling of variability in damping is essential since there exists a significant variability of damping loss factors over various samples of the test structures. Modeling of damping variability will help to predict accurately a variability of dynamic response. Hence, methods of stochastic FE model updating need to be developed for identifying variability of damping matrix parameters.

REFERENCE

- [1] J.E. Mottershead, M.I. Friswell, Model updating in structural dynamics: A survey, *J. Sound Vib.* (1993). <https://doi.org/10.1006/jsvi.1993.1340>.
- [2] M.I. Friswell, J.E. Mottershead, Finite element model updating in structural dynamics, 1995. <https://doi.org/10.1007/978-94-015-8508-8>.
- [3] F. Asma, A. Bouazzouni, Finite element model updating using FRF measurements, *Shock Vib.* 12 (2005) 377–388. <https://doi.org/10.1155/2005/581634>.
- [4] M.I. Friswell, D.J. Inman, D.F. Pilkey, Direct updating of damping and stiffness matrices, *AIAA J.* 36 (1998) 491–494. <https://doi.org/10.2514/3.13851>.
- [5] S. Pradhan, S. V. Modak, Normal response function method for mass and stiffness matrix updating using complex FRFs, *Mech. Syst. Signal Process.* (2012). <https://doi.org/10.1016/j.ymsp.2012.04.019>.
- [6] S. Pradhan, S.V. Modak, A two-stage approach to updating of mass, stiffness and damping matrices, *Int. J. Mech. Sci.* 140 (2018) 133–150. <https://doi.org/10.1016/J.IJMECSCI.2018.02.033>.
- [7] J.L. Beck, L.S. Katafygiotis, Updating models and their uncertainties I: Bayesian statistical framework, *J. Eng. Mech.* 124 (1998). [https://doi.org/https://doi.org/10.1061/\(ASCE\)0733-9399\(1998\)124:4\(455\)](https://doi.org/https://doi.org/10.1061/(ASCE)0733-9399(1998)124:4(455)).
- [8] X.G. Hua, Y.Q. Ni, Z.Q. Chen, J.M. Ko, An improved perturbation method for stochastic finite element model updating, *Int. J. Numer. Methods Eng.* 73 (2008) 1845–1864. <https://doi.org/10.1002/nme.2151>.
- [9] M. Baruch, I.Y. Bar-Itzhack, Optimal weighted orthogonalization of measured modes, *AIAA J.* 17 (1979) 927–928. <https://doi.org/10.2514/3.7529>.
- [10] M. Imregun, W.J. Visser, D.J. Ewins, Finite element model updating using frequency response function data: I. Theory and initial investigation, *Mech. Syst. Signal Process.* 9 (1995) 187–202. <https://doi.org/10.1006/MSSP.1995.0015>.
- [11] H.H. Khodaparast, J.E. Mottershead, M.I. Friswell, Perturbation methods for the estimation of parameter variability in stochastic model updating, *Mech. Syst. Signal Process.* 22 (2008) 1751–1773. <https://doi.org/10.1016/J.YMSSP.2008.03.001>.
- [12] Y. Govers, M. Link, Stochastic model updating-Covariance matrix adjustment from uncertain experimental modal data, *Mech. Syst. Signal Process.* 24 (2009) 696–706. <https://doi.org/10.1016/j.ymsp.2009.10.006>.

EFFECTIVENESS OF THE PROBABILITY DENSITY EVOLUTION METHOD FOR DYNAMIC AND RELIABILITY ANALYSES OF MASONRY STRUCTURES

Massimiliano Lucchesi¹, Barbara L. Pintucchi¹, and Nicola Zani¹

¹University of Florence, Department of Civil and Environmental Engineering
via S. Marta 3, Florence, Italy
e-mail: {massimiliano.lucchesi, barbara.pintucchi, nicola.zani}@unifi.it

Keywords: Masonry, Tower, Structural dynamics, Mechanical parameters uncertainties, Probability Density Evolution Method.

Abstract. *The main objective of this paper is to examine the possibility of using the probability density evolution theory (PDEM) to determine the evolution of the probability of some structural parameters, during dynamic processes of masonry buildings subjected to seismic actions. The study is mainly motivated by the computational burden that is required by the Monte Carlo method in the case of step-by-step dynamic analyses of structures with large size, complex geometry and a highly non-linear constitutive equation. The PDEM requires the deterministic solution of the dynamic system in a limited number of cases (much lower than that required by the Monte Carlo method), together with the numerical solution of a linear partial differential equation of the first order. First of all, the effectiveness of the method is verified in the case of a simple problem whose explicit solution is known, mainly to determine the most suitable numerical method for solving the differential equation. Then, the dynamic behavior of a masonry tower is analysed. The structure is modeled as a beam with a hollow rectangular section, made of a no-tension material with softening in compression. It is subjected to the action of a real earthquake. The Young's modulus of the material is assumed to be a random variable, and the probability density function of the displacement at the top of the tower is determined throughout the time-history. The results obtained at some time-steps are compared with those provided by the Monte Carlo method. Although the example examined is quite simple, the PDEM appears to be very promising to study more complex masonry structures.*

1 INTRODUCTION

The analyses of structures subjected to dynamic loads are now generally conducted using refined mechanical models and adequate numerical techniques. However, even when the problem is well posed so that both the existence and the uniqueness of the solution are guaranteed, the parameters describing the geometric and mechanical characteristics of the structures and the external actions are generally affected by uncertainties that must be taken into account.

In general, the state equations of the dynamic system - typically obtained by discretizing a structure into finite elements, and the initial joint probability density function (Pdf) of all the considered random variables are assigned. Thus, a stochastic process is obtained that is parametrized over the time, and has the Euclidean space as state space.

In many applications it is necessary to address the problem of determining the Pdf of some quantities of interest at predetermined times. This goal is in most cases achieved using the Monte Carlo method which, at least in principle, allows to consider complex models and geometries without having to resort to unrealistic simplifying hypotheses. On the other hand, this method can require extremely long computational times [1]. For this reason, the probabilistic analyses of masonry structures are rarely conducted via dynamic (time-history) analyses.

A different way to deal with the problem is to use the generalized density evolution equation, which is a consequence of the principle of preservation of probability [2], and leads to writing a linear partial differential equation for any quantity whose Pdf has to be determined. The coefficients of this equations at each instant are function of the state variables, and therefore they can be obtained from the (deterministic) solution of the dynamic system. The probability density evolution method has been implemented into the MADY code, which has already the routines for dynamic analysis of plane, three-dimensional, or beam and shell-based structures [4]. To our knowledge, this method has not been used in studying masonry structures, despite that uncertainties in the constitutive parameters and geometry have particular relevance.

In this paper, the potential of the application of the generalized density evolution method to masonry constructions is investigated. Firstly, different numerical method to solve the density evolution equation have been checked and the time and space step refinements needed have been identified with reference to a simple SDOF problem. Then, some preliminary numerical results have been obtained for a masonry tower represented as a simple beam model, applying a seismic action and accounting for the uncertainties of the Young's modulus. The response in terms of displacement at the top of the tower has been compared with the results obtained via the Monte Carlo method.

2 PROBABILISTIC METHOD

Let

$$M\ddot{Y} + f(\dot{Y}, Y) = B(Y, t)\xi(t), \quad Y(0) = Y_0, \dot{Y}(0) = Y_1 \quad (1)$$

be the discretized equations of motion, where $t \in [0, T]$ is the time, Y , \dot{Y} , \ddot{Y} are the displacement, velocity and acceleration n -dimensional vectors, respectively, M is the mass matrix, f is the internal force vector, B is the input force influence matrix, ξ is the external excitation vector and Y_0 and \dot{Y}_0 are the initial displacement and velocity vector, respectively. Equations (1) can be rewritten as

$$\dot{X} = A(X, t) + \bar{B}(X, t)\xi(t), \quad X(0) = X_0, \quad (2)$$

where

$$X = \begin{pmatrix} \dot{Y} \\ Y \end{pmatrix}$$

is the state vector, and

$$A(X, t) = \begin{pmatrix} -M^{-1}f(Y) \\ \dot{Y} \end{pmatrix}, \quad \bar{B}(X, t) = \begin{pmatrix} -M^{-1}B(Y, t) \\ 0 \end{pmatrix}, \quad X_0 = \begin{pmatrix} Y_1 \\ Y_0 \end{pmatrix}.$$

If randomness are present, coming from the properties of the system, it is assumed that there is a probability space $(\Omega, \mathcal{B}, \mathbb{P})$, where $\Omega \subset \mathbb{R}^m$ is the sample space, a collection of outcomes ω , \mathcal{B} is the Borel σ -algebra of Ω and \mathbb{P} is the probability measure.

If the deterministic problem is well posed then, at each instant t , equations (2) have one and only one solution

$$X_t = H(\theta, t),$$

with $\theta = \Theta(\omega)$ for some smooth real or vector function Θ , with probability density function $p_\Theta(\theta)$. (It is supposed that there are no randomness in the initial conditions and that X_0 has been fixed once and for all). As a consequence of the 'conservation of probability', which implies $\dot{\Theta}=0$, it holds

$$\dot{X}_t = \frac{\partial H(\theta, t)}{\partial t}.$$

We are interested in determining the probability density function $p_Z(z, t)$ of some random variable $Z_t = \phi \circ X_t$, where $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$ is a smooth and injective function. In order to determine function $p_Z(z, t)$, we proceed as proposed in [2]. Let $p_{Z\Theta}(z, \theta, t)$ be the joint probability density function of Z_t and Θ . Then $p_{Z\Theta}(z, \theta, t)$ is the solution of the linear PDE

$$\frac{\partial p_{Z\Theta}(z, \theta, t)}{\partial t} + \dot{Z}_t \frac{\partial p_{Z\Theta}(z, \theta, t)}{\partial z} = 0 \tag{3}$$

with the initial condition

$$p_{Z\Theta}(z, \theta, 0) = \delta(z - z_0)p_\Theta(\theta),$$

where $z_0 = \phi(X_0)$. Once $p_{Z\Theta}(z, \theta, t)$ has been numerically determined by the finite difference method, the marginal probability density function $p_Z(z, t)$ can be obtained by the equation

$$p_Z(z, t) = \int_{\Omega} p_{Z\Theta}(z, \theta, t) d\theta.$$

3 CHOICE OF THE DIFFERENCE SCHEME

In order to choose a difference scheme and select the appropriate parameters for solving Eq. (3), various schemes proposed in literature have been tested.

Specifically, three finite difference schemes have been used, a one-sided, a two-sided (Lax-Wendroff) and a Total Variation Diminishing (TVD) scheme. The latter is a generalization of

the other two, to which it can be traced back with an appropriate choice of parameters. The effectiveness of each scheme has been evaluated with reference to a simple problem for which the explicit solution is available [2][3]. Namely, the problem of an undamped SDOF system with a random frequency, under free oscillations with the initial displacement $x_o = 0.1\text{m}$, and null initial velocity.

In order to guarantee the convergence, the ratio between the step-time Δt and the space mesh size Δx , denoted by λ , has been assumed equal to 0.1. Some investigations have pointed out that, once the stability condition is guaranteed, the accuracy of the results does not significantly depend on the value of λ as well as Δt . Conversely, the choice of Δx as well as the number of the deterministic responses, equal to the number of samples, can appreciably affect the obtained results. In the following, the largest Δx considered is equal to $3.125 * 10^{-3}\text{m}$, together with the corresponding $\Delta t = 3.125 * 10^{-4}\text{m}$. Then, by maintaining the same value of λ , the equation has been solved by assuming a step in space equal to $\Delta x/2$ and $\Delta x/4$. In addition, the solution has been evaluated by considering both 25, 50 and 100 deterministic responses (number of samples).

The obtained results are plotted in the following figures, and compared with the explicit solution. In particular, Figures 1, 2 and 3 show the mean value and standard deviation of x as a function of t obtained respectively with the three method. Figures 4 and 5 show the probability

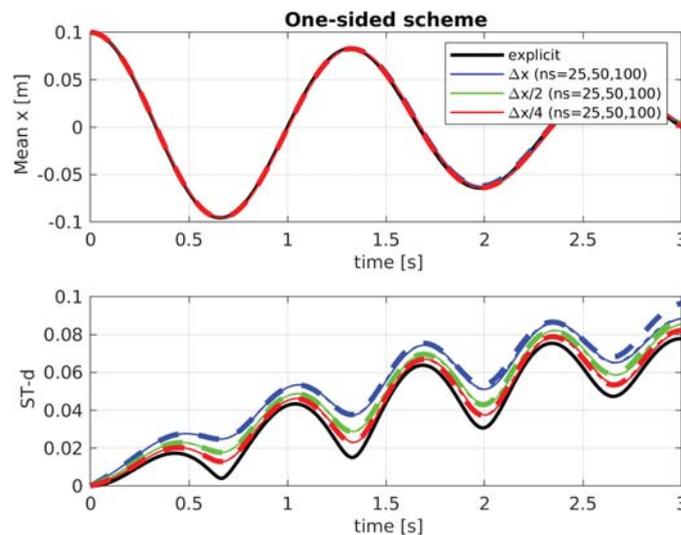


Figure 1: Mean value and standard deviation of x vs t given by the one-sided scheme.

density function, obtained with the three schemes considered at $t=0.9$ s, $t=1$ s and $t=1.1$ s, for the spatial mesh size equal to Δx , $\Delta x/2$ and $\Delta x/4$, while Figure 6 shows the distribution function (Cdf) for $t = 0.9$ s, for the considered different schemes and 100 samples.

Figures 1, 2, 3 evidence that the standard deviation is more sensitive than the mean value to the mesh size, whatever the scheme used. Moreover, it is evidenced that the best results are obtained with the TVD scheme.

This is also confirmed by the results shown in Figures 4 and 5. In particular, Figure 4 shows that the solution obtained with the scheme of Lax-Wendroff oscillates near the discontinuity, also assuming negative values.

Overall, these figures suggest that, for all the schemes used, the best results are obtained from

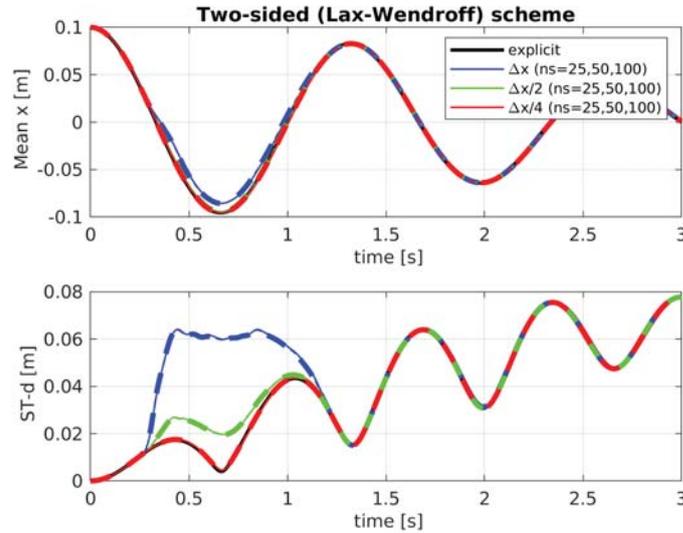


Figure 2: Mean value and standard deviation of x vs t given by the two-sided scheme.

an appropriate combination of the number of samples with the amplitude value of the mesh size. Lastly, it should be noted that the Cdf, shown in Figure 6, is little influenced by the choice of the calculation scheme.

4 DETERMINISTIC MODEL

The model used is a continuous beam model with a hollow rectangular cross-section and distributed mass m . It is implemented into the finite element code MADY and makes use of a constitutive equation formulated in terms of generalized stress and strain, i.e. $N = N(\epsilon, \kappa)$, $M = M(\epsilon, \kappa)$ [5], [6]. This constitutive law has been developed assuming that the sections remain plane, accounting for the axial stress alone σ_z , and describing the masonry behavior by means of a law $\sigma_z = \sigma_z(\epsilon_z)$. In particular, masonry is assumed to have a null tensile strength and a limited compressive strength σ_c . A softening behavior in compression is accounted for, and a linear piecewise law is defined as a function of the strain $\mu = \epsilon_u/\epsilon_c$ (Fig. 7). A damage function α has been defined as a function of the non-dimensionalized generalized strains $\eta = \epsilon/\epsilon_c$, $\chi = \kappa h/\epsilon_c$, such that the damaged beam section has reduced mechanical properties - Young's modulus E and σ_c - with respect to the undamaged one. For the sake of brevity it is not described in detail here, but an analogous procedure can be found in [7], [8], [9] for other types of sections.

The actual geometry of the Torre Grossa of S. Gimignano used as reference real tower has been approximated by the beam model with a rectangular cross-section and an overall height of 50 m [10]. The constraint offered by the neighbouring buildings extending for a height of 20 m from the soil has been modelled with a set of lateral elastic links (Figure 7).

5 RESULTS

The PDEM has been applied to the study of the tower modelled as shown in Fig. 7. The deterministic nonlinear dynamic analyses have been conducted by applying an input ground motion recorded during the Tabas, Iran event of 1978; the accelerogram has a magnitude of 7.4, a duration of 63.40s and a PGA of 0.925g.

The main purpose of the study presented here is to verify the effectiveness of the method for such dynamic systems which, differently from the simple oscillator studied above, have strong

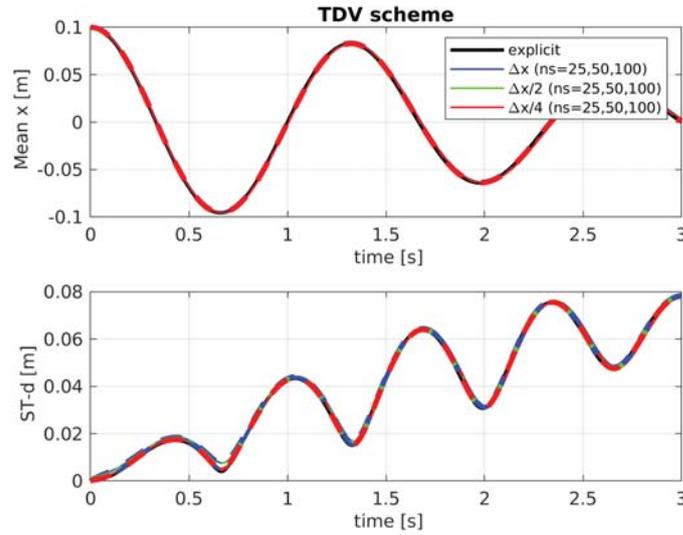


Figure 3: Mean value and standard deviation of x vs t given by the TDV scheme.

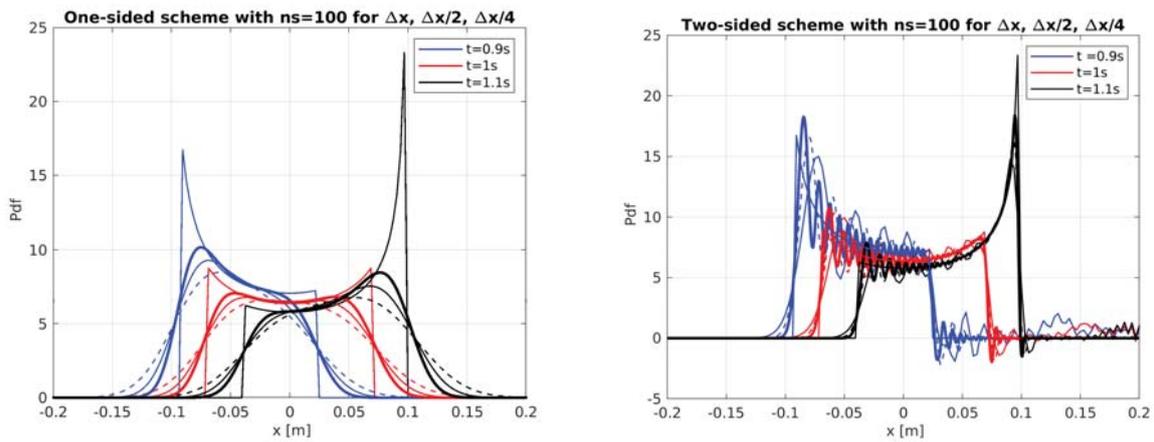


Figure 4: Pdf obtained via the one-sided and two-sided scheme.

non-linearities. On the basis of the results described in the foregoing, the TDV scheme has been used for the integration of equation (3). Moreover, the investigation is limited to determining the trend over time of the Pdf of the displacement at the top of the tower, assigned the Pdf of the Young's modulus of the material, a mechanical parameter that greatly affects the response of masonry structures. For this parameter, a mean value $\mu = 1.8$ GPa and a uniform distribution on the interval $[1.195, 2.405]$ have been assumed. The results obtained with PDEM at some time-steps, for various values of both the number of samples (ns) and the mesh size are compared with those provided by the Monte Carlo Method (MC), with 50,000 samples.

Figure 8 shows the mean and standard deviation as function of time, while Figure 9 shows the Pdf at four steps of time, calculated with the PDEM. Once again, it is observed that the best results are obtained with an appropriate combination of the number of samples and the length of the mesh size. Figure 10 shows the comparison of the Pdfs, obtained with $\Delta x/4$ and $ns = 200$, with those deduced via MC for $ns = 50000$. In all cases, it is evidenced that the Pdf calculated with PDEM is nonzero in a wider range than that obtained with MC. Nevertheless,

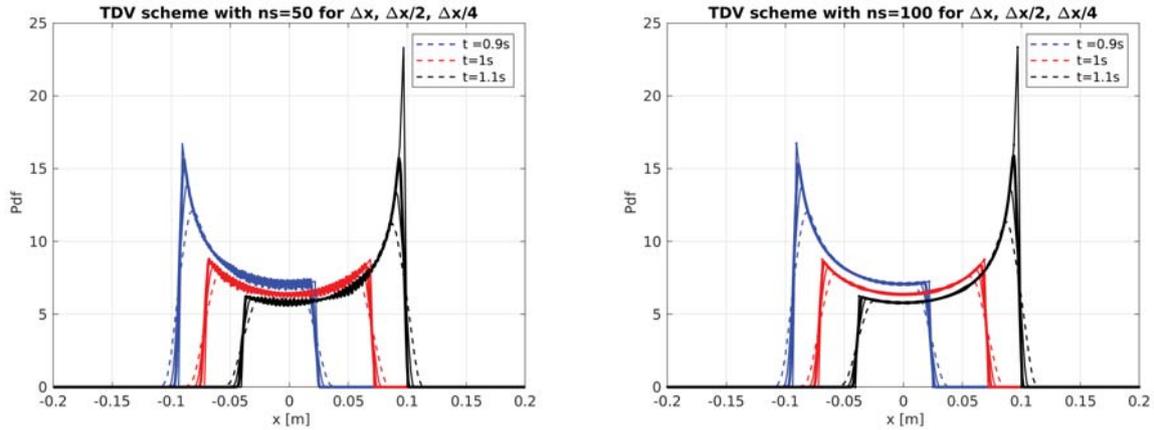


Figure 5: Pdf obtained via the TDV scheme by varying the number of samples n_s and Δx .

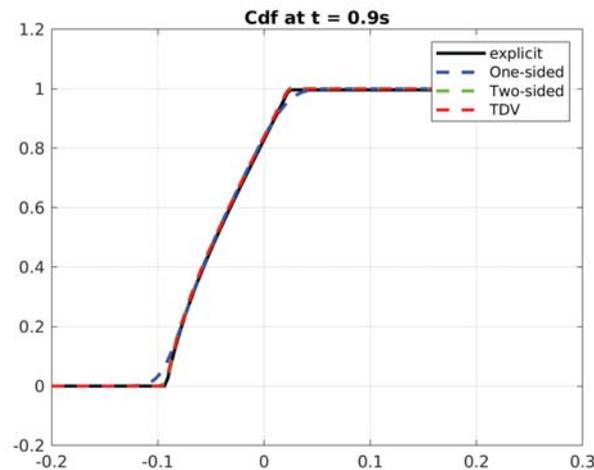


Figure 6: Cdf obtained via the different schemes compared to the explicit one.

the comparison between the Cdf shown in Figure 11, evidences a good agreement between the results obtained with the two different methods.

6 CONCLUSIONS

In the paper, the probabilistic density evolution method is proposed for analysing the seismic response of historic masonry structures. Developed at the early 2000s, the method is applied here for the first time to the study of masonry towers.

Some comparisons with the Monte Carlo method shows the accuracy of the results obtained by the PDEM, that is much less consuming than the former.

Given the high uncertainties involved in the analyses of complex masonry structures, their high non-linearities, the large size and articulated geometries, the method appears particularly suitable for the assessment of the dynamic response of such structures, given the low number of deterministic results needed for a good prevision of the uncertainties propagation.

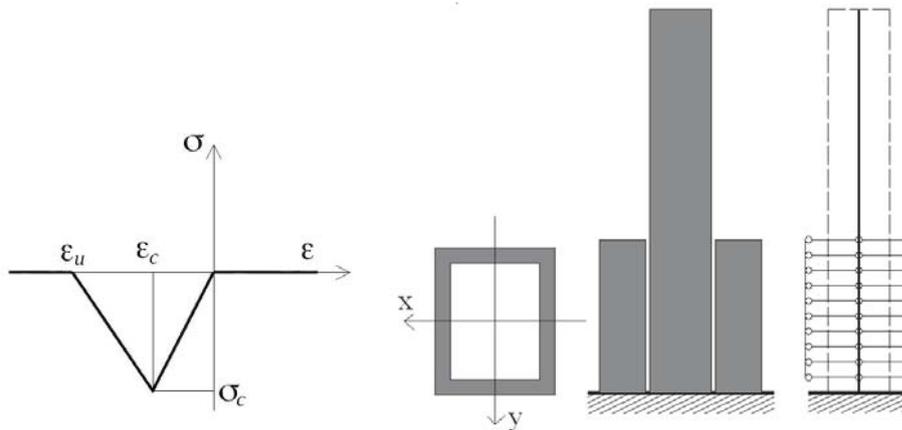


Figure 7: (from the left) Masonry behaviour, simplified geometrical scheme of the tower, and FE mesh.

REFERENCES

- [1] E. Zio, *The Monte Carlo simulation method for system reliability and risk analysis*, Springer-Verlag, 2013.
- [2] J. Li, J. Chen, *Stochastic dynamics of structures*, J. Wiley and Sons, 2009.
- [3] J. Li, J. B. Chen, Probability density evolution method for dynamic response analysis of structures with uncertain parameters, *Computational Mechanics*, **34**, 400–409, 2004.
- [4] M. Lucchesi, B. Pintucchi, N. Zani, MADY, a computer code for numerical modelling masonry structures. *in preparation*, 2020.
- [5] M. Lucchesi, B. Pintucchi, A numerical model for non-linear dynamics analysis of masonry slender structures. *European Journal of Mechanics A/Solids*, **26**, 88–105, 2007.
- [6] M. Lucchesi, B. Pintucchi, M. Šilhavý, N. Zani, On the dynamics of viscous masonry beams. *Continuum Mechanics and Thermodynamic*, **27**, 349–365, 2015.
- [7] M. Lucchesi, B. Pintucchi, N. Zani, Dynamic analysis of FRP-reinforced masonry arches via a no-tension model with damage. A. Di Tommaso, C. Gentilini, and G. Castellazzi eds. *Key Engineering Materials*, **624 KEM**, 619–626, 2015.
- [8] B. Pintucchi, N. Zani, A simple model for performing nonlinear static and dynamic analyses of unreinforced and FRP-strengthened masonry arches. *European Journal of Mechanics /A Solids*, **59**, 210–231, 2016.
- [9] B. Pintucchi, Dynamic analysis of FRP-strengthened masonry. B. Ghiassi, G. Milani eds. *Numerical Modeling of Masonry and Historical Structures: From Theory to Application*, 659–681, 2019.
- [10] G. Bartoli, M. Betti, S. Giordano, In situ static and dynamic investigations on the Torre Grossa masonry tower *Engineering Structures*, **52**, 718–733, 2013.
- [11] M. Lucchesi, B. Pintucchi, N. Zani, The generalized density evolution equation for the dynamic analysis of slender masonry structures. A. Di Tommaso, C. Gentilini, and G. Castellazzi eds. *Key Engineering Materials*, **817 KEM**, 350–355, 2019.

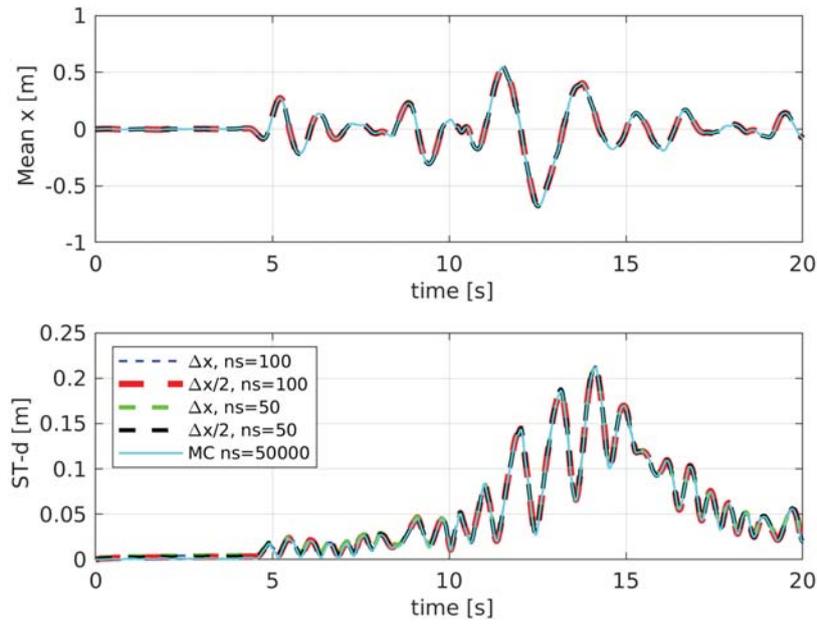


Figure 8: Mean and standard deviation obtained with the PDEM compared to the MC results.

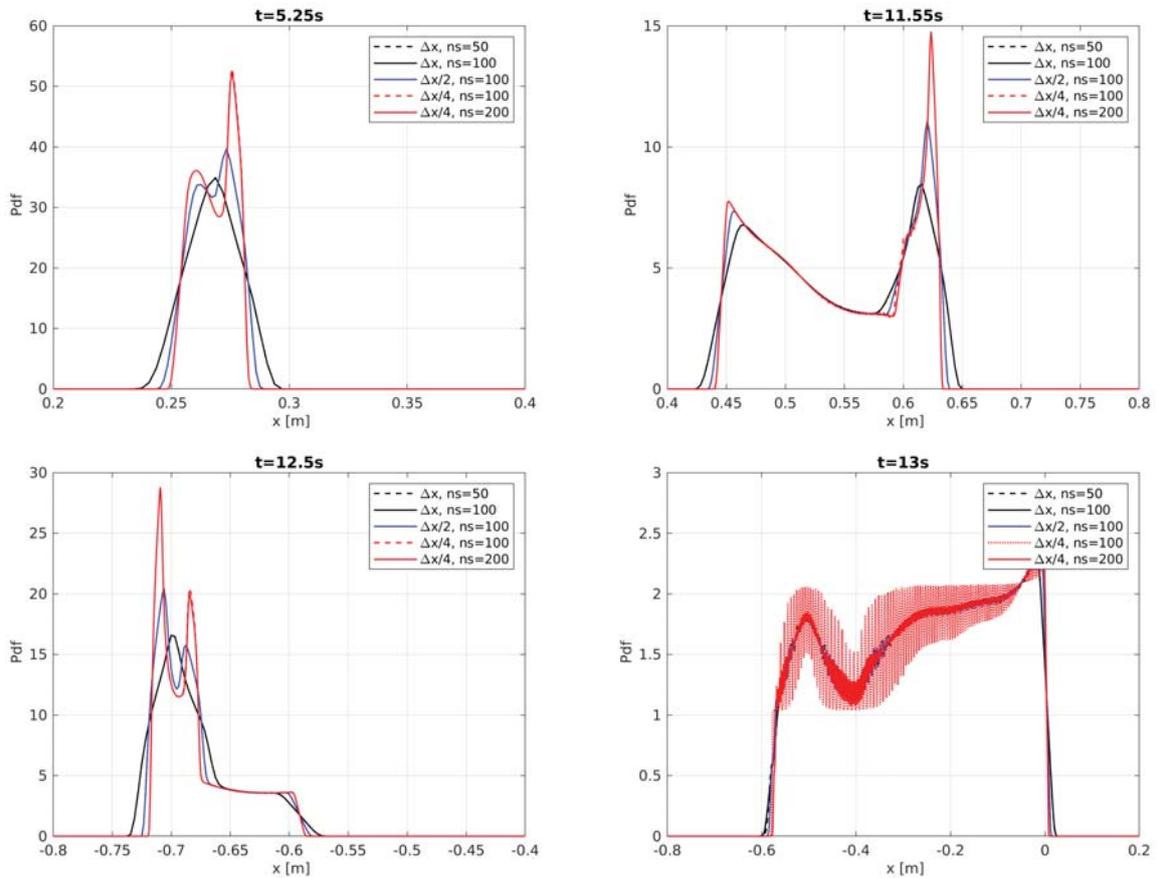


Figure 9: Pdf obtained by varying Δx and the number of samples (ns).

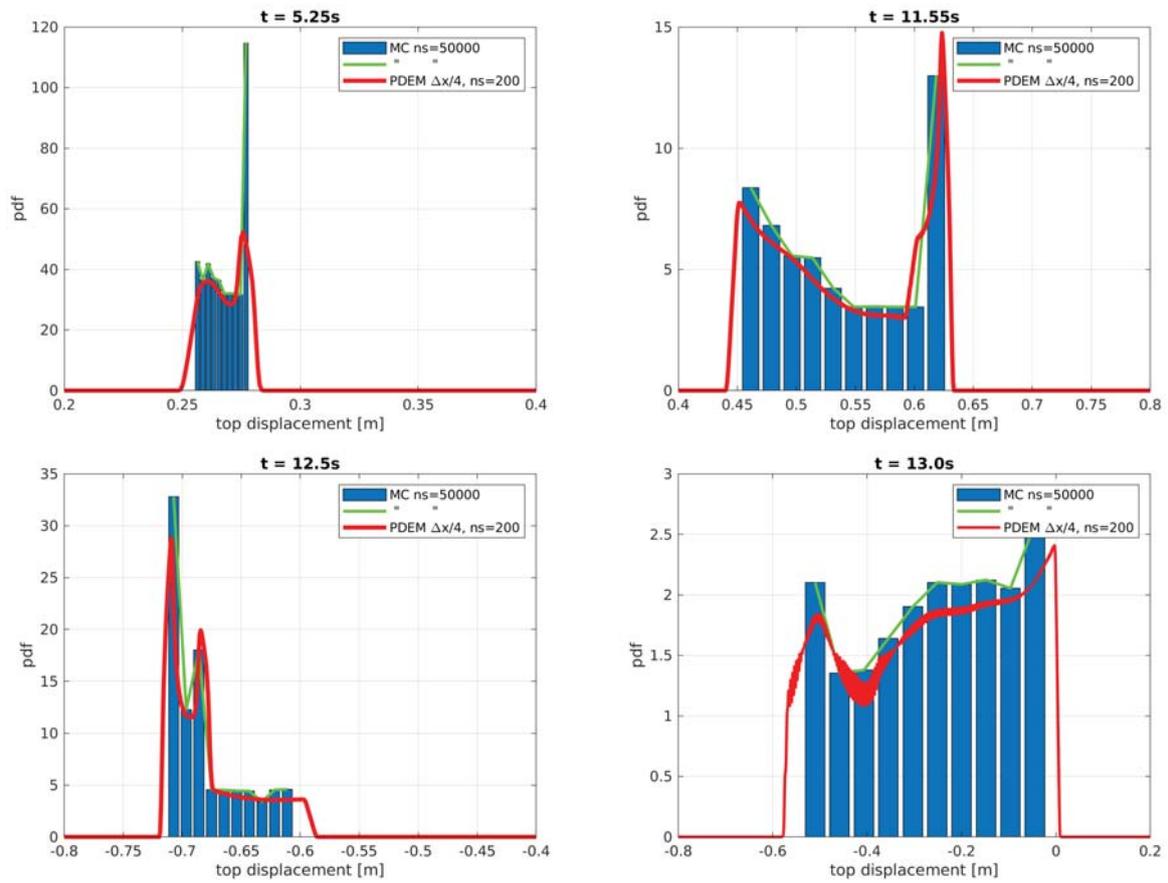


Figure 10: Pdf obtained via the PDEM compared with the MC results.

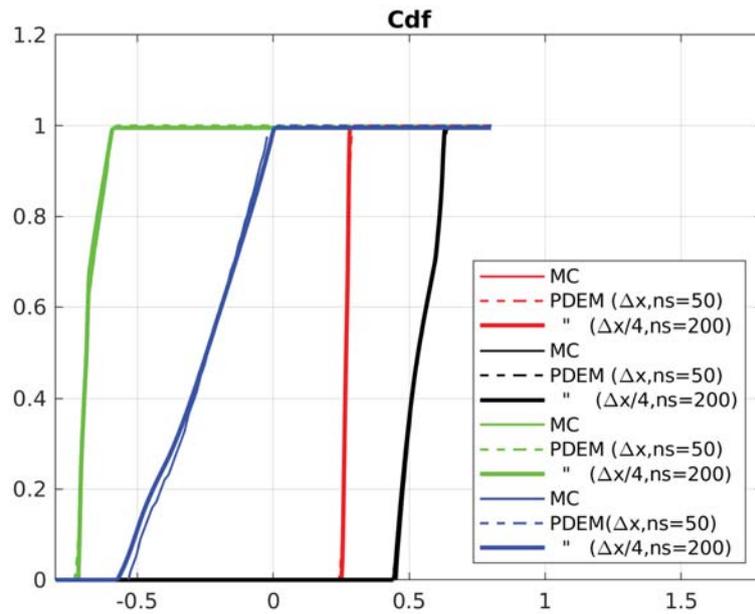


Figure 11: Cdf obtained via the PDEM compared with those from MC results (red lines for $t = 5.25s$, black for $t = 11.55s$, green for $t = 12.5s$ and blue for $t = 13s$).

FEM SHAKEDOWN ANALYSIS OF KIRCHHOFF-LOVE PLATES UNDER UNCERTAINTY OF STRENGTH

Ngọc Trình Trần¹ and Manfred Staat²

¹ Hanoi Architectural University
Hanoi, Vietnam
e-mail: trindhkt@gmail.com

² FH Aachen University of Applied Sciences, Germany
Institute of Bioengineering
Heinrich-Mußmann-Str. 1, 52428 Jülich, Germany
e-mail: m.staat@fh-aachen.de

Abstract

A new formulation to calculate the shakedown limit load of Kirchhoff plates under stochastic conditions of strength is developed. Direct structural reliability design by chance constrained programming is based on the prescribed failure probabilities, which is an effective approach of stochastic programming if it can be formulated as an equivalent deterministic optimization problem.

We restrict uncertainty to strength, the loading is still deterministic. A new formulation is derived in case of random strength with lognormal distribution. Upper bound and lower bound shakedown load factors are calculated simultaneously by a dual algorithm.

Keywords: Kirchhoff Plate, Limit Analysis, Shakedown Analysis, Primal Dual Programming, Stochastic Programming, Chance Constrained Programming.

1 INTRODUCTION

Plates are very important structural elements, which are widely used in civil and mechanical engineering. The common examples of plates are slabs in civil engineering structures, bearing plate under columns, many parts of mechanical components. In this chapter, we consider bending of such plates subjected to lateral loads. The bending stiffness of a plate depends on the cube of its thickness. The classical theory divides plates into following groups: thin plates with small deflection, thin plates with large deflections, and thick plates.

The following assumptions are made in the small deflections theory of thin plates:

- a) *There is no deformation in the middle plane of the plate. This plane remains neutral during bending.*
- b) *The normal to the middle plane of the plate remains straight and normal to the deformed middle plane.*
- c) *The normal stresses in the transverse direction to the plate are negligible.*

The above assumptions on which A.E.H. Love based his plate theory were proposed by Gustav R. Kirchhoff [1]. Consequently, thin plates with small deflections theory are called Kirchhoff-Love plate or Kirchhoff plate for short. This theory is suitable for plates with length of span at least 10 times the thickness. Many engineering problems lie in the above category and satisfactory results are obtained by the classical thin plates theory.

If the span is less than 10 times the thickness, the thin plates assumptions (a) and (b) no longer apply. The Reissner-Mindlin plate thick plates theory, which accounts for shear deformations, or a three dimensional analysis can be recommended for such plates [12].

Limit analysis of plates in bending has been studied analytically and numerically [11]–[23]. Due to limitations of analytical methods, alternative numerical approaches such as finite element methods (FEM), meshfree methods or isogeometric analysis (IGA) have been developed.

In [24] a dual algorithm has been developed to calculate simultaneously both the upper and lower bounds of the plastic collapse limit and shakedown limit of thin plates. We reformulate a similar algorithm as deterministic equivalent of a chance constrained program in which the lower bound and upper bound limit and shakedown load of plate under uncertain strength is computed.

Limit and shakedown analysis state problems as a mathematical programming. If the strength of a plate is a random variable, we may consider the problems as a stochastic programming problem. Many models of stochastic programming have been proposed such as approximate polyhedral dynamic programming [25]–[27], nominal solutions [28], measurement-based optimization [29], [30], worst-case and distributional robustness analysis [31]–[33], robust optimization [34], recourse programming [35]–[37] and chance constrained optimization (CCOPT) [38], [39]. In this paper the CCOPT approach is used to treat the problem of shakedown analysis of plate under uncertainty condition of strength. If the thickness deterministic and the yield stress is distributed normally or lognormal a deterministic equivalent formulation can be derived, which allows a most effective numerical calculation of limit and shakedown loads for a prescribed failure probability of the structure.

2 BASIC RELATIONS IN THIN PLATE THEORY

In this section the necessary relations are listed using the notations as indicated in the plate element shown in Fig. 1.

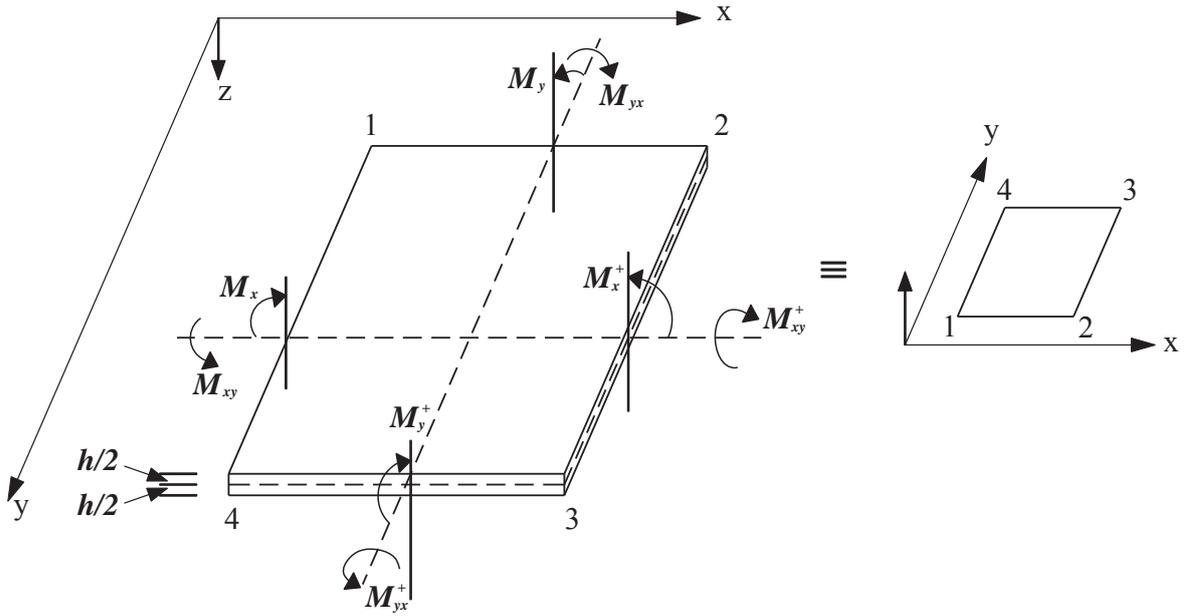


Figure 1: Plate element with internal moment resultants

Let u, v and w be the displacement at any point (x, y, z) in the plate. Similar to elasticity theory, the inelastic behavior of thin plates is analyzed under Kirchhoff's assumption that the normal to the middle plane of the plate remains straight and normal to the deformed middle plane. This assumption yields $u = -z \frac{\partial w}{\partial x}$, $v = -z \frac{\partial w}{\partial y}$ and the strains are obtained as

$$\begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial y} \\ \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \end{Bmatrix} = \begin{Bmatrix} -z \frac{\partial^2 w}{\partial x^2} \\ -z \frac{\partial^2 w}{\partial y^2} \\ -2z \frac{\partial^2 w}{\partial x \partial y} \end{Bmatrix} = z \begin{Bmatrix} \chi_x \\ \chi_y \\ \chi_{xy} \end{Bmatrix}. \quad (2.1)$$

Here $\{\chi\} = [\chi_x \ \chi_y \ \chi_{xy}]^T$ is the vector of curvatures. The kinematic relations can be written as follows:

$$\dot{\chi} = -\nabla^2 \dot{w}, \quad (2.2)$$

where $\dot{\chi}$ is the curvature rate vector and \dot{w} is the transversal velocity.

2.1 Yield criteria

Similar to fully plastic beams, the limit state of greatest load carrying capacity is obtained for a double rectangular distribution of the stresses across the thickness h of the plate. Therefore, the limit values of the bending moments in the x and y directions and of the twisting moment are

$$M_x = \sigma_x \frac{h^2}{4}, \quad M_y = \sigma_y \frac{h^2}{4}, \quad M_{xy} = \sigma_{xy} \frac{h^2}{4}. \quad (2.3)$$

Contrary to the bending of beams, however, the normal stresses σ_x and σ_y are not equal to the yield limits in uniaxial tension; rather, they must satisfy a yield condition for plane stress, taking into account the in-plane shear stress σ_{xy} . The out-of-plane shear stresses, σ_{xz}, σ_{yz} are usually neglected. Consider a general yield condition of the form

$$f(\sigma_x, \sigma_y, \sigma_{xy}) = 0 \quad (2.4)$$

applicable to plane stress states. In a fully plasticized cross section, the stresses $\sigma_x, \sigma_y, \sigma_{xy}$ are constant. Expressing (2.4) in terms of bending and twisting moments, we have the corresponding yield criterion for a plate,

$$f\left(\frac{4M_x}{h^2}, \frac{4M_y}{h^2}, \frac{4M_{xy}}{h^2}\right) = 0. \quad (2.5)$$

As an illustration, the *von Mises yield criterion* can be written in the form

$$f(\sigma_x, \sigma_y, \sigma_{xy}) \equiv \sigma_x^2 - \sigma_x \sigma_y + \sigma_y^2 + 3\sigma_{xy}^2 - \sigma_0^2 = 0. \quad (2.6)$$

Expressing from (2.6) stresses in terms of moments, the criterion takes the form

$$f(M_x, M_y, M_{xy}) \equiv M_x^2 - M_x M_y + M_y^2 + 3M_{xy}^2 - M_0^2 = 0 \quad (2.7)$$

in which

$$M_0 = \sigma_0 \frac{h^2}{4}. \quad (2.8)$$

In matrix form, the von Mises yield criterion can be written as follows:

$$f(\mathbf{m}) = \sqrt{\mathbf{m}^T \mathbf{P} \mathbf{m}} - m_0 = 0, \quad (2.9)$$

where $\mathbf{m} = \{M_x, M_y, M_{xy}\}^T$ is the vector of bending and twisting moments, $m_0 \equiv M_0$ is the fully plastic limit moment per unit length of a plate section and σ_0 is the uniaxial yield stress of material,

$$\mathbf{P} = \frac{1}{2} \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & 0 \\ 0 & 0 & 6 \end{bmatrix}. \quad (2.10)$$

A more complex yield surface of plates and shells has been considered in [40].

3 STATIC APPROACH WITH CHANCE CONSTRAINED PROGRAMMING

Consider a convex polyhedral load domain \mathcal{L} and a special loading path consisting of all load vertices $\hat{P}_k (k=1, \dots, m)$ of \mathcal{L} . The total moment $\mathbf{m}(\mathbf{x}, t)$ at a point $\mathbf{x} \in \Omega$ of the considered plate \mathcal{P} at time t is decomposed into an elastic reference moment $\mathbf{m}^E(\mathbf{x}, t)$ and a residual moment $\boldsymbol{\rho}(\mathbf{x}, t)$. Here, $\mathbf{m}^E(\mathbf{x}, t)$ denotes the fictitious moment that would appear in a purely elastic reference structure \mathcal{P}^E under the same loading conditions as the original struc-

ture, and $\boldsymbol{\rho}(\mathbf{x}, t)$ represents a residual moment field that is induced by the evolution of plastic strains

$$\mathbf{m}(\mathbf{x}, t) = \mathbf{m}^E(\mathbf{x}, t) + \boldsymbol{\rho}(\mathbf{x}, t). \quad (3.1)$$

According to Melan's static shakedown theorem the structure will shakedown, if there exists a time-independent residual moment field $\bar{\boldsymbol{\rho}}(\mathbf{x})$ such that the yield condition is satisfied for any loading path at any time t and in any point \mathbf{x} of the plate. Based on this lower bound theorem, for a plate made up of elastic perfectly plastic material, the maximum enlarging of the load domain allowing still for shakedown, characterized by load factor α^- that can be obtained by solving the following optimization problem:

$$\begin{aligned} \alpha^- = \max \alpha \\ \text{s.t.} : \begin{cases} \nabla^2 \bar{\boldsymbol{\rho}}(\mathbf{x}) = 0 & \text{in } \Omega \\ f[\alpha \mathbf{m}^E(\mathbf{x}, t) + \bar{\boldsymbol{\rho}}(\mathbf{x})] \leq m_0 \end{cases} \end{aligned} \quad (3.2)$$

By discretizing the entire problem domain Ω into finite elements and applying the Gauss-Legendre integration technique, eqs. (3.2) can be rewritten in the following form:

$$\begin{aligned} \alpha^- = \max \alpha \\ \text{s.t.} : \begin{cases} \sum_{i=1}^{NG} w_i \mathbf{B}_i^T \bar{\boldsymbol{\rho}}_i = 0 & \text{in } \Omega \\ f(\alpha \mathbf{m}_{ik}^E + \bar{\boldsymbol{\rho}}_i) \leq m_0 & \forall i = \overline{1, NG} \quad \forall k = \overline{1, m} \end{cases} \end{aligned} \quad (3.3)$$

in which \mathbf{B}_i is the deformation matrix, w_i is integration weight at Gauss point i and NG denotes the total number of Gauss points of the structure.

Let us now consider the situation that the plastic moment of the plate is not given but must be modelled $m_0 = m_0(\omega)$ a random variable on a certain probability space. Under uncertainty, the inequalities in (3.3) are not always satisfied, the probability of the i^{th} yield condition is required to be satisfied is greater than some reliability level ψ_i . Problem (3.3) becomes a chance constraint stochastic program:

$$\begin{aligned} \alpha^- = \max \alpha \\ \text{s.t.} : \begin{cases} \sum_{i=1}^{NG} w_i \mathbf{B}_i^T \bar{\boldsymbol{\rho}}_i = \mathbf{0} & \text{in } \Omega \\ \text{Prob}[f(\alpha \mathbf{m}_{ik}^E + \bar{\boldsymbol{\rho}}_i) - m_{0i}(\omega) \leq 0] \geq \psi_i & \forall i = \overline{1, NG} \quad \forall k = \overline{1, m} \end{cases} \end{aligned} \quad (3.4)$$

Let the plastic moment $m_i(\omega)$ be distributed normally with mean μ_i and standard deviation σ_i , in short $m_i(\omega) \sim \mathcal{N}(\mu_i, \sigma_i^2)$. Based on the methodology of chance constrained programming, problem (3.4) can be converted into a equivalent deterministic program as shown in [10], [11]:

$$\alpha^- = \max \alpha$$

$$\text{s.t.:} \begin{cases} \sum_{i=1}^{NG} w_i \mathbf{B}_i^T \bar{\mathbf{p}}_i = 0 & \text{in } \Omega \\ f(\alpha \mathbf{m}_{ik}^E + \bar{\mathbf{p}}_i) \leq \mu_i - \kappa \sigma_i & \forall i = \overline{1, NG} \quad \forall k = \overline{1, m} \end{cases} \quad (3.5)$$

where $\kappa = \Phi^{-1}(\psi_i)$ is the inverse normal cumulative distribution function (normal quantile function) of the plastic moment at Gauss point i .

Let the plastic moment $m_i(\omega)$ be distributed lognormally. This means that $\ln[m_i(\omega)]$ is distributed normally with mean μ_i and standard deviation σ_i , in short $\ln[m_i(\omega)] \sim \mathcal{N}(\mu_i, \sigma_i^2)$. The stochastic program (3.4) can be relaxed into an equivalent deterministic optimization problem after some transformations [2,3]:

$$\alpha^- = \max \alpha$$

$$\text{s.t.:} \begin{cases} \sum_{i=1}^{NG} w_i \mathbf{B}_i^T \bar{\mathbf{p}}_i = 0 & \text{in } \Omega \\ f(\alpha \mathbf{m}_{ik}^E + \bar{\mathbf{p}}_i) \leq e^{\mu_i - \kappa \sigma_i} & \forall i = \overline{1, NG} \quad \forall k = \overline{1, m} \end{cases} \quad (3.6)$$

4 KINEMATIC APPROACH WITH CHANCE CONSTRAINED PROGRAMMING

An upper bound to the shakedown limit of plates can be obtained using the kinematic shakedown theorem which has following two statements:

Upper bound: Shakedown will occur for a structure subject to repeated or cyclic loads, if the rate of plastic dissipation power exceeds the work rate of external forces for any admissible plastic strain-rate cycles and all loading paths.

Lower bound: Shakedown cannot occur, if the rate of plastic dissipation power is less than the work rate of external forces for any one admissible plastic strain-rate cycle or any one loading path.

In this investigation, we use von Mises yield criterion. The power of plastic dissipation per unit area of the plate can be formulated as a function of strain rate:

$$\dot{D}_p = \sigma_0 \sqrt{\dot{\boldsymbol{\varepsilon}}^T \mathbf{Q} \dot{\boldsymbol{\varepsilon}}} \quad (4.1)$$

where (with (2.10))

$$\mathbf{Q} = \mathbf{P}^{-1} = \frac{1}{3} \begin{bmatrix} 4 & 2 & 0 \\ 2 & 4 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.2)$$

The plastic dissipation power of the plate domain Ω can be written

$$\dot{D}_{\text{int}}(\dot{\boldsymbol{\chi}}) = \int_{\Omega-h/2}^{h/2} \int_{\Omega} \dot{D}_p \, dz d\Omega = m_0 \int_{\Omega} \sqrt{\dot{\boldsymbol{\chi}}^T \mathbf{Q} \dot{\boldsymbol{\chi}}} \, d\Omega \quad (4.3)$$

in which m_0 is the plastic limit moment per unit length of a plate section is computed as (2.8)

We introduce here an admissible cycle of a plastic curvature field $\Delta\boldsymbol{\chi}^p$. At each load vertex, the plastic curvature rate may not necessarily be compatible at each instant during the time cycle, but the plastic curvature accumulation over the cycle is required to be kinematically compatible such that

$$\Delta\boldsymbol{\chi}^p = \sum_{k=1}^m \dot{\boldsymbol{\chi}}^p = \nabla^2 \dot{w} \quad (4.4)$$

Based on the above statements and the mathematical programming theory, an upper bound of the shakedown load factor can be found by solving the following convex nonlinear programming (the superscript p is neglected for simplicity):

$$\begin{aligned} \alpha^+ = \min & \sum_{k=1}^m \int_A \dot{D}_{\text{int}}(\dot{\boldsymbol{\chi}}) d\Omega \\ \text{s.t.} & \begin{cases} \Delta\boldsymbol{\chi}^p = \sum_{k=1}^m \dot{\boldsymbol{\chi}} = \nabla^2 \dot{w} & \text{in } \Omega \\ \dot{w} = 0 & \text{on } \partial\Omega \\ \sum_{k=1}^m \int_{\Omega} \mathbf{m}^E(x, \hat{P}_k) \dot{\boldsymbol{\chi}}^T d\Omega = 1 \end{cases} \end{aligned} \quad (4.5)$$

We denote the nodal variables of the finite element by $\mathbf{u} = [w \ \partial w / \partial x \ \partial w / \partial y]^T$. The discretized formulation by FEM is as follows:

$$\begin{aligned} \alpha^+ = \min & \sum_{k=1}^m \sum_{i=1}^{NG} w_i m_0 \sqrt{\dot{\boldsymbol{\chi}}_{ik}^T \mathbf{Q} \dot{\boldsymbol{\chi}}_{ik}} \\ \text{s.t.} & \begin{cases} \sum_{k=1}^m \dot{\boldsymbol{\chi}}_{ik} = \mathbf{B}_i \mathbf{u} & \forall i = \overline{1, NG} \\ \sum_{k=1}^m \sum_{i=1}^{NG} w_i \dot{\boldsymbol{\chi}}_{ik}^T \mathbf{m}_{ik}^E = 1 \end{cases} \end{aligned} \quad (4.6)$$

If the yield stress of the material is random, then the plastic moment is an uncertain quantity and the objective function of (4.6) is a stochastic variable. Firstly, we must properly define the minimum of a random function. This can be done in such a way that one looks for a minimum lower bound η of the objective function under the constraint that the probability of violation of that bound is prescribed in [39]

$$\begin{aligned} \min & \eta \\ \text{s.t.} & \begin{cases} \text{Prob} \left(\sum_{k=1}^m \sum_{i=1}^{NG} w_i m_0(\omega) \sqrt{\dot{\boldsymbol{\chi}}_{ik}^T \mathbf{Q} \dot{\boldsymbol{\chi}}_{ik}} \geq \eta \right) = \psi \\ \sum_{k=1}^m \dot{\boldsymbol{\chi}}_{ik} = \mathbf{B}_i \mathbf{u} & \forall i = \overline{1, NG} \\ \sum_{k=1}^m \sum_{i=1}^{NG} w_i \dot{\boldsymbol{\chi}}_{ik}^T \mathbf{m}_{ik}^E = 1 \end{cases} \end{aligned} \quad (4.7)$$

Problem (4.7) is a stochastic program, which can be converted into an equivalent deterministic program by using a chance constrained programming technique [10], [11].

$$\alpha^+ = \min \sum_{k=1}^m \sum_{i=1}^{NG} w_i (\mu_i - \kappa \sigma_i) \sqrt{\dot{\boldsymbol{\chi}}_{ik}^T \mathbf{Q} \dot{\boldsymbol{\chi}}_{ik}}$$

$$\text{s.t.} : \begin{cases} \sum_{k=1}^m \dot{\boldsymbol{\chi}}_{ik} = \mathbf{B}_i \dot{\mathbf{u}} & \forall i = \overline{1, NG} \\ \sum_{k=1}^m \sum_{i=1}^{NG} w_i \dot{\boldsymbol{\chi}}_{ik}^T \mathbf{m}_{ik}^E = 1 \end{cases} \quad (4.8)$$

In case of a lognormal distribution of strength, the stochastic problem (4.7) can be converted into the equivalent deterministic program (4.9) by using the duality property [3]:

$$\alpha^+ = \min \sum_{k=1}^m \sum_{i=1}^{NG} w_i e^{(\mu_i - \kappa \sigma_i)} \sqrt{\dot{\boldsymbol{\chi}}_{ik}^T \mathbf{Q} \dot{\boldsymbol{\chi}}_{ik}}$$

$$\text{s.t.} : \begin{cases} \sum_{k=1}^m \dot{\boldsymbol{\chi}}_{ik} = \mathbf{B}_i \dot{\mathbf{u}} & \forall i = \overline{1, NG} \\ \sum_{k=1}^m \sum_{i=1}^{NG} w_i \dot{\boldsymbol{\chi}}_{ik}^T \mathbf{m}_{ik}^E = 1 \end{cases} \quad (4.9)$$

5 A DUAL ALGORITHM FOR SHAKEDOWN ANALYSIS OF A KIRCHHOFF PLATE

For the sake of simplicity, we set some new notations:

$$\dot{\mathbf{k}}_{ik} = w_i \mathbf{Q}^{1/2} \dot{\boldsymbol{\chi}}_{ik}, \quad \mathbf{t}_{ik} = (\mathbf{Q}^{-1/2})^T \mathbf{m}_{ik}^E, \quad \hat{\mathbf{B}}_i = w_i \mathbf{Q}^{1/2} \mathbf{B}_i, \quad (5.1)$$

where

$$\mathbf{Q}^{1/2} \mathbf{Q}^{-1/2} = \mathbf{I}, \quad \mathbf{Q} = (\mathbf{Q}^{1/2})^T \mathbf{Q}^{1/2}. \quad (5.2)$$

By substituting (5.1) into (4.10) one obtains a simplified version for the upper bound of the shakedown limit load (primal problem)

$$\alpha^+ = \min \sum_{k=1}^m \sum_{i=1}^{NG} e^{(\mu_i - \kappa \sigma_i)} \sqrt{\dot{\mathbf{k}}_{ik}^T \dot{\mathbf{k}}_{ik}}$$

$$\text{s.t.} : \begin{cases} \sum_{k=1}^m \dot{\mathbf{k}}_{ik} - \hat{\mathbf{B}}_i \dot{\mathbf{u}} = \mathbf{0} & \forall i = \overline{1, NG} \\ \sum_{i=1}^{NG} \sum_{k=1}^m \dot{\mathbf{k}}_{ik}^T \mathbf{t}_{ik} - 1 = 0 \end{cases} \quad (5.3)$$

In order to allow a direct nonlinear of the nonsmooth optimization problem, a ‘smooth regularization method’ can be used for overcoming this technical problem. For this purpose, a very small positive number ε_0^2 is added to $D_{\text{int}}(\dot{\mathbf{k}}_{ik})$. An efficient technique for large-scale optimization problems, which are successfully applied in [9] is used. Using a penalty method to eliminate the first constraint in (5.3) leads to the penalty function

$$F_p = \sum_{i=1}^{NG} \left\{ \sum_{k=1}^m e^{(\mu_i - \kappa \sigma_i)} \sqrt{\dot{\mathbf{k}}_{ik}^T \dot{\mathbf{k}}_{ik} + \varepsilon_0^2} + \frac{c}{2} \left(\sum_{k=1}^m \dot{\mathbf{k}}_{ik} - \hat{\mathbf{B}}_i \dot{\mathbf{u}} \right)^T \left(\sum_{k=1}^m \dot{\mathbf{k}}_{ik} - \hat{\mathbf{B}}_i \dot{\mathbf{u}} \right) \right\}, \quad (5.4)$$

where c is a penalty parameter such that $c \gg 1$. The corresponding Lagrange function of (5.4) is

$$L = F_p - \alpha \left(\sum_{i=1}^{NG} \sum_{k=1}^m \dot{\mathbf{k}}_{ik}^T \mathbf{t}_{ik} - 1 \right). \quad (5.5)$$

We denote

$$\boldsymbol{\beta}_i = -c \left(\sum_{k=1}^m \dot{\mathbf{k}}_{ik} - \hat{\mathbf{B}}_i \dot{\mathbf{u}} \right). \quad (5.5)$$

By employing Newton method to solve the Karush-Kuhn-Tucker (KKT) conditions of the Lagrange function (4.17) and after some manipulations, one gets the following system:

$$\mathbf{K} d\dot{\mathbf{u}} = -\mathbf{K}\dot{\mathbf{u}} + \mathbf{f}_1 + \mathbf{f}_2(\alpha + d\alpha), \quad (5.6)$$

in which

$$\begin{aligned} \mathbf{K} &= \sum_{i=1}^{NG} \hat{\mathbf{B}}_i^T \mathbf{E}_i^{-1} \hat{\mathbf{B}}_i \\ \mathbf{f}_1 &= -\sum_{i=1}^{NG} \hat{\mathbf{B}}_i^T \mathbf{E}_i^{-1} \sum_{k=1}^m \mathbf{M}_{ik}^{-1} (\boldsymbol{\beta}_i + \alpha \mathbf{t}_{ik}) \frac{\dot{\mathbf{k}}_{ik}^T \dot{\mathbf{k}}_{ik}}{\sqrt{\dot{\mathbf{k}}_{ik}^T \dot{\mathbf{k}}_{ik} + \varepsilon_0^2}} \\ \mathbf{f}_2 &= \sum_{i=1}^{NG} \hat{\mathbf{B}}_i^T \mathbf{E}_i^{-1} \sum_k^m \mathbf{M}_{ik}^{-1} \sqrt{\dot{\mathbf{k}}_{ik}^T \dot{\mathbf{k}}_{ik} + \varepsilon_0^2} \mathbf{t}_{ik} \end{aligned} \quad (5.7)$$

and

$$\begin{aligned} \mathbf{M}_{ik} &= e^{(\mu_i - \kappa \sigma_i)} \mathbf{I} + (\boldsymbol{\beta}_i + \alpha \mathbf{t}_{ik}) \frac{\dot{\mathbf{k}}_{ik}^T}{\sqrt{\dot{\mathbf{k}}_{ik}^T \dot{\mathbf{k}}_{ik} + \varepsilon_0^2}} \\ \mathbf{E}_i &= \frac{\mathbf{I}}{c} + \sum_k^m \mathbf{M}_{ik}^{-1} \sqrt{\dot{\mathbf{k}}_{ik}^T \dot{\mathbf{k}}_{ik} + \varepsilon_0^2} \end{aligned} \quad (5.8)$$

The system (4.19) with the two last terms on the right-hand side may be interpreted as the linear system arising in purely elastic computations with the global stiffness matrix \mathbf{K} . The matrix \mathbf{E}_i^{-1} plays the role of the elastic matrix. Solving this system by the same procedure as for the purely elastic calculation will ensure the kinematic boundary condition for the displacement rate to be satisfied automatically. We have the incremental vectors of nodal variables $\dot{\mathbf{u}}$, curvature rate $\dot{\mathbf{k}}_{ik}$ and $\boldsymbol{\beta}_i$ as follows :

$$\begin{aligned} d\dot{\mathbf{u}} &= d\dot{\mathbf{u}}_1 + d\dot{\mathbf{u}}_2(\alpha + d\alpha) \\ d\dot{\mathbf{k}}_{ik} &= (d\dot{\mathbf{k}}_{ik})_1 + (d\dot{\mathbf{k}}_{ik})_2(\alpha + d\alpha) \\ d\boldsymbol{\beta}_i &= (d\boldsymbol{\beta}_i)_1 + (d\boldsymbol{\beta}_i)_2(\alpha + d\alpha) \end{aligned} \quad (5.9)$$

where

$$\begin{aligned}
 d\dot{\mathbf{u}}_1 &= -\dot{\mathbf{u}} + \mathbf{K}^{-1}\mathbf{f}_1 \\
 d\dot{\mathbf{u}}_2 &= \mathbf{K}^{-1}\mathbf{f}_2 \\
 (d\dot{\mathbf{k}}_{ik})_1 &= -\mathbf{M}_{ik}^{-1}\sqrt{\dot{\mathbf{k}}_{ik}^T\dot{\mathbf{k}}_{ik} + \varepsilon_0^2}(d\boldsymbol{\beta}_i)_1 - \mathbf{M}_{ik}^{-1}\left(\bar{m}_i\dot{\mathbf{k}}_{ik} + \sqrt{\dot{\mathbf{k}}_{ik}^T\dot{\mathbf{k}}_{ik} + \varepsilon_0^2}\boldsymbol{\beta}_i\right) \\
 (d\dot{\mathbf{k}}_{ik})_2 &= -\mathbf{M}_{ik}^{-1}\sqrt{\dot{\mathbf{k}}_{ik}^T\dot{\mathbf{k}}_{ik} + \varepsilon_0^2}(d\boldsymbol{\beta}_i)_2 - \mathbf{M}_{ik}^{-1}\sqrt{\dot{\mathbf{k}}_{ik}^T\dot{\mathbf{k}}_{ik} + \varepsilon_0^2}\mathbf{t}_{ik} \\
 (d\boldsymbol{\beta}_i)_1 &= -\mathbf{E}_i^{-1}\sum_k^m \mathbf{M}_{ik}^{-1}\bar{m}_i\dot{\mathbf{k}}_{ik} - \mathbf{E}_i^{-1}\left[\hat{\mathbf{B}}_i d\dot{\mathbf{u}}_1 - \left(\sum_{k=1}^m \dot{\mathbf{k}}_{ik} - \hat{\mathbf{B}}_i\dot{\mathbf{u}}\right)\right] - \boldsymbol{\beta}_i \\
 (d\boldsymbol{\beta}_i)_2 &= -\mathbf{E}_i^{-1}\hat{\mathbf{B}}_i d\dot{\mathbf{u}}_2 - \mathbf{E}_i^{-1}\sum_k^m \mathbf{M}_{ik}^{-1}\sqrt{\dot{\mathbf{k}}_{ik}^T\dot{\mathbf{k}}_{ik} + \varepsilon_0^2}\mathbf{t}_{ik}
 \end{aligned} \tag{5.10}$$

and

$$(\alpha + d\alpha) = \frac{1 - \sum_{i=1}^{NG} \sum_{k=1}^m \mathbf{t}_{ik}^T \left[\dot{\mathbf{k}}_{ik} + (d\dot{\mathbf{k}}_{ik})_1 \right]}{\sum_{i=1}^{NG} \sum_{k=1}^m \mathbf{t}_{ik}^T (d\dot{\mathbf{k}}_{ik})_2} \tag{5.11}$$

The vectors $d\dot{\mathbf{q}}, d\dot{\mathbf{k}}_{ik}, d\boldsymbol{\beta}_i$ and $d\alpha$ are actually Newton directions, which assure that a suitable step along them will lead to a decrease of the objective function of the primal problem (5.3) and to an increase of the objective function of the objective function of the dual problem (3.8). Based on (5.9-5.11) we can update the vectors of $\dot{\mathbf{q}}, \dot{\mathbf{k}}_{ik}, \boldsymbol{\beta}_i$ and α . The dual algorithm for limit and shakedown analysis is presented in detail in [3].

6 NUMERICAL EXAMPLES

We investigate a L-shape plate subjected to uniform pressure. Length $L = 10m$, plate thickness $t = 0.1m$, the mean value of yield stress $E(\sigma_0) = 250 \text{ MPa}$ and the standard deviation $\sigma = 0.1E(\sigma_0)$. The reliability level is assumed $\psi = 0.9999$. Let us calculate limit and shakedown load factors.

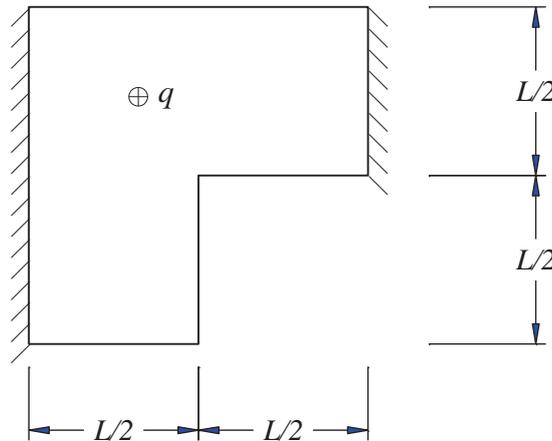


Figure 2: L-shape plate loaded by a uniform pressure

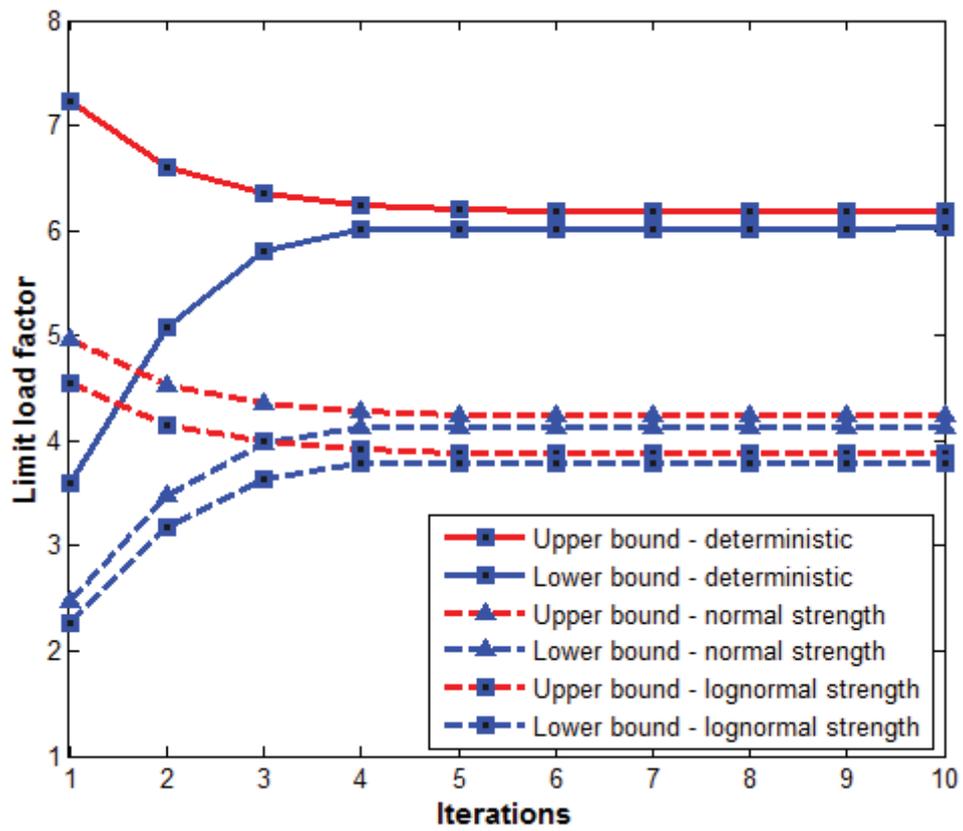


Figure 3: Convergence of limit load factors

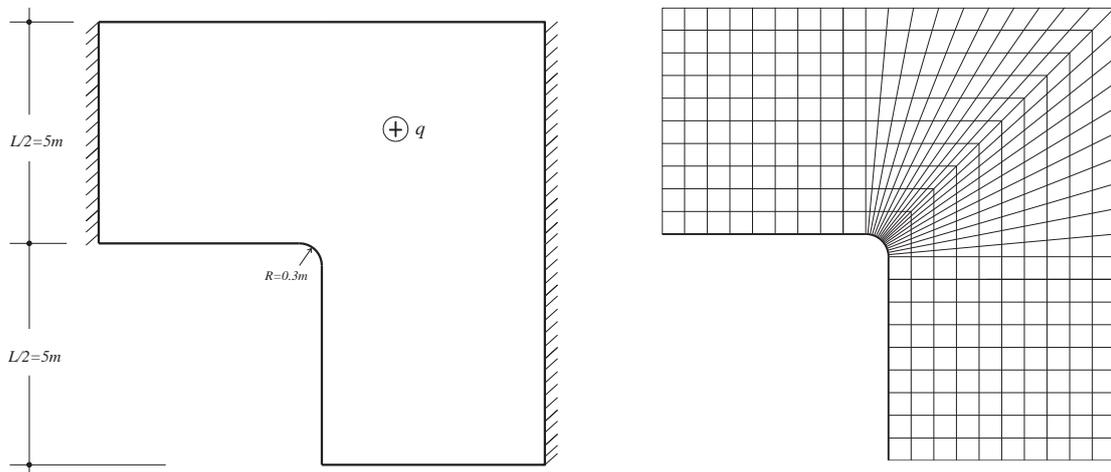


Figure 4. L-shape Plate: rounding at the corner

Authors	Lower bound	Upper bound	
Le <i>et al.</i> [15]	–	6.219	
Tran <i>et al.</i> [24]	6.044	6.173	deterministic
Present	6.022	6.190	
	3.785	3.882	normal
	4.135	4.242	lognormal

Table 1: Limit load factor in comparison for case of simple supported plate

For shakedown analysis, the stress singularity at the sharp reentrant corner has to be removed by rounding the plate at the corner as shown in Figure 4. The FE mesh is made with 380 DKQ elements. Table 2 shows the limit and shakedown load factors if a uniform load varies in the domain $q = [0 \ 1]$. If we compare with the upper and lower bounds in Table 2, the limit load factors are similar. The convergence of shakedown load factors is shown in Figure 5.

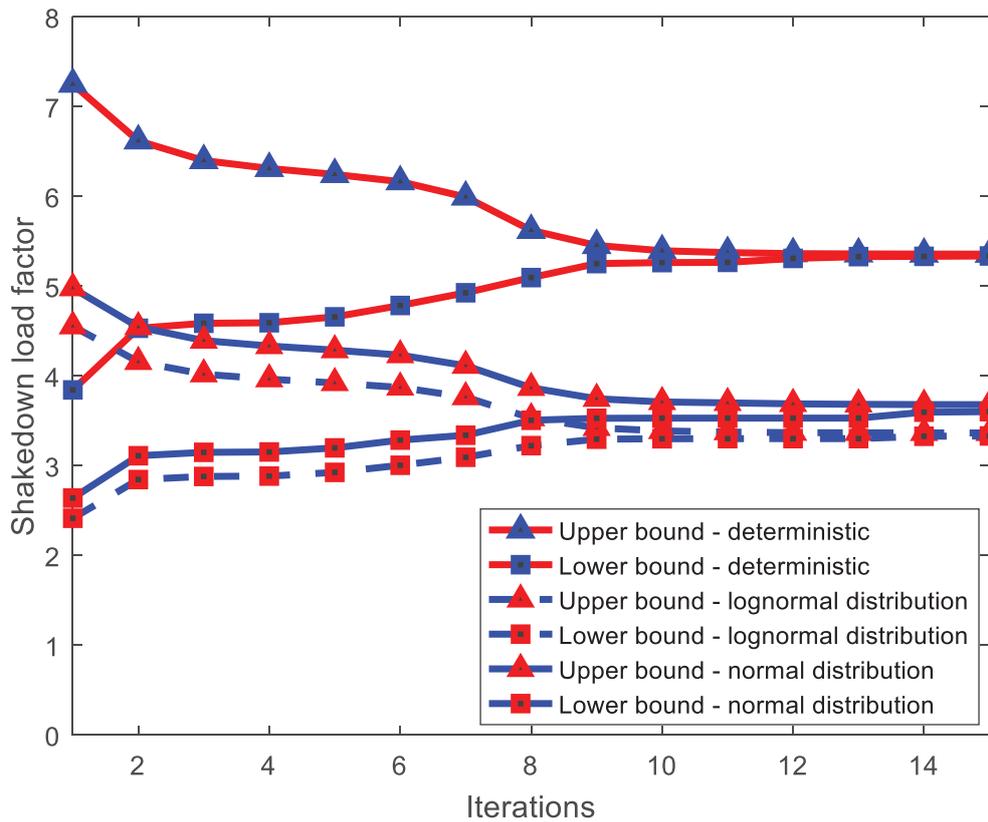


Figure 5: Convergence of Shakedown load factors

	Lower bound	Upper bound	
Limit analysis	5.979	6.224	deterministic
	4.137	4.273	lognormal
	3.805	3.909	normal
Shakedown analysis	5.339	5.355	deterministic
	3.619	3.677	lognormal
	3.363	3.382	normal

Table 2: Limit and shakedown load factors for plate in Figure 4

7 CONCLUSIONS

Reliability analysis of plates and shells calculates the failure probability after structural design for a given loading [13]. We have presented a probabilistic design method for plates, which allows the most effective numerical calculation of limit and shakedown loads for a prescribed failure probability of the structure with stochastic plastic moment. The implementation of the extension to stochastic loading, obtained in [2], is under preparation. The stochastic programming approach can be proposed as a method for structural optimization. Shakedown analysis has the advantage that it yields a design, which is optimum for all possible time-variant loadings in a considered load domain [41], [42].

REFERENCES

- [1] G. Kirchhoff, Über das Gleichgewicht und die Bewegung einer elastischen Scheibe. *J. reine angew. Math.* **40**: 51-88, 1850.
- [2] N.T. Trần, M. Staat, Direct plastic structural design under random strength and random load by chance constrained programming. *Eur J Mech A Solids*, **85**(1), art. no. 104106, 2021.
- [3] N. T. Trần, M. Staat, Direct plastic structural design under lognormally distributed strength by chance constrained programming. *Optim. Eng.* **21**(1), 131-157, 2020.
- [4] Ngọc Trinh Trần, *Limit and Shakedown analysis of structures under stochastic conditions*. PhD thesis, Technische Universität Carolo-Wilhelmina zu Braunschweig, Braunschweig, Germany, 2018.
- [5] N.T. Trần, T.N. Trần, H.G. Matthies, G.E. Stavroulakis, M. Staat, Shakedown analysis under stochastic uncertainty by chance constrained programming. O. Barrera, A. Cocks, A. Ponter eds. *Advances in direct methods for materials and structures*. Springer, Cham, Heidelberg, 85-103, 2018.
- [6] M. Staat, Limit and shakedown analysis under uncertainty. *Int J Comput Methods*, **11**(3), Article ID 1343008, 2014.
- [7] M. Staat, M. Heitzer, Probabilistic limit and shakedown problems. M. Staat, M. Heitzer eds. *Numerical methods for limit and shakedown analysis. Deterministic and probabilistic approach*. Part VII. NIC Series Vol. 15, John von Neumann Institute for Computing, Jülich, 217-268, 2003. <http://hdl.handle.net/2128/2926>

- [8] M.A. Save, C.E. Massonnet, G. de Saxcé, *Plastic limit analysis of plates, shells and disks. 2nd Edition*, North Holland, 1997.
- [9] D. Khoi Vu, *Dual limit and shakedown analysis of structures*. PhD thesis. Collection des publications de la Faculté des Sciences Appliquées, Université de Liège, Belgique, 2001.
- [10] N.T. Trần, T.N. Trần, H.G. Matthies, G.E. Stavroulakis, M. Staat, Shakedown analysis of plate bending under stochastic uncertainty by chance constrained programming. *Proc. VII Eur. Congr. Comput. Methods Appl. Sci. Eng. (ECCOMAS Congr. 2016)*, no. June, pp. 3007–3019, 2016.
- [11] N.T. Trần, T.N. Trần, H.G. Matthies, G.E. Stavroulakis, M. Staat, *Shakedown analysis of plate bending analysis under stochastic uncertainty by chance constrained programming*. M. Papadrakakis, V. Papadopoulos, G. Stefanou, V. Plevris eds. ECCOMAS Congress 2016, VII European Congress on Computational Methods in Applied Sciences and Engineering. Crete Island, Greece, 5–10 June 2016, Vol. 2, pp. 3007-3019, 2016.
- [12] T.N. Trần, M. Staat, Shakedown analysis of Reissner-Mindlin plates using the edge-based smoothed finite element method. K. Spiliopoulos, D. Weichert eds. *Limit states of materials and structures: Direct methods*. Springer, Dordrecht, 101-117, 2014.
- [13] T.N. Trần, R. Kreißig, M. Staat, Probabilistic limit and shakedown analysis of thin plates and shells. *Structural Safety*, **31**(1), 1-18, 2009.
- [14] C.V. Le, H. Nguyen-Xuan, H. Nguyen-Dang, Upper and lower bound limit analysis of plates using FEM and second-order cone programming. *Comput. Struct.*, **88**(1-2), 65-73, 2010.
- [15] C.V. Le, M. Gilbert, H. Askes, Limit analysis of plates using the EFG method and second-order cone programming. *Int. J. Numer. Meth. Engng*, **78**, 1532-1552, 2009
- [16] T. Belytschko, P.G. Hodge, Numerical methods for the limit analysis of plates. *Trans. ASME, J. Appl. Mech.*, **35**, 796-801, 1968.
- [17] C.T. Morley, *The ultimate bending strength of reinforced concrete slabs*. PhD thesis, Cambridge University, 1965.
- [18] L. Capsoni, A. Corradi, Limit analysis of plates-a finite element formulation. *Struct. Eng. Mech.*, **8**(4), 325-341, 1999.
- [19] E.N. Fox, Limit analysis for plates: the exact solution for a clamped square plate of isotropic homogeneous material obeying the square yield criterion and loaded by uniform pressure. *Math. Phys. Eng. Sci.*, **277**(1265), 121-155, 1974.
- [20] R.H. Wood, A partial failure of limit analysis for slabs, and the consequences for future research. *Mag.Concr. Res.*, **21**, 79-90, 1969.
- [21] W.C. McCarthy, L.A. Traina, A plate bending finite element model with a limit analysis capacity. *Math. Model.*, **8**(Supplement C),486-492, 1987.
- [22] K. Krabbenhoft, L. Damkilde, Lower bound limit analysis of slabs with nonlinear yield criteria. *Comput. Struct.*, **80**(27-30), 2043–2057, 2002.
- [23] S. Timoshenko, S. Woinowsky-Krieger, *Theory of plates and shells. 2nd Edition*. McGraw Hill,1959.

- [24] T.N. Tran, A dual algorithm for shakedown analysis of plate bending. *Numer. Methods Eng.*, **86**(7), 862-875, 2011.
- [25] J. Björnberg and M. Diehl, Approximate robust dynamic programming and robustly stable MPC. *Automatica*, **42**(5), 777-782, 2006.
- [26] L. Zéphyr, P. Lang, B. F. Lamond, P. Côté, Approximate stochastic dynamic programming for hydroelectric production planning. *Eur. J. Oper. Res.*, **262**(2), 586-601, 2017.
- [27] K. Fukushima, Y. Waki, A polyhedral approximation approach to concave numerical dynamic programming. *J. Econ. Dyn. Control*, **37**(11), 2322-2335, 2013.
- [28] B. Srinivasan, S. Palanki, D. Bonvin, Dynamic optimization of batch processes: I. Characterization of the nominal solution. *Comput. Chem. Eng.*, **27**(1), 1-26, 2003.
- [29] B. Srinivasan, D. Bonvin, E. Visser, S. Palanki, Dynamic optimization of batch processes: II. Role of measurements in handling uncertainty. *Comput. Chem. Eng.*, **27**(1), 27-44, 2003.
- [30] G. Francois, D. Bonvin, Chapter One - Measurement-based real-time optimization of chemical processes. S. Pushpavanam ed. *Control and Optimisation of Process Systems*, vol. 43, Academic Press, 1-50, 2013.
- [31] S. Rasoulilian, L.A. Ricardez-Sandoval, Worst-case and distributional robustness analysis of a thin film deposition process. *IFAC-PapersOnLine*, **48**(8), 1126-1131, 2015.
- [32] Z.K. Nagy, R.D. Braatz, Distributional uncertainty analysis of a batch crystallization process using power series and polynomial chaos expansions. *IFAC Proc. Vol.*, **39**(2), 655-660, 2006.
- [33] M. Skelin, M. Geilen, F. Catthoor, S. Hendseth, Worst-case performance analysis of SDF-based parameterized dataflow. *Microprocess. Microsyst.*, **52**, 439-460, 2017.
- [34] Y. Aliari, A. Haghani, Planning for integration of wind power capacity in power generation using stochastic optimization. *Renew. Sustain. Energy Rev.*, **59**, 907-919, 2016.
- [35] M. Riis, K.A. Andersen, Applying the minimax criterion in stochastic recourse programs. *Eur. J. Oper. Res.*, **165**(3), 569-584, 2005.
- [36] S. Zier, K. Marti, Limit load analysis of plane frames under stochastic uncertainty. D. Weichert and A. Ponter eds. *Limit states of materials and structures*. Springer Netherlands, 113-134, 2009.
- [37] A. Prepoka, *Stochastic programming*. Springer Netherlands, 1995.
- [38] A. Charnes, W. Cooper, G.H. Symonds, Cost horizons and certainty equivalence: An approach in stochastic programming of heating oil. *Manage. Sci.*, **4**, 235-263, 1958.
- [39] A. Charnes, W.W. Cooper, Chance-constrained programming. *Manage. Sci.*, **6**(1), 73-79, 1959.
- [40] T.N. Trần, R. Kreißig, Duc Khôi Vu, M. Staat: Upper bound limit and shakedown analysis of shells using the exact Ilyushin yield surface. *Computers & Structures*, **86**(17-18), 1683-1695, 2008.

- [41] K. Wiechmann, E. Stein, Shape optimization for elasto-plastic deformation under shakedown conditions. *Int J Solids Struct*, **43**(22-23), 7145-7165, 2006.
- [42] J. Atkočiūnas, L. Rimkus, V. Skaržauskas, E. Jarmolajeva, Optimal shakedown design of plates. *Mechanika*, **67**(5), 14-23, 2007.

LINEAR ALGEBRA OF LINEAR AND NONLINEAR BAYESIAN CALIBRATION

Michaël Baudin¹, Régis Lebrun²

¹EDF R&D
6 quai Watier, 78401, Chatou
e-mail: michael.baudin@edf.fr

² Airbus Group
6 rue Marceau, 92130 Issy-Les-Moulineaux
regis.lebrun@airbus.com

Keywords: Linear Algebra, calibration, Bayesian, OpenTURNS

Abstract. *Calibration aims at combining observations with a model in order to reduce the discrepancy between the observations and the predictions of the model by updating its parameters. The Bayesian setup can improve the identifiability of the parameters of the model and amounts to regularizing the problem. The Gaussian prior is largely used by practitioners, mainly because of its ease of use and implementation. The naive implementation of the associated formula, however, can unnecessarily increase the condition number of the matrices involved in the process, in a similar way that the normal equations can increase the condition number of the matrix involved in the least squares problem. This means that the computed parameters may be more sensitive to changes in the data. In this paper, we present a way to use the Cholesky decomposition of the matrices involved in Bayesian calibration. It amounts to compute the Mahalanobis distance using the inverse of the Cholesky factors and leads to the expression of an extended residual which dimension is increased compared to the usual residual. This method can reduce the condition number of the matrices and lead to an improved accuracy in specific cases. We present applications of these ideas and an implementation of this method in the OpenTURNS library.*

Contents

1	Introduction	2
1.1	Purpose	2
1.2	Observations, model and parameters	3
2	Bayesian calibration	5
2.1	Gaussian distribution	5
2.2	Bayesian calibration	6
2.3	Gaussian non linear calibration	7
2.4	Cholesky decomposition for the nonlinear Gaussian calibration	8
2.5	Linear Gaussian calibration	9
2.6	Cholesky decomposition for linear Gaussian calibration	10
3	Calibration of an ill-conditioned exponential model	10
3.1	Description of the model	11
3.2	Linear Gaussian calibration	12
3.3	Value of the cost function	13
3.4	Condition number of the matrices	13
4	Conclusion	15
	Bibliography	15

1 Introduction

1.1 Purpose

When we want to assess the uncertainties in a computer code which simulates a physical system, calibration is an important step. Calibration reduces the discrepancies between the observations and the predictions of the model by adjusting the model parameters [10].

Least squares is the most widely used method of calibration. The implicit assumption of least squares is that the observation errors, i.e. the difference between the observations and the predictions, have a Gaussian distribution. When the computer code is linear, the least squares solution can be computed using linear algebra. In the more general case where the code is nonlinear, numerical optimization methods must be used. In both cases, problems arise when the solution is not identifiable or nearly so.

Bayesian calibration is a way to mitigate the lack of identifiability of the problem. When the prior distribution is not necessarily Gaussian, the posterior distribution of the parameter is generally not known. In this case, the most general-purpose algorithm is to use a Monte-Carlo Markov Chain (MCMC) algorithm such as the Metropolis-Hastings algorithm. The algorithm generates a sample which is designed to have the required posterior distribution. This generally requires many model evaluations, as the algorithm generates a distribution by conditional sampling. This issue can be partially solved by using a surrogate model.

In the special case where the prior is Gaussian, however, more detailed calculations can be handled and this is the main topic of this paper. In the case where the model is linear, the distribution of the posterior random vector is Gaussian, with known mean and covariance matrix. When the model is non linear, the distribution of the posterior is not known, but the

parameter value which has maximum density can be computed: this is the MAP estimator. It requires, however, to solve a non linear optimization problem.

When there is no prior distribution, the implementation of Gaussian least squares calibration based on the *normal equations* involve the Gramian matrix. These equation can be ill-conditioned, although it may happen that a special structure prevent this to happen. In the least squares context, the classical solution to this problem is to use the Cholesky decomposition or orthogonal decompositions such as the QR or SVD decomposition. In the Gaussian calibration framework, there is no known equivalent. The main purpose of this paper is to provide the formula for robust implementation of the linear and non linear Gaussian calibration. These formulas are based on the Cholesky decomposition.

This paper is structured as follows. In the first part, we present the Bayesian Gaussian calibration, and present a method to use the Cholesky decomposition for Gaussian calibration, both in linear and non linear cases.

1.2 Observations, model and parameters

We assume that we observe a quantity for different experimental conditions. These results are gathered in an observation vector \mathbf{y} with dimension n , where n is the number of observations. On the other hand, we consider a model g which predicts this quantity depending on a set of experimental inputs \mathbf{x} and the vector of parameters $\boldsymbol{\theta}$. These variables are formally introduced in the following definition.

Definition 1. (Calibration inputs) *Let $\mathbf{x} \in \mathbb{R}^m$ be the vector of experimental inputs and let $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^m$ the input observables of each experiment, where n is the number of observations.*

We denote by $\boldsymbol{\theta} \in \mathbb{R}^p$ the vector of parameters to calibrate, where p is the number of parameters.

We make the hypothesis that the vector of predictions is produced by the computer model $g : \mathbb{R}^m \times \mathbb{R}^p \rightarrow \mathbb{R}$. In other words, we assume that the relation between the i -th prediction of the computer model and the i -th experimental condition \mathbf{x}_i is:

$$z_i = g(\mathbf{x}_i, \boldsymbol{\theta}) \in \mathbb{R}$$

for $i = 1, \dots, n$. The vector of predictions is $\mathbf{z} \in \mathbb{R}^n$. Let $\mathbf{y} \in \mathbb{R}^n$ be the vector of observations.

In other words, the input data of calibration are the observations \mathbf{y} , the real scalar function g and the experimental conditions $\mathbf{x}_1, \dots, \mathbf{x}_n$.

We make the hypothesis that $n \geq p$, i.e. the number of observations is greater than the number of parameters, which leads to an over-determined problem. Although some methods that we are going to present can be applied when $n < p$ (especially Bayesian methods), this is a situation that we do not often see in our applications, and this is why we do not consider this case in this paper.

In the remaining of the presentation, the *explicit* dependence from z_i to \mathbf{x}_i is not relevant for the development and the analysis of calibration algorithms. Let $\mathbf{h} : \mathbb{R}^p \rightarrow \mathbb{R}^n$ be the function which i -th component is:

$$h_i(\boldsymbol{\theta}) = g(\mathbf{x}_i, \boldsymbol{\theta})$$

for $i = 1, \dots, n$ and any $\boldsymbol{\theta} \in \mathbb{R}^p$. We then use the following compact notation:

$$\mathbf{z} = \mathbf{h}(\boldsymbol{\theta})$$

where $\mathbf{h}(\boldsymbol{\theta}) = (h_1(\boldsymbol{\theta}), \dots, h_n(\boldsymbol{\theta}))^T \in \mathbb{R}^n$.

With these notations, the input data of calibration are the observations \mathbf{y} and the vector model \mathbf{h} .

The following definition introduces the probabilistic hypothesis in which each observation y_i is the sum of the prediction $h_i(\boldsymbol{\theta})$ and a random variable.

Definition 2. (Standard hypothesis of probabilistic calibration) *Let us assume that:*

$$\mathbf{y} = \mathbf{h}(\boldsymbol{\theta}^*) + \boldsymbol{\epsilon} \quad (1)$$

where $\boldsymbol{\theta}^* \in \mathbb{R}^p$ is a fixed, but unknown, value of the parameter $\boldsymbol{\theta}$ and $\boldsymbol{\epsilon} : \Omega \rightarrow \mathbb{R}^n$ is a random vector with zero mean:

$$\mathbf{E}(\boldsymbol{\epsilon}) = \mathbf{0} \in \mathbb{R}^n$$

and finite covariance. In the special case where the error covariance matrix is diagonal, then:

$$\mathbf{Cov}(\boldsymbol{\epsilon}) = (\sigma^*)^2 \mathbf{I}$$

where $\mathbf{I} \in \mathbb{R}^{n \times n}$ is the identity matrix and $\sigma^* > 0$ is the fixed unknown standard error. In the most general case,

$$\mathbf{Cov}(\boldsymbol{\epsilon}) = R^*$$

where $R^* \in \mathbb{R}^{n \times n}$ is a symmetric covariance matrix positive semi-definite.

In some situations, we say that the value $\boldsymbol{\theta}^*$ is the "true" value of the parameter. This allows to distinguish it from the current value $\boldsymbol{\theta}$ and the estimate $\hat{\boldsymbol{\theta}}$.

We further restrict the previous hypotheses and suppose that the R^* is positive definite, such that the associated probability density function is well defined. This condition, more restrictive, implies that eigenvalues of the matrix are positive, which simplifies the definition of the multivariate Gaussian distribution that we will use.

The calibration of the model \mathbf{h} aims at reducing the discrepancy between the predictions of the computer model $\mathbf{h}(\boldsymbol{\theta}) \in \mathbb{R}^n$ and the observations \mathbf{y} by tuning the value of the parameter $\boldsymbol{\theta}$. For a given value of $\boldsymbol{\theta}$, the discrepancy between the predictions and the observations is the *residual*.

Definition 3. (Residuals) *The residual vector is:*

$$\mathbf{r}(\boldsymbol{\theta}) = \mathbf{y} - \mathbf{h}(\boldsymbol{\theta})$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$.

The equation 1 implies that, when the value of $\boldsymbol{\theta}$ is the "true" value $\boldsymbol{\theta}^*$, then the residual is:

$$\mathbf{r}(\boldsymbol{\theta}^*) = \mathbf{y} - \mathbf{h}(\boldsymbol{\theta}^*) = \boldsymbol{\epsilon}.$$

In order to quantify the discrepancy between observations and predictions, we chose the Euclidean norm which is, as we are going to see soon, intrinsically related to the Gaussian distribution.

One of the goals of calibration is to compute the value of the true parameter $\boldsymbol{\theta}^*$. To do this, we define the estimator $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}^*$.

At this stage, we have not set any restriction on the distribution of the residual ϵ . More precisely, the distribution of ϵ is not necessarily Gaussian. In the specific setting where ϵ is Gaussian, however, we can get more details on the solution.

Since the observation vector \mathbf{y} is a random variable, the estimator $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}^*$ depending on \mathbf{y} also is a random variable and a secondary goal of calibration is to get, if possible, the distribution of $\hat{\boldsymbol{\theta}}$ which models the uncertainty of calibration produced by the observation errors.

If the problem is ill conditioned, a small change in the observations \mathbf{y} can lead to a significant change in $\hat{\boldsymbol{\theta}}$. Some methods (e.g. the BLUE, 3DVAR, regularised least squares, Bayesian inversion) integrate some regularisation which mitigates the impact of observation errors on the estimation $\hat{\boldsymbol{\theta}}$ and can manage the lack of identifiability. Some optimization methods like the Levenberg-Marquardt method [6, 7] also integrate some form of regularization. We will illustrate this topic in both the theoretical and practical parts of this paper.

When we calibrate some parameters in order to reduce the discrepancy between observations and measures, different methods with different names have, in fact, the same goal. We sometimes use the term *inversion* [5], *calibration*, be it Bayesian or not, least squares [3] or *data assimilation*, Kalman filter with, often, the same goals. In this paper, we will try to clarify the link between some of these methods.

The sections 2.4 and 2.6 present how to use the Cholesky decomposition in this framework which is, up to our best knowledge, an original contribution on this topic.

2 Bayesian calibration

In this section, we present methods to perform Bayesian calibration. More precisely, we focus on methods in which the distribution of the prior is Gaussian.

2.1 Gaussian distribution

The fundamental probability distribution function in this paper is the Gaussian distribution, which is introduced in the next definition.

Definition 4. (Absolutely continuous multivariate Gaussian distribution.) *Let $\mathbf{X} \in \mathbb{R}^n$ be a random vector in n dimensions with mean $\mathbf{E}(\mathbf{X}) = \boldsymbol{\mu} \in \mathbb{R}^n$ and covariance matrix $\mathbf{Cov}(\mathbf{X}) = \Sigma$ where $\boldsymbol{\mu} \in \mathbb{R}^n$ and $\Sigma \in \mathbb{R}^{n \times n}$ is a symmetric positive definite matrix. We say that \mathbf{X} has an absolutely continuous Gaussian distribution if its probability density function is:*

$$f(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n \det(\Sigma)}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu})\right) \quad (2)$$

for any $\mathbf{x} \in \mathbb{R}^n$.

Since the matrix Σ is, by assumption, symmetric and positive definite, its determinant $\det(\Sigma)$ is nonzero, which guarantees that the denominator of the previous fraction is correctly defined.

In the special case where $\mathbf{Cov}(\mathbf{X}) = \sigma^2 \mathbf{I}$ where $\boldsymbol{\mu} \in \mathbb{R}^n$, $\mathbf{I} \in \mathbb{R}^{n \times n}$ is the identity matrix and $\sigma > 0$, therefore:

$$f(\mathbf{x}) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{\|\mathbf{x} - \boldsymbol{\mu}\|_2^2}{2\sigma^2}\right) \quad (3)$$

for any $\mathbf{x} \in \mathbb{R}^n$.

The previous equation makes clear why the Euclidian norm plays a central role in the Gaussian distribution.

2.2 Bayesian calibration

In this section, we present the Bayesian calibration and, more specifically, the Gaussian Bayesian calibration.

We begin by introducing the prior and posterior distributions.

Definition 5. (Bayesian calibration: prior and posterior distributions.) *We assume that the parameter $\theta \in \mathbb{R}^p$ has a true value θ^* unknown, but constant. We make the hypothesis that the parameter θ has a known distribution $p(\theta)$, called the prior distribution which represents the uncertainty of the true parameter value θ^* . For any $\mathbf{y} \in \mathbb{R}^n$ such that $p(\mathbf{y}) > 0$, Bayes theorem states that the distribution of θ given \mathbf{y} is:*

$$p(\theta|\mathbf{y}) = \frac{p(\mathbf{y}|\theta)p(\theta)}{p(\mathbf{y})}$$

for any $\theta \in \mathbb{R}^p$. The expression $p(\mathbf{y}|\theta)$ is the likelihood. The distribution of $\theta|\mathbf{y}$ is the posterior distribution. Since \mathbf{y} is observed, the denominator of the previous fraction is constant, so that the posterior distribution is proportional to the numerator:

$$p(\theta|\mathbf{y}) \propto p(\mathbf{y}|\theta)p(\theta) \quad (4)$$

for any $\theta \in \mathbb{R}^p$.

The likelihood $p(\mathbf{y}|\theta)$ is the conditional probability density function of the vector of observations \mathbf{y} .

We can now define the Bayesian calibration with Gaussian hypotheses, or, more briefly, the Gaussian calibration.

Definition 6. (Gaussian calibration) *We make the hypothesis that the parameter θ has a Gaussian distribution with known mean and covariance matrices :*

$$\theta \sim \mathcal{N}(\boldsymbol{\mu}, B),$$

where $\boldsymbol{\mu} \in \mathbb{R}^p$ is the mean and $B \in \mathbb{R}^{p \times p}$ is the covariance matrix. The mean of the Gaussian distribution is called the background in data assimilation.

We make the hypothesis that the observations have the following conditional distribution:

$$\mathbf{y}|\theta \sim \mathcal{N}(\mathbf{h}(\theta), R),$$

where $R \in \mathbb{R}^{n \times n}$ is the covariance matrix of observations.

In this Bayesian setting, the values of $\boldsymbol{\mu}$, B et R are known beforehand.

Let us emphasise that in the Gaussian calibration framework, there are two Gaussian distributions: the first is the prior distribution of θ and the second is the \mathbf{y} distribution.

The following theorem is given in [9], p.22-23.

Theorem 1. (Posterior distribution of Gaussian calibration) *We consider hypotheses of definition 6. We denote by $\|\cdot\|_B$ the Mahalanobis distance associated with the matrix B :*

$$\|\theta - \boldsymbol{\mu}\|_B^2 = (\theta - \boldsymbol{\mu})^T B^{-1} (\theta - \boldsymbol{\mu}),$$

for any $\boldsymbol{\theta}, \boldsymbol{\mu} \in \mathbb{R}^p$. We denote by $\|\cdot\|_R$ the Mahalanobis distance associated with the matrix R :

$$\|\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})\|_R^2 = (\mathbf{y} - \mathbf{h}(\boldsymbol{\theta}))^T R^{-1} (\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})).$$

Therefore, the posterior distribution of $\boldsymbol{\theta}$ given the observations \mathbf{y} are:

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto \exp\left(-\frac{1}{2} (\|\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})\|_R^2 + \|\boldsymbol{\theta} - \boldsymbol{\mu}\|_B^2)\right) \quad (5)$$

pour tout $\boldsymbol{\theta} \in \mathbb{R}^p$.

Data assimilation identifies the vector $\hat{\boldsymbol{\theta}}$ associated with the maximum value of the posterior distribution : this is the maximum *a posteriori* estimate or MAP.

Theorem 2. (MAP estimator of Gaussian calibration) *We consider hypotheses of definition 6. The maximum of the posterior distribution of $\boldsymbol{\theta}$ given the observations \mathbf{y} is reached for:*

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta} \in \mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2} (\|\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})\|_R^2 + \|\boldsymbol{\theta} - \boldsymbol{\mu}\|_B^2). \quad (6)$$

We can expand the equation 6 using the definition of the Mahalanobis norm, which leads to the minimisation of the cost function ([1], p.20, p.53):

$$c(\boldsymbol{\theta}) = \frac{1}{2} (\mathbf{y} - \mathbf{h}(\boldsymbol{\theta}))^T R^{-1} (\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})) + \frac{1}{2} (\boldsymbol{\theta} - \boldsymbol{\mu})^T B^{-1} (\boldsymbol{\theta} - \boldsymbol{\mu}) \quad (7)$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$.

2.3 Gaussian non linear calibration

The solution of the data assimilation problem is the solution of the optimization problem defined by theorem 2.

Definition 7. (Cost function of non linear Gaussian calibration) *The cost function of the Gaussian nonlinear calibration is:*

$$c(\boldsymbol{\theta}) = \frac{1}{2} \|\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})\|_R^2 + \frac{1}{2} \|\boldsymbol{\theta} - \boldsymbol{\mu}\|_B^2 \quad (8)$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$. We consider the hypotheses of definition 6. The maximum of the posterior distribution of $\boldsymbol{\theta}$ given the observations \mathbf{y} is reached at:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta} \in \mathbb{R}^p}{\operatorname{argmin}} c(\boldsymbol{\theta}). \quad (9)$$

The 3DVAR algorithm aims at solving the optimization problem in which we search for the minimum of the cost function c . In general, this requires to use a nonlinear optimization algorithm.

The gradient of the cost function C can be explicitly defined depending on the matrix R and B and the gradient of the function \mathbf{h} , which can improve the performance of the optimization algorithm.

Theorem 3. (Unicity of the solution of the optimization problem.) *The Hessian matrix of the cost function of the Gaussian nonlinear calibration problem is symmetric and positive definite. Therefore, the solution of the problem associated with the cost function 9 is unique.*

2.4 Cholesky decomposition for the nonlinear Gaussian calibration

In this section, we present a nonlinear Gaussian calibration which uses the Cholesky decomposition of the covariance matrices.

Theorem 4. (Mahalanobis distance and Cholesky decomposition) *Let $L_B \in \mathbb{R}^{p \times p}$ be the Cholesky factor of the matrix B :*

$$B = L_B L_B^T$$

where L_B is a lower triangular matrix. Therefore, the Mahalanobis distance between $\boldsymbol{\theta} \in \mathbb{R}^p$ and $\boldsymbol{\mu} \in \mathbb{R}^p$ is:

$$\|\boldsymbol{\theta} - \boldsymbol{\mu}\|_B^2 = \|L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu})\|_2^2,$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$. Let $L_R \in \mathbb{R}^{n \times n}$ be the Cholesky factor of the matrix R :

$$R = L_R L_R^T$$

where L_R is a lower triangular matrix. Therefore, the Mahalanobis distance between $\mathbf{y} \in \mathbb{R}^n$ and $\mathbf{h}(\boldsymbol{\theta}) \in \mathbb{R}^n$ is:

$$\|\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})\|_R^2 = \|L_R^{-1}(\mathbf{y} - \mathbf{h}(\boldsymbol{\theta}))\|_2^2,$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$.

Proof. We have

$$\begin{aligned} \|\boldsymbol{\theta} - \boldsymbol{\mu}\|_B^2 &= (\boldsymbol{\theta} - \boldsymbol{\mu})^T B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}) \\ &= (\boldsymbol{\theta} - \boldsymbol{\mu})^T (L_B L_B^T)^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}) \\ &= (\boldsymbol{\theta} - \boldsymbol{\mu})^T (L_B^{-1})^T L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}) \\ &= (L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}))^T L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}) \end{aligned}$$

which concludes the proof for the matrix B . The proof for the matrix R is similar. \square

Theorem 5. (3DVAR and Cholesky decomposition) *Let $\mathbf{r}_e \in \mathbb{R}^{(p+n) \times (p+n)}$ be the extended residual defined by:*

$$\mathbf{r}_e(\boldsymbol{\theta}) = \begin{pmatrix} L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}) \\ L_R^{-1}(\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})) \end{pmatrix} \quad (10)$$

Therefore, the solution of the 3DVAR problem is equivalently the solution of the non linear least squares problem in extended dimension:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta} \in \mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{r}_e(\boldsymbol{\theta})\|_2^2. \quad (11)$$

Proof. Indeed, the theorem 4 allows to express the cost function as:

$$c(\boldsymbol{\theta}) = \frac{1}{2} \|L_R^{-1}(\mathbf{y} - \mathbf{h}(\boldsymbol{\theta}))\|_2^2 + \frac{1}{2} \|L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu})\|_2^2,$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$. It is easy to see that the previous expression is the Euclidian norm of the extended residual defined in equation 10. \square

2.5 Linear Gaussian calibration

In this section, we present the calibration with Gaussian prior distribution in the special case where the model is linear. This method provides the best linear unbiased estimator in this setting, and this is why it is sometimes called the *BLUE*. In data assimilation, this is called *Kalman filter*, with the difference that the Kalman filter is often used sequentially by updating the parameter θ within an iterative loop, instead of being managed in just one pass as we do.

Let us assume that the function \mathbf{h} is linear with respect to θ . In this special case, we can compute the solution of the problem by solving a linear system of equations.

Theorem 6. (Solution of linear Gaussian calibration.) *We consider the hypotheses of the definition 6. We assume that \mathbf{h} is linear with respect to θ , i.e., for any $\theta \in \mathbb{R}^p$, we have:*

$$\mathbf{h}(\theta) = \mathbf{h}(\mu) + J(\theta - \mu). \quad (12)$$

Let A be the matrix:

$$A = (B^{-1} + J^T R^{-1} J)^{-1}. \quad (13)$$

Let K be the Kalman matrix defined by:

$$K = A J^T R^{-1}. \quad (14)$$

Therefore, the unique maximum of the posterior distribution of θ given the observations is ([1], p.53):

$$\hat{\theta} = \mu + K(\mathbf{y} - \mathbf{h}(\mu)). \quad (15)$$

The estimator $\hat{\theta}$ is now defined ; the next theorem introduces the distribution of this estimator.

Theorem 7. (Solution of the linear Gaussian calibration) *We consider the same hypotheses as in the theorem 6. Therefore :*

$$p(\theta|\mathbf{y}) \propto \exp\left(\frac{1}{2}(\theta - \hat{\theta})^T A^{-1}(\theta - \hat{\theta})\right) \quad (16)$$

for any $\theta \in \mathbb{R}^p$ where $\hat{\theta}$ is defined by the equation 15 and the matrix A is given by the equation 13. In other words,

$$\text{Cov}(\hat{\theta}) = A = (B^{-1} + J^T R^{-1} J)^{-1}.$$

The following theorem establishes the covariance of the bayesian Gaussian calibration ([9], p.36 and 66 and [1], p.93-95).

Theorem 8. (Covariance matrix of the linear Gaussian calibration) *Under the hypotheses of the theorem 6, we have:*

$$\hat{\theta} \sim \mathcal{N}(\theta, A)$$

where $\hat{\theta}$ is defined by the equation 15 and matrix A is defined by 13.

2.6 Cholesky decomposition for linear Gaussian calibration

In this section, we present a linear Gaussian calibration method using the Cholesky decomposition of the covariance matrices. This method extends the method presented in section 2.4.

Theorem 9. (Solution of linear Gaussian calibration with Cholesky decomposition) *We consider the same hypotheses as in theorem 6. Therefore, the unique maximum of the posterior distribution of θ given the observations \mathbf{y} is equivalently defined as the solution of the linear least squares problem:*

$$\hat{\theta} = \underset{\theta \in \mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2} \|\bar{A}(\theta - \mu) - \bar{\mathbf{y}}\|_2^2. \quad (17)$$

where $\bar{A} \in \mathbb{R}^{(p+n) \times p}$ is the extended matrix:

$$\bar{A} = \begin{pmatrix} L_B^{-1} \\ -L_R^{-1} J \end{pmatrix} \quad (18)$$

and $\bar{\mathbf{y}} \in \mathbb{R}^{p+n}$ is the extended vector:

$$\bar{\mathbf{y}} = \begin{pmatrix} \mathbf{0} \\ -L_R^{-1}(\mathbf{y} - \mathbf{h}(\mu)) \end{pmatrix}$$

where the p first components of $\bar{\mathbf{y}}$ are zero.

Proof. The proof uses the equation 10 in the special case where \mathbf{h} is linear. The equation 12 implies:

$$\mathbf{y} - \mathbf{h}(\theta) = \mathbf{y} - \mathbf{h}(\mu) - J(\theta - \mu),$$

for any $\theta \in \mathbb{R}^p$. We substitute the previous equation into the extended residual defined by the equation 10:

$$\begin{aligned} \mathbf{r}_e(\theta) &= \begin{pmatrix} L_B^{-1}(\theta - \mu) \\ L_R^{-1}(\mathbf{y} - \mathbf{h}(\mu) - J(\theta - \mu)) \end{pmatrix} \\ &= \begin{pmatrix} L_B^{-1}(\theta - \mu) \\ L_R^{-1}(\mathbf{y} - \mathbf{h}(\mu)) - L_R^{-1}J(\theta - \mu) \end{pmatrix} \\ &= \begin{pmatrix} L_B^{-1}(\theta - \mu) \\ -L_R^{-1}J(\theta - \mu) \end{pmatrix} + \begin{pmatrix} \mathbf{0} \\ L_R^{-1}(\mathbf{y} - \mathbf{h}(\mu)) \end{pmatrix} \\ &= \begin{pmatrix} L_B^{-1} \\ -L_R^{-1}J \end{pmatrix} (\theta - \mu) + \begin{pmatrix} \mathbf{0} \\ L_R^{-1}(\mathbf{y} - \mathbf{h}(\mu)) \end{pmatrix} \\ &= \bar{A}(\theta - \mu) - \bar{\mathbf{y}} \end{aligned}$$

by definition of \bar{A} and $\bar{\mathbf{y}}$. □

3 Calibration of an ill-conditioned exponential model

In this section, we consider the calibration of an exponential model with the linear Gaussian calibration (BLUE). The goal of this section is to quantify the influence of the condition number of the matrices involved onto the results and compare the numerical values obtained by the two methods.

We first consider the method presented in the section 2.5 which uses the Kalman matrix. Then we use the method presented in the section 2.6 which uses the Cholesky decomposition of the covariance matrices B and R .

i	1	2	3	4	5	6	7	8	9	10
y_i	7.125	-1.414	4.099	14.58	1.381	20.19	26.82	52.52	76.96	122.4

Table 1: A sample of ten observed input and the corresponding independent realizations of the observed outputs.

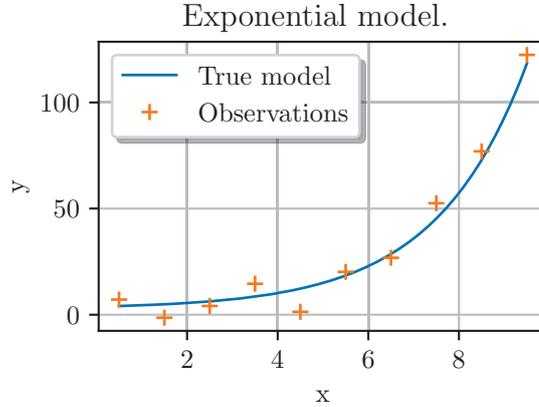


Figure 1: Observations in the exponential model.

3.1 Description of the model

We present the calibration of a model which uses the exponential function. Our aim is to provide the details of the experiment so that it can be reproduced by the interested reader.

The model is the function:

$$g(x) = \theta_1 + \exp(\theta_2 x)$$

for any $x \in [0.5, 9.5]$ where $\theta \in \mathbb{R}^2$ is the vector of parameters. Notice that this model is linear with respect to the parameter θ_1 , but not with respect to θ_2 . We consider $n = 10$ observations. The observed values of the inputs are $x_i = i - 0.5$ for $i = 1, \dots, n$. We assume that the observation error is the random variable $\epsilon \sim \mathcal{N}(0, 5)$, that is, an observation error which has the Gaussian distribution with zero mean and standard deviation equal to 5. We generate the noisy observations of the output by generating independent realization of the observation error:

$$y_i = g(x_i, \theta) + \epsilon_i,$$

for $i = 1, \dots, n$, where $\epsilon_1, \dots, \epsilon_n$ are independent realizations of the random variable ϵ . The observations are the couples $\{(x_i, y_i)\}_{i=1, \dots, n}$.

The true values of the vector of parameters is $\theta^* = (2.8, 0.5)^T$.

In order to reproduce the experiment, we generated a sample of the observations once for all. This sample is presented in the table 1, rounded to 4 significant digits. Using a constant sample allows to reproduce the experiment more easily.

The figure 1 presents the observations and the model.

Let us now describe the parameters of the Gaussian calibration. The mean of the Gaussian prior is $\mu = (1, 1)^T$. We use the prior covariance matrix:

$$B = \begin{pmatrix} 4 & \frac{1}{2} \\ \frac{1}{2} & 7 \end{pmatrix}.$$

Its condition number is $\kappa_2(B) = 1.807$, which is not perfect (because $\kappa_2(B) > 1$), but almost ideal because its order of magnitude is almost equal to the one of a perfectly conditioned matrix.

Moreover, we use a constant variance of the observation errors:

$$\sigma_Y^2 = 3.$$

The Jacobian matrix is computed based on symbolic differentiation:

$$J = \begin{pmatrix} 1 & 0.8243 \\ 1 & 6.723 \\ 1 & 30.46 \\ 1 & 115.9 \\ 1 & 405.1 \\ 1 & 1346 \\ 1 & 4323 \\ 1 & 1.356 \times 10^4 \\ 1 & 4.178 \times 10^4 \\ 1 & 1.269 \times 10^5 \end{pmatrix}$$

In the next paragraphs, we use the Gaussian linear calibration. Since the model is not linear with respect to θ_2 , the BLUE estimator does not provide the exact solution of the calibration problem, but only computes the exact solution of the linearized problem. This is why the exact solution of this problem is not necessarily θ^* , nor μ , but can only be computed from calculation.

3.2 Linear Gaussian calibration

We present the results with 4 significant digits (rounded to nearest), which is sufficient for our purpose. We use a "K" subscript for results which are obtained using the Kalman matrix and the "C" subscript for results which are obtained using the Cholesky decomposition.

Using the Kalman matrix, we get:

$$\hat{\theta}_K = [0.9999, 0.8943].$$

When the computation uses the Kalman matrix, the covariance matrix is A , as defined by the equation 13:

$$A = \text{Cov}(\hat{\theta}_K) = \begin{pmatrix} 3.999 & -4.175 \times 10^{-5} \\ -4.175 \times 10^{-5} & 6.019 \times 10^{-10} \end{pmatrix}.$$

This leads to the 95% confidence interval:

$$\left(\hat{\theta}_1\right)_K \in [-3.267, 5.267] \quad \left(\hat{\theta}_2\right)_K \in [0.8942, 0.8943].$$

With the method based on the extended linear least squares problem using the Cholesky decomposition, we get:

$$\hat{\theta}_C = [-100.4, 0.8953].$$

For this method, the covariance matrix is the Gram matrix:

$$\text{Cov}(\hat{\theta}_C) = \begin{pmatrix} 0.3413 & -3.563 \times 10^{-6} \\ -3.563 \times 10^{-6} & 2.033 \times 10^{-10} \end{pmatrix}.$$

This leads to the following 95% confidence interval:

$$\left(\hat{\theta}_1\right)_C \in [-101.7, -99.08] \quad \left(\hat{\theta}_2\right)_C \in [0.8953, 0.8954].$$

We observe that the results produced by the Kalman matrix seem to be correct, with a reduced confidence interval for the parameter θ_2 , while the parameter θ_1 seem to have a relatively large confidence interval. Based on these results, it seem that the true of θ_1 is approximately in the interval from -3 to 5. This does not match the result obtained from the extended linear least squares method, which computes a relatively close value of the parameter θ_2 , but has a parameter θ_1 much more negative, approximately in the interval from -102 to -99.

The figure 1 may seem to indicate that a value of θ_1 close to 0, as is the case for the method based on the Kalman matrix, may be more likely than a value of θ_1 close to -100, as is the case for the method using the Cholesky decomposition. This is, however, a false conclusion regarding the Gaussian linear calibration problem, which is different from the calibration that might be done visually. Firstly, the method uses a linearized model and, more importantly, it uses a Gaussian prior.

In order to see which result is more accurate, we use two complementary criteria.

- We evaluate the cost function defined by the equation 6: the best value is the lowest. This criterion evaluates the accuracy with respect to the consequences: in terms of stability analysis of algorithms, this the forward error analysis.
- We evaluate the condition number of the matrices involved in the methods: the best method has lowest condition numbers. This criterion evaluates the accuracy with respect to the sources: in terms of stability analysis of algorithms, this the backward error analysis.

3.3 Value of the cost function

The value of the cost function at both optimum points is equal to:

$$c(\hat{\theta}_K) = 5.954 \times 10^4, \quad c(\hat{\theta}_C) = 4.449 \times 10^4.$$

We see that the method using the extended linear least squares problem has a smaller cost function value. Both methods use the same cost function, and differ only by the way they minimise it. Hence, the method which produces a lower value of the cost function achieves a better accuracy.

One possible cause for the difference in the function value may be that the function evaluation is associated with a loss of accuracy. This is in fact impossible because the matrices B and R involved in the Mahalanobis distance are respectively very well and perfectly conditioned. Therefore, the evaluation of the cost function cannot be affected by a massive loss of accuracy.

3.4 Condition number of the matrices

To see how the condition number may magnify the rounding errors in algebraic computations, we shortly describe the accuracy that can be expected when we use 64 bits floating point numbers. The IEEE754 standard for these numbers uses a precision of 53 bits, which leads to a unit roundoff approximately equal to 10^{-16} , that is approximately 16 significant digits (for normalized floating point numbers). We these numbers, when the condition number of function or algorithm is equal to 10^d , the maximum number of lost digits is equal to d (but this upper bound is not always reached). For example, if the condition number of an algorithm is equal to 10^4 , therefore there are at least approximately $16 - 4 = 12$ significant digits in the result. This is why we, quite arbitrarily, write that the condition number is "low" when it is lower than 10^8

(because approximately half of the digits are corrects), "extreme" if it is greater than 10^{16} and "high" otherwise.

Let us now focus on the linear Gaussian calibration. The base 10 logarithm of the condition number of the Kalman matrix is extreme:

$$\log_{10}(\kappa_2(K)) = 20.87.$$

This shows that the use of the Kalman matrix can produce a massive loss of accuracy when we compute $\hat{\theta}_K$ from it, with a potentially total loss of accuracy when we use 64 bits floating point numbers. The condition number of the covariance matrix of the parameter based on the Kalman matrix is also ill-conditioned:

$$\log_{10} \left(\kappa_2 \left(\mathbf{Cov} \left(\hat{\theta}_K \right) \right) \right) = 10.38,$$

although this condition number is not as bad as the previous one.

The condition number of the matrix \bar{A} involved in the extended linear least squares problem 18 is:

$$\log_{10} \left(\kappa_2 \left(\bar{A} \right) \right) = 4.656.$$

This shows that the matrix associated to the computation based on Cholesky decomposition is acceptable. The covariance matrix of $\hat{\theta}_C$ is the Gram matrix of the extended linear least squares problem:

$$\mathbf{Cov}(\hat{\theta}_C) = \hat{\sigma}_C^2 \bar{G}^{-1}. \quad (19)$$

where $\hat{\sigma}_C^2$ is the unbiased estimator of the variance and \bar{G} is the Gram matrix (also known as the information matrix) of the extended linear least squares problem:

$$\bar{G} = \bar{A}^T \bar{A}.$$

The base 10 logarithm of the condition number is:

$$\log_{10} \left(\kappa_2 \left(\mathbf{Cov}(\hat{\theta}_C) \right) \right) = 9.312,$$

which is relatively high. Let us notice, however, that the condition number of $\mathbf{Cov}(\hat{\theta}_C)$ is necessarily equal to the square of the condition number of \bar{A} . Indeed, the equation 19 implies:

$$\kappa_2 \left(\mathbf{Cov}(\hat{\theta}_C) \right) = \kappa_2 \left(\bar{A} \right)^2.$$

Therefore,

$$\log_{10} \left(\kappa_2 \left(\mathbf{Cov}(\hat{\theta}_C) \right) \right) = 2 \log_{10} \left(\kappa_2 \left(\bar{A} \right) \right),$$

which confirms the numerical values we obtained with the exponential model, since $9.312 = 2 \times 4.656$.

In order to analyse in more depth the root causes of the condition numbers of the matrices, let us compute the condition number of the Jacobian matrix, which is involved in both methods:

$$\log_{10} \left(\kappa_2(J) \right) = 4.675.$$

This matrix, which must be managed by both two methods, has therefore a relatively low condition number. We see that its condition number is close to the one of the matrix \bar{A} , which is the expected result given the definition of \bar{A} . Hence, the method which uses the extended linear least squares problem does not artificially increase the condition number of the matrices, as opposed to the method which uses the Kalman matrix.

4 Conclusion

We have analysed the linear and non linear Gaussian calibration of a computer model and presented several methods to compute its solution. While the classical method, which uses the Kalman matrix, is mathematically satisfactory, its implementation in floating point arithmetic artificially reduces the accuracy of the solution and magnifies the errors in the data.

We presented a new method which involves an extended least squares problem. We have shown an example in which the new method actually performs with more accuracy. These methods are implemented in the OpenTURNS software [2].

REFERENCES

- [1] Mark Asch, Marc Bocquet, and Maëlle Nodet. *Data assimilation. Methods, algorithms and applications*. SIAM, 2016.
- [2] Michaël Baudin, Anne Dufloy, Bertrand Iooss, and Anne-Laure Popelin. *OpenTURNS: An Industrial Software for Uncertainty Quantification in Simulation*, pages 2001–2038. Springer International Publishing, Cham, 2017.
- [3] Ake Björck. *Numerical Methods for Least Squares Problems*. Society for Industrial Applied Mathematics, 1996.
- [4] Guillaume Damblin. Calage statistique des paramètres d’un modèle physique de condensation en thermohydraulique à l’échelle système. Technical report, DEN / DANS / DM2S / STMF / LGLS / NT / 2018-63096 / AA, 2018.
- [5] P. C. Hansen. The l-curve and its use in the numerical treatment of inverse problems. In *Computational Inverse Problems in Electrocardiology*, ed. P. Johnston, *Advances in Computational Bioengineering*, pages 119–142. WIT Press, 2000.
- [6] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Mathematics*, 2(2):164–168, 1944.
- [7] Donald Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics*, 11(2):431—441, 1963.
- [8] Gilbert Strang. *Introduction to Linear Algebra*. Wellesley - Cambridge Press, 2009.
- [9] Albert Tarantola. *Inverse problem theory*. SIAM, 2005.
- [10] Timothy G Trucano, Laura Painton Swiler, Takeru Igusa, William L Oberkampf, and Martin Pilch. Calibration, validation, and sensitivity analysis: What’s what. *Reliability Engineering & System Safety*, 91(10-11):1331–1357, 2006.

ASSESSMENT OF VARIANTS OF THE METHOD OF MOMENTS AND POLYNOMIAL CHAOS APPROACHES TO AERODYNAMIC UNCERTAINTY QUANTIFICATION

E.M. Papoutsis-Kiachagias¹, V.G. Asouti¹ and K.C. Giannakoglou¹

¹National Technical University of Athens,
Parallel CFD & Optimization Unit, School of Mechanical Engineering
Zografou Campus, 9 Iroon Polytechniou Str
e-mail: {vpapout,vasouti,kgianna}@mail.ntua.gr

Keywords: Uncertainty Quantification, Method of Moments, Polynomial Chaos, Continuous Adjoint, Aerodynamics

Abstract. *In real life problems uncertainties, for instance through uncertain boundary conditions or manufacturing imperfections, may affect the performance of an aerodynamic shape significantly. In order to measure the impact of uncertainties on the performance of an aerodynamic shape, statistical moments (usually the mean value and variance) of the Quantity of Interest (QoI, e.g. the lift or drag forces) have to be quantified through Uncertainty Quantification (UQ) techniques. This paper compares a number of variants of the Method of Moments (MoM) and the non-intrusive polynomial chaos (niPCE) approaches to UQ. Regarding the MoM, first- and second-order derivatives of the QoI with respect to the uncertain variables are necessary for formulating its first-(FOSM) and second-order (SOSM) variants, respectively. These are computed using a combination of continuous adjoint and direct differentiation of the governing (flow) equations. The statistical moments of the QoI can, then, easily be computed in terms of the QoI value and derivatives computed at the mean values of the uncertain variables. The results of the above-mentioned MoM variants are additionally compared to a number of niPCE variants developed and utilized by the authors in the past. The UQ variants of MoM and niPCE are then compared in terms of cost and accuracy of the computed statistical moments of the QoI. Comparisons also include results of the Monte Carlo method which acts as the reference method. The two benchmark cases used for the comparison include an airfoil under uncertain flow conditions and fluid properties as well as the DrivAer car model, under uncertain flow conditions.*

1 INTRODUCTION

A number of UQ approaches have been developed during the last years to help propagate the uncertainty from the inputs of the aerodynamic analysis problem (e.g. uncertain boundary conditions) to the QoI. A recent review of many of them and their incorporation into robust design optimization loops with a focus on air vehicles can be found in [1]. In general, most UQ methods have a cost that scales with a power of the number M of the uncertain variables $c_i, i \in [1, M]$, making UQ computationally feasible for problems with only a moderate M . This paper focuses on some variants of the Method of Moments (MoM), [2, 3, 4, 5], and non-Intrusive Polynomial Chaos Expansion (niPCE) [6, 7, 8, 9] with a potential for a relatively low UQ cost and compares them in terms of cost and accuracy of computation of the statistical moments of the QoI; Monte Carlo (MC) acts as the reference method for computing the latter.

According to the MoM, the QoI (usually lift or drag for external aerodynamics problems) is expanded into a Taylor series in terms of c . The statistical moments of the QoI are, then, obtained analytically by using this expansion in the integrals that define them. By keeping only the first-order term in the Taylor expansion and computing the first two statistical moments of the QoI, namely its mean and standard deviation, a First-Order Second-Moment (FOSM) UQ method is formulated [4]. If second-order terms are also maintained in the Taylor expansion, a Second-Order Second-Moment (SOSM) approach is devised. The FOSM approach calls for the computation of first-order derivatives of the QoI with respect to (w.r.t.) c while the SOSM approach additionally requires second-order derivatives, a.k.a. the Hessian matrix. First-order derivatives w.r.t. c are computed based on the continuous adjoint method developed by the group of authors in the past [10], at a cost that is independent of the value of M . As discussed in sections 2 and 4, this gives rise to the only UQ method known to the authors with a cost that does not scale with M . The most efficient method to compute second-order derivatives requires the combined use of adjoint and direct differentiation (DD, being the equivalent of the tangent-linear mode in an Automatic Differentiation tool), with a cost that scales linearly with M [11].

Additionally, a number of niPCE approaches are presented and compared to the two MoM variants. These include a) standard quadrature niPCE, b) regression-assisted niPCE computations based on at least as many QoI evaluations as the number of polynomial weights, [9, 12], and c) an adjoint-assisted regression approach by using the sensitivity derivatives of the QoI w.r.t. c , to reduce the computational cost [13]. These variants were also compared to each other in terms of cost and accuracy by the authors in [14] and will not, thus, be analyzed in detail herein.

The rest of this paper is structured as follows: in section 2, the mathematical background of the MoM is presented, including the flow and adjoint equations, as well as the combination of adjoint and DD for the computation of the Hessian matrix. In section 3, the niPCE variants are presented in brief. In section 4, the cost of the two studied MoM variants is compared to that of the aforementioned niPCE approaches and, in section 5, the results of the developed MoM variants are compared with each other as well as with those obtained by niPCE and verified with reference results produced using MC simulations. Finally, conclusions are drawn in section 6.

2 MoM FRAMEWORK

2.1 Flow Equations and QoI

The cases examined in this paper are governed by the steady-state Navier-Stokes PDEs for incompressible flows,

$$R^p = -\frac{\partial v_j}{\partial x_j} = 0 \quad (1a)$$

$$R_i^v = v_j \frac{\partial v_i}{\partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j} + \frac{\partial p}{\partial x_i} = 0, \quad i = 1, 2, 3, \quad (1b)$$

where v_i are the velocity components, $\tau_{ij} = (\nu + \nu_t) \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right)$ the stress tensor components, p the (relative to the exit/reference) static pressure divided by the constant density, ν the (constant) kinematic viscosity of the fluid and ν_t the turbulent viscosity, computed only for turbulent flows. Twice repeated indices within the same term imply summation. In cases where turbulent flow are studied, the Spalart–Allmaras turbulence model [15] is used to effect closure; according to the latter, the turbulent viscosity is given by $\nu_t = \tilde{\nu} f_{v_1}$ and computed after solving the PDE

$$R^{\tilde{\nu}} = v_j \frac{\partial \tilde{\nu}}{\partial x_j} - \frac{\partial}{\partial x_j} \left[\left(\nu + \frac{\tilde{\nu}}{\sigma} \right) \frac{\partial \tilde{\nu}}{\partial x_j} \right] - \frac{c_{b2}}{\sigma} \left(\frac{\partial \tilde{\nu}}{\partial x_j} \right)^2 - \tilde{\nu} P(\tilde{\nu}) + \tilde{\nu} D(\tilde{\nu}) = 0, \quad (2)$$

for the turbulence variable $\tilde{\nu}$. In the above equation, the production and destruction terms are given by $P(\tilde{\nu}) = c_{b1} \tilde{Y}$ and $D(\tilde{\nu}) = c_{w1} f_w(\tilde{Y}) \frac{\tilde{\nu}}{\Delta^2}$ respectively, with $\tilde{Y} = f_{v_3} Y + \frac{\tilde{\nu}}{\Delta^2 \kappa^2} f_{v_2}$, $Y = \|\vec{S}\| = \|e_{ijk} \frac{\partial v_k}{\partial x_j}\|$ the vorticity magnitude and Δ the distance from the wall.

Dealing with external aerodynamics (flows around bodies such as airfoils or cars), eqs. 1 are associated with the following set of boundary conditions,

$$S_I \begin{cases} \mathbf{v} = |v_\infty| [\cos(\alpha_\infty) \sin(\alpha_\infty)]^T \\ \frac{\partial p}{\partial x_j} n_j = 0 \\ \tilde{\nu} = ct \end{cases} \quad S_W \begin{cases} v_i = 0 \\ \frac{\partial p}{\partial x_j} n_j = 0 \\ \tilde{\nu} = 0 \end{cases} \quad S_O \begin{cases} \frac{\partial v_i}{\partial x_j} n_j = 0 \\ p = 0 \\ \frac{\partial \tilde{\nu}}{\partial x_j} n_j = 0, \end{cases} \quad (3)$$

where S_W is the contour of the aerodynamic body and S_I and S_O the parts of the farfield boundary in which the flow enters and exits the domain. Overall, $S = S_I \cup S_O \cup S_W$ is the boundary of the computational domain Ω and $|v_\infty|$, α_∞ are the farfield velocity magnitude and angle, respectively.

Without loss in generality, the drag or lift force coefficients,

$$J = \frac{\int_{S_W} (p \delta_i^j - \tau_{ij}) n_j r_i}{N_F} dS, \quad N_F = \frac{1}{2} A_{ref} U_{ref}^2, \quad (4)$$

is the QoI throughout this paper, where \mathbf{r} is the unit vector aligned with/perpendicular to the farfield velocity for the drag/lift computation, in the absence of uncertainties, A_{ref} is a reference area and U_{ref} is a reference velocity magnitude, usually coinciding with that of the farfield velocity.

2.2 MoM-based UQ

In the MoM, the QoI is developed using a Taylor expansion around the mean values of the uncertain variables, \bar{c} ; the order of the MoM is defined by the largest order of the derivatives of

the QoI w.r.t. \mathbf{c} retained in the Taylor expansion,

$$J(\bar{\mathbf{c}} + \Delta\mathbf{c}) = J|_{\bar{\mathbf{c}}} + \left. \frac{\delta J}{\delta c_i} \right|_{\bar{\mathbf{c}}} \Delta c_i + \frac{1}{2} \left. \frac{\delta^2 J}{\delta c_i \delta c_j} \right|_{\bar{\mathbf{c}}} \Delta c_i \Delta c_j + O(\Delta\mathbf{c}^3). \quad (5)$$

The FOSM approach is formulated by retaining only $\delta J/\delta c_i$ in eq. 5, substituting it into the expressions of the mean value (μ_J) and standard deviation (σ_J) of the QoI and analytically integrating, to obtain, [4],

$$\mu_J(\mathbf{c}) = J|_{\bar{\mathbf{c}}}, \quad \sigma_J(\mathbf{c}) = \sqrt{\sum_{i=1}^M \left[\left. \frac{\delta J}{\delta c_i} \right|_{\bar{\mathbf{c}}} \right]^2 \sigma_i^2}, \quad (6)$$

with σ_i standing for the known standard deviation of the i -th uncertain variable. In the FOSM approach, the expressions of μ_J and σ_J , eqs. 6, are independent of the probability density function (PDF) of \mathbf{c} . Hence, the FOSM-based UQ method can be used with any PDF. In order to avoid any misinterpretation regarding the summation convention, the summation symbol is retained whenever deemed necessary.

If, additionally, we assume that the uncertain variables follow a normal distribution and maintain the second-order derivatives in eq. 5, the SOSM-based statistical moments of J can be computed through

$$\mu_J(\mathbf{c}) = J|_{\bar{\mathbf{c}}} + \frac{1}{2} \sum_{i=1}^M \left[\left. \frac{\delta^2 J}{\delta c_i^2} \right|_{\bar{\mathbf{c}}} \right] \sigma_i^2, \quad \sigma_J(\mathbf{c}) = \sqrt{\sum_{i=1}^M \left[\left. \frac{\delta J}{\delta c_i} \right|_{\bar{\mathbf{c}}} \right]^2 \sigma_i^2 + \frac{1}{2} \sum_{i=1}^M \sum_{j=1}^M \left[\left. \frac{\delta^2 J}{\delta c_i \delta c_j} \right|_{\bar{\mathbf{c}}} \right]^2 \sigma_i^2 \sigma_j^2}, \quad (7)$$

requiring the additional computation of second-order derivatives w.r.t. \mathbf{c} . It is important to note that, in contrast to other UQ methods like nPCE and MC that rely on flow evaluations in a number of combinations of values of the uncertain variables, the MoM depends only on flow and derivative evaluations performed at the nominal operating conditions $\bar{\mathbf{c}}$; for the sake of convenience, the latter index will be omitted hereafter.

The computation of the first- and second-order derivatives required by the above-mentioned MoM is described in brief in the sections that follow.

2.3 Computation of $\frac{\delta J}{\delta c_l}$

First-order derivatives are computed using the continuous adjoint method. According to the latter, an augmented QoI is defined as

$$L = J + \int_{\Omega} u_i R_i^v d\Omega + \int_{\Omega} q R^p d\Omega + \int_{\Omega} \tilde{v}_a R^{\tilde{v}} d\Omega, \quad (8)$$

where Ω is the computational domain, u_i the adjoint velocity components, q the adjoint pressure and \tilde{v}_a the adjoint turbulence variable. Then, eq. 8 is differentiated w.r.t. \mathbf{c} . After setting the multipliers of $\delta v_i/\delta c_l$, $\delta p/\delta c_l$ and $\delta \tilde{v}/\delta c_l$ to zero in the field integrals of the developed form of

$\delta J/\delta c_l$, the continuous adjoint PDEs for incompressible, turbulent flows are derived [10],

$$R^q = -\frac{\partial u_j}{\partial x_j} = 0 \quad (9a)$$

$$R_i^u = u_j \frac{\partial v_j}{\partial x_i} - \frac{\partial(v_j u_i)}{\partial x_j} - \frac{\partial \tau_{ij}^a}{\partial x_j} + \frac{\partial q}{\partial x_i} + \tilde{\nu}_a \frac{\partial \tilde{\nu}}{\partial x_i} - \frac{\partial}{\partial x_l} \left(\tilde{\nu}_a \tilde{\nu} \frac{\mathcal{C}_Y}{Y} e_{mjk} \frac{\partial v_k}{\partial x_j} e_{mli} \right) = 0, \quad i = 1, 2, 3 \quad (9b)$$

$$R^{\tilde{\nu}_a} = -\frac{\partial(v_j \tilde{\nu}_a)}{\partial x_j} - \frac{\partial}{\partial x_j} \left[\left(\nu + \frac{\tilde{\nu}}{\sigma} \right) \frac{\partial \tilde{\nu}_a}{\partial x_j} \right] + \frac{1}{\sigma} \frac{\partial \tilde{\nu}_a}{\partial x_j} \frac{\partial \tilde{\nu}}{\partial x_j} + 2 \frac{c_{b2}}{\sigma} \frac{\partial}{\partial x_j} \left(\tilde{\nu}_a \frac{\partial \tilde{\nu}}{\partial x_j} \right) + \tilde{\nu}_a \tilde{\nu} \mathcal{C}_{\tilde{\nu}} \\ + \frac{\partial \nu_t}{\partial \tilde{\nu}} \frac{\partial u_i}{\partial x_j} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) + (-P + D) \tilde{\nu}_a = 0, \quad (9c)$$

where $\tau_{ij}^a = (\nu + \nu_t) \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$ are the adjoint stress tensor components. The $\mathcal{C}_{\tilde{\nu}}$ and \mathcal{C}_Y expressions can be found in [16].

After satisfying the continuous adjoint PDEs and differentiating eq. 4 w.r.t. \mathbf{c} , the remaining terms of $\delta J/\delta c_l = \delta L/\delta c_l$ read [17],

$$\frac{\delta J}{\delta c_l} = \int_{S_W} \left(\frac{\delta p}{\delta c_l} \delta_i^j - \frac{\delta \tau_{ij}}{\delta c_l} \right) r_i n_j dS + \int_S \left(u_i v_j n_j + \tau_{ij}^a n_j - q n_i + \tilde{\nu}_a \tilde{\nu} \frac{\mathcal{C}_Y}{Y} e_{mjk} \frac{\partial v_k}{\partial x_j} e_{moi} n_o \right) \frac{\delta v_i}{\delta c_l} dS \\ + \int_S u_i n_i \frac{\delta p}{\delta c_l} dS - \int_S u_i n_j \frac{\delta \tau_{ij}}{\delta c_l} dS, - \int_{\Omega} u_i \frac{\partial}{\partial x_j} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) \frac{\delta \nu}{\delta c_l} d\Omega. \quad (10)$$

It should be noted that \mathbf{r} and U_{ref} in eq. 4 are not considered to change w.r.t. \mathbf{c} . Following the methodology presented in [10], the adjoint boundary conditions are formulated by eliminating boundary integrals containing variations of v_i, p, τ_{ij} and $\tilde{\nu}$, where necessary, and read

$$S_I \begin{cases} u_i = 0 \\ \frac{\partial q}{\partial x_j} n_j = 0 \\ \tilde{\nu}_a = 0 \end{cases}, S_W \begin{cases} u_i = -\frac{r_i}{N_F} \\ \frac{\partial q}{\partial x_j} n_j = 0 \\ \tilde{\nu}_a = 0 \end{cases}, S_O \begin{cases} q = u_n v_n + 2\nu \frac{\partial u_i}{\partial x_j} n_i n_j \\ + \tilde{\nu}_a \tilde{\nu} \frac{\mathcal{C}_Y}{Y} e_{mjk} \frac{\partial v_k}{\partial x_j} e_{moi} n_o n_i, \\ u_i v_n + \nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) n_j t_i \\ + \tilde{\nu}_a \tilde{\nu} \frac{\mathcal{C}_Y}{Y} e_{mjk} \frac{\partial v_k}{\partial x_j} e_{moi} n_o t_i = 0 \\ v_j n_j \tilde{\nu}_a + \left(\nu + \frac{\tilde{\nu}}{\sigma} \right) \frac{\partial \tilde{\nu}_a}{\partial x_j} n_j = 0, \end{cases} \quad (11)$$

where \mathbf{n} and \mathbf{t} are the normal and tangential unit vectors and indices n and t indicate the normal and tangential velocity components, respectively. A realistic assumption is that the solution of the adjoint equations costs approximately as much as that of the flow equations. Hence, the flow and adjoint fields are computed at the cost of one Equivalent Flow Solution (EFS) each. EFS is used as the cost unit for the UQ methods that are compared in this paper.

Herein, uncertainties are associated with the flow conditions and properties. In specific, the uncertain variables are the free-stream flow angle, $c_1 = \alpha_\infty$, the magnitude of the farfield velocity, $c_2 = |v_\infty|$ and the flow kinematic viscosity, $c_3 = \nu$. Though the MoM has a comparative cost advantage to other UQ methods as M gets higher, even a case of $M = 3$ is enough for

demonstrating the benefits of the method. Considering the adjoint boundary conditions, the gradient of J w.r.t. c_l is computed as follows,

$$\frac{\delta J}{\delta c_l} = \int_{S_I} (\tau_{ij}^a n_j - q n_i) \frac{\delta v_i}{\delta c_l} dS - \int_{\Omega} u_i \frac{\partial}{\partial x_j} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) \frac{\delta v}{\delta c_l} d\Omega. \quad (12)$$

Since $v_i|_{S_I}$ directly depends on \mathbf{c} , eq. 3, $\frac{\delta v_i}{\delta c_l}|_{S_I}$ is computed analytically as,

$$\frac{\delta \mathbf{v}}{\delta c_1} \Big|_{S_I} = |v_{\infty}| \begin{bmatrix} -\sin(\alpha_{\infty}) \\ \cos(\alpha_{\infty}) \end{bmatrix} \quad \frac{\delta \mathbf{v}}{\delta c_2} \Big|_{S_I} = \begin{bmatrix} \cos(\alpha_{\infty}) \\ \sin(\alpha_{\infty}) \end{bmatrix} \quad \frac{\delta \mathbf{v}}{\delta c_3} \Big|_{S_I} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (13)$$

and $\frac{\delta v}{\delta c_l}$ is straightforward. Hence, all components of $\delta J/\delta c_l$ are computed at a cost of 2 EFS, irrespective of M . This makes the FOSM an affordable UQ method for problems with many uncertain variables.

2.4 Computation of $\frac{\delta^2 J}{\delta c_l \delta c_m}$

The SOSM approach additionally requires the computation of $\frac{\delta^2 J}{\delta c_l \delta c_m}$, a.k.a. the Hessian of J w.r.t. \mathbf{c} . The Hessian matrix can be computed using all possible combinations of adjoint and DD as outlined in [11], there for compressible flows. Based on [11], the most cost-efficient approach to compute the Hessian matrix is based on the DD of the flow equations to compute $\frac{\delta v_i}{\delta c_m}, \frac{\delta p}{\delta c_m}$ and the adjoint method to avoid the computation of $\frac{\delta^2 v_i}{\delta c_m \delta c_l}, \frac{\delta^2 p}{\delta c_m \delta c_l}$; the latter approach is also mentioned as the DD-AV approach in [11] and has a cost of $M + 2$ EFS for computing the Hessian matrix; this includes the solution of the flow and adjoint PDEs. In what follows, the same approach is presented in brief for laminar, incompressible flows.

Let the derivative of any flow quantity ϕ w.r.t. the uncertain variables c_l be denoted as

$$\tilde{\phi}^l = \frac{\delta \phi}{\delta c_l} \quad (14)$$

Then, the derivative of the QoI w.r.t. \mathbf{c} is written as

$$\frac{\delta J}{\delta c_l} \Big|_{DD} = - \int_{S_W} \delta_l^3 E_{ij} \frac{n_j r_i}{N_F} dS - \int_{S_W} \nu \tilde{E}_{ij}^l \frac{n_j r_i}{N_F} dS + \int_{S_W} \tilde{p}^l \frac{n_i r_i}{N_F} dS \quad (15)$$

where $E_{ij} = \frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}$ is the strain tensor and δ_l^3 is the Kronecker symbol. In order to compute \tilde{E}_{ij}^l and \tilde{p}^l , the flow equations, eq. 1, are directly differentiated w.r.t. \mathbf{c} , yielding $l \in [1, 3]$ (in general $l \in [1, M]$) sets of PDEs

$$\tilde{R}^p = - \frac{\partial \tilde{v}_j^l}{\partial x_j} = 0 \quad (16a)$$

$$\tilde{R}_i^v = \tilde{v}_j^l \frac{\partial v_i}{\partial x_j} + v_j \frac{\partial \tilde{v}_i^l}{\partial x_j} - \nu \frac{\partial \tilde{E}_{ij}^l}{\partial x_j} - \delta_l^3 \frac{\partial E_{ij}}{\partial x_j} + \frac{\partial \tilde{p}^l}{\partial x_i} = 0, \quad i = 1, 2, 3 \quad (16b)$$

from which \tilde{p}^l and \tilde{v}_i^l (and, in consequence, \tilde{E}_{ij}^l) can be computed. Eqs. 16 are accompanied by their boundary conditions, which are derived by differentiating eq. 3 w.r.t. \mathbf{c} , yielding

$$S_I \begin{cases} \tilde{\mathbf{v}}^l = \frac{\delta \mathbf{v}}{\delta c_l} \text{ (eq. 13)} \\ \frac{\partial \tilde{p}^l}{\partial x_j} n_j = 0 \end{cases}, S_W \begin{cases} \tilde{v}_i^l = 0 \\ \frac{\partial \tilde{p}^l}{\partial x_j} n_j = 0 \end{cases}, S_O \begin{cases} \frac{\partial \tilde{v}_i^l}{\partial x_j} n_j = 0 \\ \tilde{p}^l = 0. \end{cases} \quad (17)$$

The Hessian is computed by differentiating eq. 15 once more w.r.t. \mathbf{c} and augmenting the outcome with the field integrals of the second-order derivatives of the flow equations, multiplied with appropriate adjoint fields, i.e.

$$\frac{\delta^2 J}{\delta c_m \delta c_l} = \frac{\delta}{\delta c_m} \left(\frac{\delta J}{\delta c_l} \Big|_{DD} \right) + \int_{\Omega} u_i \frac{\delta^2 R_i^v}{\delta c_m \delta c_l} d\Omega + \int_{\Omega} q \frac{\delta^2 R^p}{\delta c_m \delta c_l} d\Omega \quad (18)$$

As it will be proven, the adjoint fields on the right hand side (r.h.s.) of eq. 18 coincide with those computed through eq. 9 (excluding terms stemming from the differentiation of the turbulence model). To help simplify the mathematical notation, let

$$\bar{\phi}^{l,m} = \frac{\delta^2 \phi}{\delta c_m \delta c_l} \quad (19)$$

denote the second-order derivative of any flow quantity ϕ w.r.t. the uncertain variables. Differentiating eq. 15 w.r.t. c_l yields

$$\begin{aligned} \frac{\delta}{\delta c_m} \left(\frac{\delta J}{\delta c_l} \Big|_{DD} \right) &= - \int_{S_W} \delta_l^3 \tilde{E}_{ij}^m \frac{n_j r_i}{N_F} dS - \int_{S_W} \delta_m^3 \tilde{E}_{ij}^l \frac{n_j r_i}{N_F} dS - \int_{S_W} \nu \bar{E}_{ij}^{l,m} \frac{n_j r_i}{N_F} dS \\ &+ \int_{S_W} \bar{p}^{l,m} \frac{n_i r_i}{N_F} dS \end{aligned} \quad (20)$$

Assuming that eqs. 16 have already been solved, the first two integrals on the r.h.s. of eq. 20 can readily be computed. However, evaluating the last two integrals of the r.h.s. of the same equation would require the computation of $\bar{E}_{ij}^{l,m}$ and $\bar{p}^{l,m}$ that would come at a cost scaling with M^2 . To avoid this exponential rise of the UQ cost w.r.t. M , an adjoint system of equations can be formulated expanding the second and third terms on the r.h.s. of eq. 18 as

$$\begin{aligned} \int_{\Omega} u_i \frac{\delta^2 R_i^v}{\delta c_m \delta c_l} d\Omega + \int_{\Omega} q \frac{\delta^2 R^p}{\delta c_m \delta c_l} d\Omega &= \\ \int_{\Omega} \left[u_j \frac{\partial v_j}{\partial x_i} - \frac{\partial (v_j u_i)}{\partial x_j} - \frac{\partial \tau_{ij}^a}{\partial x_j} + \frac{\partial q}{\partial x_i} \right] \bar{v}_i^{l,m} d\Omega - \int_{\Omega} \frac{\partial u_i}{\partial x_i} \bar{p}^{l,m} d\Omega \\ + \int_S (u_i v_j n_j + \tau_{ij}^a n_j - q n_i) \bar{v}_i^{l,m} dS - \int_S \nu u_i n_j \bar{E}_{ij}^{l,m} dS + \int_S u_i n_i \bar{p}^{l,m} dS \\ + \int_{\Omega} u_i \tilde{v}_j^l \frac{\partial \tilde{v}_i^m}{\partial x_j} d\Omega + \int_{\Omega} u_i \tilde{v}_j^m \frac{\partial \tilde{v}_i^l}{\partial x_j} d\Omega - \int_{\Omega} u_i \delta_l^3 \frac{\partial \tilde{E}_{ij}^m}{\partial x_j} d\Omega - \int_{\Omega} u_i \delta_m^3 \frac{\partial \tilde{E}_{ij}^l}{\partial x_j} d\Omega \end{aligned} \quad (21)$$

By setting the multipliers of $\bar{v}_i^{l,m}$ and $\bar{p}^{l,m}$ in the field integrals of eq. 21 to zero, the latter becomes independent of second-order derivatives of the flow variables w.r.t. \mathbf{c} in the interior of the computational domain. This process leads to the formulation of an adjoint system of PDEs that is identical to the one presented in eq. 9. The corresponding adjoint boundary conditions

coincide with those of eq. 11. The remaining terms of eq. 20 and 21 give rise to the expression of the Hessian matrix

$$\begin{aligned} \frac{\delta^2 J}{\delta c_m \delta c_l} = & - \int_{S_W} \delta_l^3 \widetilde{E}_{ij}^m \frac{n_j r_i}{N_F} dS - \int_{S_W} \delta_m^3 \widetilde{E}_{ij}^l \frac{n_j r_i}{N_F} dS + \int_{S_I} (\tau_{ij}^a n_j - q n_i) \overline{v}_i^{l,m} dS \\ & + \int_{\Omega} u_i \widetilde{v}_j^l \frac{\partial \widetilde{v}_i^m}{\partial x_j} d\Omega + \int_{\Omega} u_i \widetilde{v}_j^m \frac{\partial \widetilde{v}_i^l}{\partial x_j} d\Omega - \int_{\Omega} u_i \delta_l^3 \frac{\partial \widetilde{E}_{ij}^m}{\partial x_j} d\Omega - \int_{\Omega} u_i \delta_m^3 \frac{\partial \widetilde{E}_{ij}^l}{\partial x_j} d\Omega \end{aligned} \quad (22)$$

where the components of $\overline{v}_i^{l,m}$ at the inlet are given by

$$\overline{v}^{1,1} = -\mathbf{v}, \quad \overline{v}^{1,2} = \overline{v}^{2,1} = \begin{bmatrix} -\sin(\alpha_\infty) \\ \cos(\alpha_\infty) \end{bmatrix}, \quad \overline{v}^{2,2} = \overline{v}^{1,3} = \overline{v}^{3,1} = \overline{v}^{2,3} = \overline{v}^{3,2} = \mathbf{0} \quad (23)$$

2.5 Flowchart of the MoM approach to UQ

The process of computing μ_J and σ_J using the MoM is summarized in the flowchart that follows, including the computational cost of each step.

- 1 Solution of the flow equations (eqs. 1) at $\mathbf{c} = \overline{\mathbf{c}}$, at the cost of 1 EFS, to obtain the v_i, p and \widetilde{v} fields, followed by the computation of J at $\overline{\mathbf{c}}$.
- 2 Solution of the adjoint equations (eqs. 9), at the cost of 1 EFS, to obtain the u_i, q and \widetilde{v}_a fields.
- 3 Computation of $\frac{\delta J}{\delta c_m}$, $m \in [1, M]$ from eq. 12 (negligible cost).
- 4 **if MoM == FOSM then**
- 5 | Computation of μ_J and σ_J through eq. 6 (negligible cost).
- 6 **else if MoM == SOSM then**
- 7 | Solution of the DD equations, eqs. 16, at the cost of M EFS, to obtain the \widetilde{v}_i^l and \widetilde{p}^l fields, for $l \in [1, M]$.
- 8 | Computation of $\frac{\delta^2 J}{\delta c_m \delta c_l}$ from eq. 22 with $m, l \in [1, M]$ (negligible cost).
- 9 | Computation of μ_J and σ_J through eq. 7 (negligible cost).

3 niPCE-BASED UQ

Assuming that J depends on the vector of uncertain variables $c_i, i \in [1, M]$, niPCE approximates J as

$$J(\mathbf{c}) \approx \sum_{i=0}^{Q-1} J_i H_i(\mathbf{c}), \quad (24)$$

where $Q = \frac{(M+k)!}{M!k!}$, k is the largest degree of the multivariate orthogonal polynomials $H_i(\mathbf{c})$ and J_i are their corresponding weights.

The multivariate polynomials $H_i(\mathbf{c})$ are constructed through the products of univariate orthogonal polynomials that depend on the statistical distribution of the uncertain variables and are chosen from the Wiener-Askey family, [8]. For all applications presented in this paper,

all uncertain variables are assumed to follow a Gaussian distribution and p are the normalized probabilists' Hermite polynomials.

Assuming that the polynomial weights J_i are known, the first and second statistical moments of the QoI are given by, [8]

$$\mu_J = J_0, \sigma_J = \sqrt{\sum_{i=1}^{Q-1} J_i^2} \quad (25)$$

What remains to compute μ_J and σ_J and perform UQ is to compute the coefficients J_i . These are defined through the Galerkin projection of J to $H_i(\mathbf{c})$, i.e.

$$J_i = \int \cdots \int J(\mathbf{c}) H_i(\mathbf{c}) W(\mathbf{c}) d\mathbf{c} \quad (26)$$

where $W(\mathbf{c})$ is the product of the PDFs $w_i, i \in [1, M]$ of the uncertain variables. Numerical approaches for computing J_i , with emphasis on cost reduction in cases of $M \gg$ are described in sections 3.1 and 3.2.

3.1 Gauss Quadrature

Since the polynomial coefficients are given by multidimensional integrals, eq. 26, they can be approximated numerically, by evaluating J at appropriate values of \mathbf{c} . Taking advantage of the fact that the integrand of eq. 26 is a weighted polynomial, Gauss Quadrature rules, [18], can be used to numerically compute J_i . The type of Gauss Quadrature depends on the form of $H_i(\mathbf{c})$ which, in turn, depends on the assumed statistical distribution of \mathbf{c} . For the Gaussian distribution assumed herein, Gauss–Hermite Quadrature (GHQ) is used.

Computing J_i requires $(k + 1)^M$ evaluations of J . This exponential dependency of the cost on M leads to the so-called ‘‘curse of dimensionality’’, making the GQ-based niPCE variant quite costly for cases with even a modest number of uncertain variables. To compensate for its relatively high computational cost, the GQ-based niPCE approach is quite accurate when compared to MC, as it will be shown in section 5.

3.2 Regression-based niPCE and Acceleration through Adjoint-based Gradients

An alternative for computing J_i can be pursued by means of a regression which avoids numerically integrating eq. 26 to compute J_i instead, it approximates them using a more stochastic approach, [9, 12]. In specific, if J is evaluated at L different \mathbf{c} values, eq. 24 can be used to formulate the following system of equations

$$\begin{bmatrix} H_0(\mathbf{c}_1) & \cdots & H_{Q-1}(\mathbf{c}_1) \\ \vdots & \ddots & \vdots \\ H_0(\mathbf{c}_L) & \cdots & H_{Q-1}(\mathbf{c}_L) \end{bmatrix} \begin{bmatrix} J_0 \\ \vdots \\ J_{Q-1} \end{bmatrix} = \begin{bmatrix} J(\mathbf{c}_1) \\ \vdots \\ J(\mathbf{c}_L) \end{bmatrix} \quad (27)$$

with L equations and Q unknowns. If $L = Q = \frac{(M+k)!}{M!k!}$, then eq. 27 can be solved directly to compute the Q unknown coefficients at a cost of Q EFS. To increase accuracy, J is usually oversampled and eq. 27 corresponds to a least squares problem, [19]. For what follows, an oversampling by a factor of $r = 2$ is used, i.e. $L = 2Q$. What remains to be decided are the L different \mathbf{c} points in which J should be evaluated to obtain the right-hand-side (r.h.s.) of eq. 27.

This is still an open issue in the corresponding literature and a number of approaches have been proposed, like random sampling, Latin Hypercube sampling and Hammersley sequence sampling, [20]. In the applications presented in this report, a Latin Hypercube Sampling (LHS) is used. The cost of the regression-based niPCE is compared to the other UQ variants presented in this paper in Table 1.

One flow solution (J evaluation) contributes one line in the system of eq. 27. Adding more lines to eq. 27 for each flow solution which is carried out could further accelerate the UQ process. To do so, the (continuous) adjoint method for computing sensitivity derivatives presented in section 2.3 can be employed. The latter can provide all the components of $\delta J / \delta c_i, i \in [1, M]$ by additionally solving the adjoint PDEs, at a cost of 1 EFS. Assuming an oversampling by a factor of $r > 1$, the sampling points required to obtain rQ equations for computing J_i are¹ $P = \left\lfloor \frac{r(M+k)!}{(M+1)!k!} \right\rfloor = \left\lfloor \frac{rQ}{M+1} \right\rfloor$, i.e. approximately one order of M lower than that of the typical regression approach given by eq. 27. The adjoint-assisted regression approach reads, [13]

$$\begin{bmatrix} H_0(\mathbf{c}_1) & \dots & H_{Q-1}(\mathbf{c}_1) \\ \frac{\partial H_0}{\partial c_1}(\mathbf{c}_1) & \dots & \frac{\partial H_{Q-1}}{\partial c_1}(\mathbf{c}_1) \\ \vdots & \vdots & \vdots \\ \frac{\partial H_0}{\partial c_M}(\mathbf{c}_1) & \dots & \frac{\partial H_{Q-1}}{\partial c_M}(\mathbf{c}_1) \\ \vdots & \ddots & \vdots \\ H_0(\mathbf{c}_L) & \dots & H_{Q-1}(\mathbf{c}_L) \\ \frac{\partial H_0}{\partial c_1}(\mathbf{c}_L) & \dots & \frac{\partial H_{Q-1}}{\partial c_1}(\mathbf{c}_L) \\ \vdots & \vdots & \vdots \\ \frac{\partial H_0}{\partial c_M}(\mathbf{c}_L) & \dots & \frac{\partial H_{Q-1}}{\partial c_M}(\mathbf{c}_L) \end{bmatrix} \begin{bmatrix} J_0 \\ \vdots \\ J_{Q-1} \end{bmatrix} = \begin{bmatrix} J(\mathbf{c}_1) \\ \frac{\delta J}{\delta c_1}(\mathbf{c}_1) \\ \vdots \\ \frac{\delta J}{\delta c_M}(\mathbf{c}_1) \\ \vdots \\ J(\mathbf{c}_L) \\ \frac{\delta J}{\delta c_1}(\mathbf{c}_L) \\ \vdots \\ \frac{\delta J}{\delta c_M}(\mathbf{c}_L) \end{bmatrix} \quad (28)$$

and its cost, taking into consideration that each of the P sampling points costs one flow and one adjoint solution, is given in Table 1. It can be observed that adjoint-assisted regression has the lowest cost of all niPCE variants for $M > 1$, with the gain increasing considerably in the presence of many uncertain variables. Indicatively, for the widely used case of $M = 3, k = 2$, adjoint-assisted regression has half the cost of the typical regression and almost one third of the GQ one.

It should be noted that the cost of the adjoint-assisted regression mentioned in Table 1 is valid in cases with only one QoI. Since the computation of sensitivity derivatives for many QoI requires the solution of as many adjoint PDEs, the cost benefit of the adjoint-assisted regression is mitigated in such cases. For the general case with N_Q QoI and an oversampling by a factor of r , the cost of the adjoint-assisted regression approach is $(N_Q + 1) \left\lfloor \frac{r(M+k)!}{(M+1)!k!} \right\rfloor$ EFS. Neglecting the potential necessity for rounding, the cost ratio of the adjoint-assisted regression and the typical regression is $\frac{N_Q+1}{M+1}$. This means that in cases with more uncertain variables than QoI, adjoint-assisted regression is still more efficient than the typical regression. On the other hand, in cases where $N_Q > M$, the typical regression should be preferred.

¹It should be noted that $\frac{r(M+k)!}{(M+1)!k!}$ may not be an integer, so it has to be rounded to the closest one. The $\lfloor \cdot \rfloor$ sign indicates the floor operation.

4 COST COMPARISON OF THE VARIOUS UQ METHODS

For a problem with M uncertain variables, the cost of the two MoM variants analyzed in section 2.2 is compared with that of the PCE variants briefly presented in section 3. Assuming an niPCE UQ approach with a degree of k , the cost of the niPCE and MoM variants studied thus far is summarized in Table 1. Since one set of adjoint PDEs has to be solved to compute the gradient of each QoI, the UQ variants that employ it have a cost that depends on the number of QoI, N_Q . For the regression-based niPCE variants, r is the oversampling factor.

Method	Cost (EFS)	Sample points
FOSM	$1 + N_Q$	1
SOSM	$M + 1 + N_Q$	1
niPCE - GQ	$(k + 1)^M$	$(k + 1)^M$
niPCE - Regression	$\frac{r(M+k)!}{M!k!}$	$\frac{r(M+k)!}{M!k!}$
niPCE - Regression - Adjoint	$(N_Q + 1) \left\lfloor \frac{r(M+k)!}{(M+1)!k!} \right\rfloor$	$\left\lfloor \frac{r(M+k)!}{(M+1)!k!} \right\rfloor$

Table 1: CPU cost, measured in EFS, for computing (μ_J, σ_J) with a number of niPCE and MoM variants.

Since SOSM treats J as a second-order polynomial of \mathbf{c} , it is interesting to compare its cost with the niPCE variants in case $k = 2$, i.e. when niPCE also treats J as a second-order polynomial. This is summarized in Table 2, for various M values. It can be observed that for $M > 1$, SOSM has half the cost of the cheapest niPCE variant and a significantly smaller one than all other niPCE variants as M increases.

		niPCE - GQ / niPCE - Regression / niPCE - Regression -Adjoint / SOSM					
k	M	1	2	3	4	5	6
2		3/6/6/3	9/12/8/4	27/20/10/5	81/30/12/6	243/42/14/7	729/56/16/8

Table 2: CPU cost, measured in EFS, for computing (μ_J, σ_J) with a number of niPCE variants and the SOSM approach, for a single QoI. All approaches included in this table approximate J as a second-order polynomial of \mathbf{c} . An oversampling by a factor of 2 is used for both regression-based approaches. The result with the lowest cost is marked in bold for each (M, k) pair.

5 VERIFICATION – APPLICATIONS

The results of the MoM variants presented in section 2 are compared with each other, with various niPCE variants presented in section 3 and, for the 2D case, verified using MC simulations. Upon obtaining these J values, μ_J and σ_J can directly be computed through their definitions. Due to its simplicity, MC is used as a benchmark method for validating other UQ methods. On the other hand, the fact that it requires a very large number of J evaluations makes it computationally infeasible for the 3D case examined in Section 5.2.

5.1 NACA0012 airfoil

In the first case examined, UQ is performed for the flow over the NACA0012 isolated airfoil, fig. 1, under uncertain flow conditions and properties. The flow is laminar, $Re = 2000$, drag and lift coefficients are used as the QoI and uncertainties emanate from the farfield velocity and angle, as well as the fluid kinematic viscosity. The denominator of the force coefficients is considered constant. The mean values and standard deviations of the three uncertain variables ($M = 3$) following a normal distribution are listed in Table 3. The flow is solved on grid consisting of 37800 quadrilateral elements.

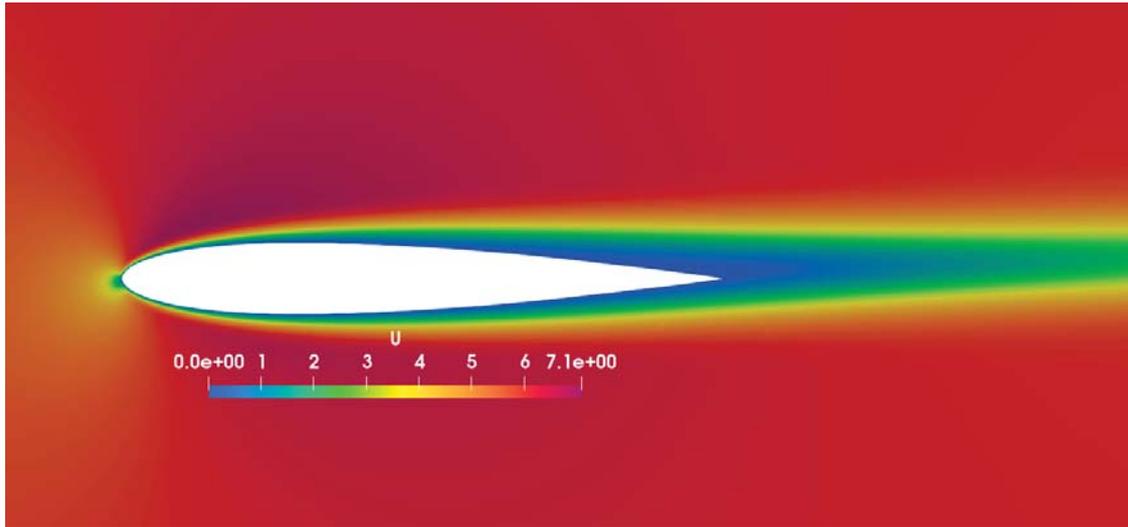


Figure 1: NACA0012: velocity magnitude contours around the airfoil, computed using the mean values of the uncertain variables.

Uncertain variable	μ	σ
Farfield velocity magnitude (m/s)	6	0.6
Farfield velocity angle (deg)	2	0.2
Kinematic viscosity (m^2/s)	3×10^{-3}	1×10^{-4}

Table 3: NACA0012: Mean values and standard deviations of the uncertain variables.

Using the mean values and standard deviations of Table 3, two UQ scenarios are studied. In the first one, only the farfield velocity magnitude and angle are considered as uncertain variables ($M = 2$) while in the second one, all three uncertain variables are used. In both scenarios, second-order polynomials are used ($k = 2$) for the niPCE variants and MC relies upon 1000 evaluations. The μ_J and σ_J approximations based on the MoM variants presented in section 2 along with the corresponding values computed based on a number of niPCE variants presented in section 3 are listed in Tables 4 and 5 for $M = 2$ and $M = 3$, respectively, together with their relative error compared to MC.

Compared to MC, FOSM computes μ_J with a relative error that is smaller than 0.38% and SOSM has an even smaller maximum deviation from MC of 0.19%. For the drag coefficient, SOSM even has the smaller deviation from MC than all other UQ methods tested herein. On the other hand, deviations of the MoM-based σ_J values from MC are higher than the ones of the niPCE variants, especially for the lift coefficient, for which a deviation of 8.8% is observed. From the FOSM- and SOSM-based statistical moments in Tables 4 and 5, it can be

observed that SOSM consistently improves the μ_J predictions over FOSM, this is not however the case for σ_J . This can be explained by analyzing the FOSM- and SOSM-based expressions of σ_J , eqs. 6 and 7, respectively. From there, it can be observed that the SOSM-based σ_J value will always be larger than the FOSM-based one, due to the addition of a positive quantity $\left(\frac{1}{2} \sum_{i=1}^M \sum_{j=1}^M \left[\frac{\delta^2 J}{\delta c_i \delta c_j}\right]_{\bar{c}}^2 \sigma_i^2 \sigma_j^2\right)$ to the FOSM-based σ_J^2 value. Hence, if the FOSM-based σ_J value is overestimated, SOSM can only increase this deviation from the reference value of MC. The MoM-based σ_J values for the drag coefficient are more accurate than those for the lift coefficient (max. deviation of 1.2% from MC).

Summarizing the findings of this study, we can deduce that both FOSM and SOSM are quite accurate when approximating the μ_J values, with the SOSM consistently outperforming the FOSM method for this statistical moment. On the other hand, the MoM-based σ_J values may deviate from the reference ones. Nevertheless, the small cost of the MoM-based approach and especially that of FOSM that is independent of M , makes it a useful tool for robust design optimization loops, as presented for instance in [21].

Method	Cost	Lift		Drag	
		μ_J (% Diff)	σ_J (% Diff)	μ_J (% Diff)	σ_J (% Diff)
FOSM	2	0.09538 (-0.38)	0.02070 (8.8)	0.08463 (-0.2)	0.01257 (0.9)
SOSM	4	0.09557 (-0.19)	0.02070 (8.8)	0.08472 (-0.16)	0.01257 (0.9)
niPCE-GQ	9	0.09577 (0.02)	0.01912 (0.5)	0.08463 (-0.2)	0.01254 (0.6)
niPCE-Regression	12	0.09571 (-0.04)	0.01911 (0.4)	0.08619 (1.6)	0.01204 (-3)
niPCE-Regression-Adjoint	8	0.09583 (0.08)	0.02012 (5)	0.08463 (-0.2)	0.01253 (0.5)
MC	1000	0.09575	0.01902	0.08486	0.01246

Table 4: NACA0012: Case with $M=2$. Mean value and standard deviation of the drag and lift coefficients as the QoI computed with the MoM variants of section 2 and the niPCE variants of section 3, compared with the outcome of MC. Numbers in the parentheses indicative the relative error w.r.t. MC. Cells corresponding to the lowest cost and smallest absolute value of relative error are marked in bold.

Method	Cost	Lift		Drag	
		μ_J (% Diff)	σ_J (% Diff)	μ_J (% Diff)	σ_J (% Diff)
FOSM	2	0.09538 (-0.36)	0.02071 (8.77)	0.08463 (-0.24)	0.01265 (1.2)
SOSM	5	0.09557 (-0.17)	0.02071 (8.77)	0.84723 (-0.13)	0.01265 (1.2)
niPCE-GQ	27	0.09575 (0.02)	0.01913 (0.5)	0.08462 (-0.25)	0.01263 (0.99)
niPCE-Regression	20	0.09584 (0.11)	0.01915 (0.61)	0.08463 (-0.23)	0.01261 (0.85)
niPCE-Regression-Adjoint	10	0.09529 (-0.47)	0.02001 (5.08)	0.08462 (-0.25)	0.01262 (0.9)
MC	1000	0.09573	0.01904	0.08483	0.01251

Table 5: NACA0012: Case with $M=3$. Notation as in Table 4.

5.2 DrivArer Car Model

The DrivArer car model, [22], developed by the Institute of Aerodynamics and Fluid Mechanics of TU Munich, is studied in this section. In specific, the fast-back configuration with a smooth underbody, with mirrors and wheels (F_S_wm_ww) is used as a test case, fig. 2. The drag coefficient is used as the QoI and the fafield velocity magnitude and angle are treated as uncertain variables, with mean values and standard deviations given by Table 6. The denominator of the drag coefficient is considered constant. For the niPCE variants, a max. polynomial degree of $k=2$ is selected.

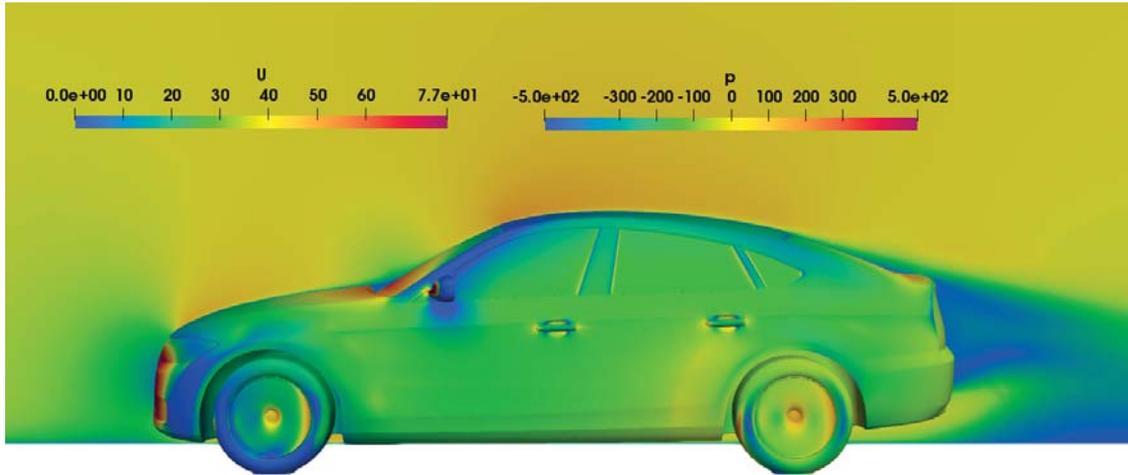


Figure 2: DrivAer: pressure contours on the car surface and velocity contours plotted on a slice along the symmetry plane of the car, computed using the mean values of the uncertain variables.

Uncertain variable	μ	σ
Farfield velocity magnitude (m/s)	38.9	1
Farfield velocity angle (deg)	0	2

Table 6: DrivAer: Mean values and standard deviations of the uncertain variables.

A hex-dominated mesh of 12 million cells is used and the steady-state RANS equations are solved, together with the Spalart–Allmaras turbulence model, [15]. Each flow evaluation takes approximately an hour in 120 cores. Due to the large computational cost, MC is not performed for this test case. The FOSM approach is instead compared to the GQ and regression variants of niPCE. The latter is conducted using the $(k+1)^M = 9$ Gauss nodes used in GQ as the samples of the regression method. It can be seen that the μ_J computed by the FOSM method does not differ much from the niPCE results (relative difference of 1.9%). Regarding σ_J , a more significant deviation of 12% is observed.

Method	Cost	Drag	
		μ_J	σ_J
FOSM	2	0.32663	0.01668
niPCE-GQ	9	0.33305	0.01908
niPCE-Regression	9	0.33306	0.01946

Table 7: DrivAer: Mean value and standard deviation of the drag coefficient computed with FOSM and two niPCE variants.

6 SUMMARY – CONCLUSIONS

In this paper, two Method of Moments approaches, namely First- and Second-Order Second-Moment (FOSM and SOSM), used for propagating uncertainties from the Quantities of Interest, were analysed in terms of cost and predictive accuracy.

The FOSM approach requires the sensitivity derivatives of the QoI w.r.t. the uncertain variables, which were computed herein using continuous adjoint, at a cost of one Equivalent Flow

Solution, irrespective of the number of uncertain variables M . This gives rise to a UQ method with a cost that does not scale with M and is equal to 2 EFS, making it ideal for UQ problems with many uncertain variables. SOSM additionally requires the computation of the Hessian matrix of the QoI with respect to the uncertain variables. This is computed using a combination of the adjoint fields already computed for FOSM and the Direct Differentiation of the flow equations, computing the variations of the flow fields w.r.t. the uncertain variables at a cost that scales linearly with M ; the total cost of the SOSM-based UQ process is $M + 2$ EFS. Even though the cost of SOSM scales with M , the fact that it only scales linearly and with a unitary multiplier of M makes it twice as efficient as the cheapest non-intrusive Polynomial Chaos Expansion method, which also utilized adjoint-based sensitivity derivatives of the QoI w.r.t. the uncertain variables. Nevertheless, the need for second-order derivatives makes its development and utilization tedious for applications with complex numerical models, like turbulent flows involving wall functions.

Regarding the accuracy of the two MoM approaches, the first two statistical moments of the lift and drag coefficients were computed for two UQ problems pertaining to the flow around a 2D airfoil, with uncertainties emanating from the farfield conditions and the fluid kinematic viscosity; these statistical moments were then verified with results obtained from Monte Carlo simulations, acting as the reference method, as well as values obtained through a number of niPCE variants. It was observed that both FOSM and SOSM compute the mean value of the QoI with high accuracy, with SOSM consistently outperforming FOSM. On the other hand, both MoM approaches exhibited a considerable difference (8.8%) from the MC results for the standard deviation of the lift coefficient, with a better behaviour observed for σ_J of the drag coefficient. In addition, it was noticed that if FOSM over-predicts the standard deviation, SOSM can only make the prediction worse since it adds an always constant contribution to the σ_J computed by FOSM.

Finally, the FOSM-based statistical moments of the drag coefficient of the DrivAer car model were compared to those computed with niPCE; the farfield velocity magnitude and angle were considered as the uncertain variables. The mean value was computed with an acceptable accuracy (1.9% deviation from the niPCE results), however the standard deviation exhibited a higher relative difference of 12%. Nevertheless, the low cost of the FOSM approach makes it an ideal candidate as the UQ method used to compute the objective functions of robust design optimization loops, as already presented by the group of authors in other publications.

Acknowledgments

This project has received funding from the Hellenic Foundation for Research and Innovation (HFRI) and the General Secretariat for Research and Technology (GSRT), under grant agreement No 603.

REFERENCES

- [1] Z. Huan, G. Zhenhong, X. Fang, and Z. Yidian. Review of robust aerodynamic design optimization for air vehicles. *Archives of Computational Methods in Engineering*, 26:685–732, 2019.
- [2] R.W. Waters and L. Huyse. Uncertainty analysis for fluid mechanics with applications. *NASA/CR 2002*, 211449, 2002.

- [3] M.M. Putko, P.A. Newman, A.C. Taylor, and L.L. Green. Approach for uncertainty propagation and robust design in CFD using sensitivity derivatives. In *AIAA Paper 2001-2528, 15th Computational Fluid Dynamics Conference*, Anaheim, CA, 2001.
- [4] E.M. Papoutsis-Kiachagias, D.I. Papadimitriou, and K.C. Giannakoglou. Robust design in aerodynamics using third-order sensitivity analysis based on discrete adjoint. Application to quasi-1D flows. *International Journal for Numerical Methods in Fluids*, 69(3):691–709, 2012.
- [5] D.I. Papadimitriou and K.C. Giannakoglou. Third-order sensitivity analysis for robust aerodynamic design using continuous adjoint. *International Journal for Numerical Methods in Fluids*, 71(5):652–670, 2013.
- [6] C. Lacor, C. Dinescu, C. Hirsch, and S. Smirnov. *Implementation of intrusive polynomial chaos in CFD codes and application to 3D Navier-Stokes*, pages 193–223. Springer International Publishing, 2013.
- [7] C. Dinescu, S. Smirnov, C. Hirsch, and C. Lacor. Assessment of intrusive and non-intrusive non-deterministic CFD methodologies based on polynomial chaos expansion. *International Journal of Engineering Systems Modeling and Simulations*, 2:87–98, 2010.
- [8] D. Xiu and G.M. Karniadakis. Modeling uncertainty in flow simulations via generalized polynomial chaos. *Journal of Computational Physics*, 187:137–167, 2003.
- [9] D. Xiu and J. Hesthaven. High-order collocation methods for differential equations with random inputs. *SIAM Journal of Scientific Computing*, 27:1118–1139, 2005.
- [10] E.M. Papoutsis-Kiachagias and K.C. Giannakoglou. Continuous adjoint methods for turbulent flows, applied to shape and topology optimization: Industrial applications. *Archives of Computational Methods in Engineering*, 23(2):255–299, 2016.
- [11] D.I. Papadimitriou and K.C. Giannakoglou. Direct, adjoint and mixed approaches for the computation of Hessian in airfoil design problems. *International Journal for Numerical Methods in Fluids*, 56(10):1929–1943, 2008.
- [12] D. Xiu. Fast numerical methods for stochastic computations: A review. *Communications in computational physics*, 5:242–222, 2009.
- [13] J. Peng, J. Hampton, and A. Doostan. On polynomial chaos expansion via gradient-enhanced l_1 -minimization. *Journal of Computational Physics*, 310:440 – 458, 2016.
- [14] E.M. Papoutsis-Kiachagias, V.G. Asouti, and K.C. Giannakoglou. Polynomial chaos-based, adjoint-enabled uncertainty quantification and robust design for aerodynamic problems. In *4th International Conference on Uncertainty Quantification in Computational Sciences and Engineering, UNCECOMP 2019*, Crete island, Greece, 24-26 June 2019.
- [15] P. Spalart and S. Allmaras. A one-equation turbulence model for aerodynamic flows. In *AIAA Paper 1992-0439, 30th Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, 6-9 January 1992.

- [16] A.S. Zymaris, D.I. Papadimitriou, K.C. Giannakoglou, and C. Othmer. Continuous adjoint approach to the Spalart-Allmaras turbulence model for incompressible flows. *Computers & Fluids*, 38(8):1528–1538, 2009.
- [17] I.S. Kavvadias, E.M. Papoutsis-Kiachagias, and K.C. Giannakoglou. On the proper treatment of grid sensitivities in continuous adjoint methods for shape optimization. *Journal of Computational Physics*, 301:1–18, 2015.
- [18] GH. Golub and Welsch JH. Calculation of gauss quadrature rules. *Mathematics of Computation*, 22:221–230, 1969.
- [19] H. Zhao, Z. Gao, Y. Gao, and C. Wang. Effective robust design of high lift NLF airfoil under multi-parameter uncertainty. *Aerospace Science and Technology*, 68:530–542, 2017.
- [20] S. Hosder, R. Walters, and M. Balch. Efficient sampling for non-intrusive Polynomial Chaos applications with multiple uncertain input variables. In *48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2007.
- [21] K.B. Fragkos, E.M. Papoutsis-Kiachagias, and K.C. Giannakoglou. pFOSM: An efficient algorithm for aerodynamic robust design based on continuous adjoint and matrix-vector products. *Computers & Fluids*, 181:57–66, 2019.
- [22] A. Heft, T. Indinger, and N. Adams. Experimental and numerical investigation of the DrivAer model. In *ASME 2012, Symposium on Issues and Perspectives in Automotive Flows*, pages 41–51, Puerto Rico, USA, 8-12 July 2012.

SOFTWARE FOR UNCERTAINTY PROPAGATION AND RELIABILITY ASSESSMENT OF INELASTIC WIND EXCITED SYSTEMS

Wei-Chu Chuang¹ and Seymour MJ Spence²

¹ Department of Civil and Environmental Engineering, University of Michigan
2350 Hayward St, Ann Arbor, MI, USA
wechuang@umich.edu

² Department of Civil and Environmental Engineering, University of Michigan
2350 Hayward St, Ann Arbor, MI, USA
smjs@umich.edu

Abstract

This paper presents a software tool for carrying out reliability assessment of inelastic wind excited systems through direct stochastic simulation. The tool address the need for practical approaches that enable efficient evaluation of the safety of such systems in order to apply probabilistic performance-based design in wind engineering. In particular, the software application is based on a recently proposed reliability assessment framework that integrates efficient inelastic response estimation approaches with a novel stochastic simulation scheme to propagate a full range of code compliant uncertainties through inelastic systems. The eight tabs of the graphical user interface of the software are designed to offer a user-friendly environment to specify all relevant data in order to define and solve reliability analysis problems that are at the core of modern performance-based wind engineering. The potential and applicability of the software are illustrated on a 3-dimensional archetype building.

Keywords: Software, Uncertainty Quantification, Reliability Assessment, Stochastic Wind Loads, Dynamic Systems.

1 INTRODUCTION

With the introduction of performance-based wind engineering, the potential of designing wind excited systems with controlled inelasticity has attracted growing interest and created a need for tools to efficiently evaluate the safety of such systems while considering a full range of uncertainties. To this end, an efficient framework that enables rapid uncertainty propagation has been developed for assessing the reliability of systems experiencing inelasticity [1]. In particular, in estimating inelastic responses for each sample of the stochastic simulation, strain-driven dynamic shakedown approaches have been developed to not only rapidly identify structural safety against common wind related failure mechanisms, e.g., low cycle fatigue and ratcheting, but also efficiently evaluate the plastic strains and deformations occurring at shakedown while considering both concentrated and distributed plasticity [2],[3]. To further provide information on the inelastic responses beyond shakedown as well as any nonlinear response time history of interest, an alternative adaptive fast nonlinear analysis (AFNA) approach that efficiently integrates the responses over the actual wind load history has been proposed for direct integration with the reliability assessment framework [4].

In order to help bridge the gap between state-of-the-art research and current practice in wind design, this paper presents a comprehensive software tool that enables the full transition from proof-of-concept to practice of reliability estimation through direct uncertainty propagation. The graphical user interface (GUI) of the tool consists in eight tabs that are designed for automatically carrying out all stages of the analysis, including defining the finite element model and model uncertainties, calibrating the stochastic wind load model, executing the reliability analysis by propagating uncertainty through the system by stochastic simulation, and finally providing options to display or save simulation results. An application to a full scale archetype building is presented to illustrate the practicality and efficiency of the software in propagating a full range of code compliant uncertainties for reliability assessment of wind excited structures.

2 SOFTWARE FOR UNCERTAINTY PROPAGATION AND RELIABILITY ASSESSMENT

2.1 Efficient Reliability Assessment Framework for Inelastic Wind Excited Systems

The reliability of a structure can be directly measured in terms of the failure probability against a limit state of interest. In particular, the failure limit state can be expressed as $g(\mathbf{Y}) = 0$. By convention, failure is defined as $g(\mathbf{Y}) < 0$. The associated failure probability can then be evaluated as:

$$P_f = P(g(\mathbf{Y}) < 0) = \int \cdots \int I[g(\mathbf{Y})] f_{\mathbf{Y}}(\mathbf{y}) d\mathbf{y}, \quad I[g(\mathbf{Y})] = \begin{cases} 1, & \text{if } g(\mathbf{Y}) < 0 \\ 0, & \text{if } g(\mathbf{Y}) \geq 0 \end{cases} \quad (1)$$

where \mathbf{Y} is a vector of random variables, including uncertainties in both structural system and external loads. In this work, all random variables in the analysis were carefully chosen so as to be compliant with those considered in the derivation of load factors stipulated in design codes [5]-[7]. In order to investigate the system level reliability of the structure and the possibility of designing buildings with controlled inelasticity, a reliability assessment framework considering not only traditional limit states, e.g., component yield, but also system-level inelastic limit states has been proposed based on the concept of dynamic shakedown [1]. In particular, reliabilities are estimated for the following four limit states (LS) of interest:

1. LS1: component-level yield limit state (traditional limit state used in current design);
2. LS2: system-level first yield limit state;

3. LS3: system-level inelastic limit state (defined as the failure to achieve the state of dynamic shakedown);
4. LS4: inelastic displacement-based limit states.

To evaluate failure probabilities associated with inelastic limit states while considering a full range of uncertainty, various efficient approaches have been developed to rapidly estimate inelastic responses of wind excited systems [2]-[4], enabling direct propagation of uncertainties through the system. The failure probability can then be solved through the integral of Eq. (1) for each limit state using simulation-based methods. In particular, an efficient stochastic simulation scheme based on conditional simulation [8] was developed to address the computational challenges associated with direct Monte Carlo simulation, especially when the reliability index of interest is associated with small failure probabilities, i.e., is in the tail of the distribution, which usually requires very large sample sizes if reasonable accuracy is to be achieved. This conditional simulation scheme is based on partitioning the wind hazard curve into a set of mutually exclusive and collectively exhaustive events, thereby enabling unbiased and high fidelity estimation of small failure probabilities (e.g., 10^{-6}) from small sample sets through the total probability theorem. To further account for extreme wind events from each wind direction, a sector-by-sector based approach, where the failure probability is defined from the most critical sector, was developed within the aforementioned conditional stochastic simulation scheme. Finally, the failure probability can be further transformed into a commonly used reliability measure in terms of the “reliability index”, β , for each limit state of interest based on the assumption of the first-order reliability method, i.e., $\beta = \Phi^{-1}(1 - P_f)$.

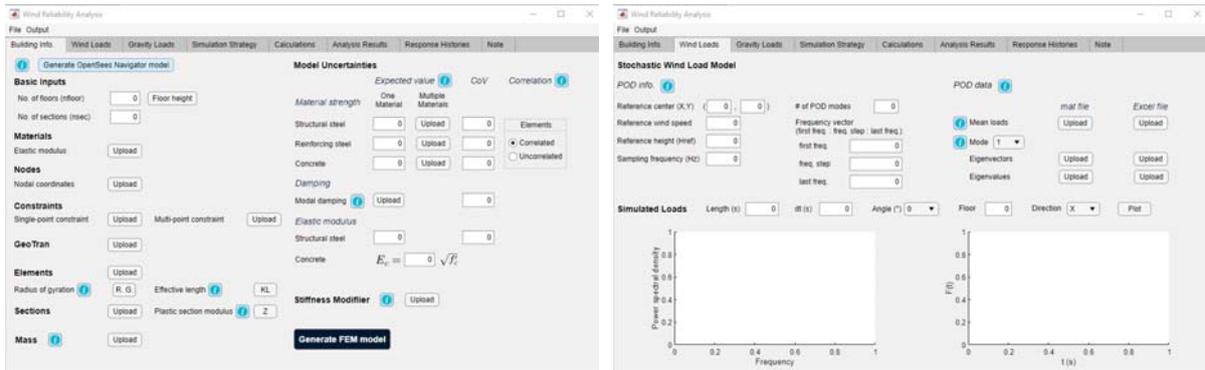
2.2 Software Description

The software is equipped with a GUI where the user can, after introducing all relevant input information, estimate the reliability of wind excited systems against various elastic and inelastic limit states at both component and system levels, as defined in Section 2.1, through stochastic simulation while considering a full range of uncertainties. To carry out the pre- and post-processing for the reliability assessment, the GUI is made up of eight tabs, as shown in Figure 1, where the user can load the model, introduce the necessary information to set up the problem, perform a preliminary gravity check, run the reliability analysis, and visualize and export analysis results. The various input parameters for the problem, ranging from the finite element model, uncertainties, stochastic wind load model, simulation strategy, etc., to the analysis options are all input through the tabs. A detailed description of all menus and tabs are outlined an accompanying manual distributed with the software.

The first step to carrying out reliability analysis using the software is to define all necessary information for the building model. In particular, OpenSees (Open System for Earthquake Engineering Simulation) finite element models can be directly imported into the GUI through various *tcl* files for model generation on the *Building Info.* tab. Furthermore, to account for the uncertainties (record-to-record variability) in the wind loads, the stochastic wind load model is required. In this software, a wind tunnel informed proper orthogonal decomposition (POD) model is adopted for simulating stochastic wind loads. Hence, the user is required to introduce relevant POD data and information into the software on the *Wind Loads* tab. Statistical information for uncertainties associated with the structural system, gravity and wind loads must also be provided for running the reliability analysis.

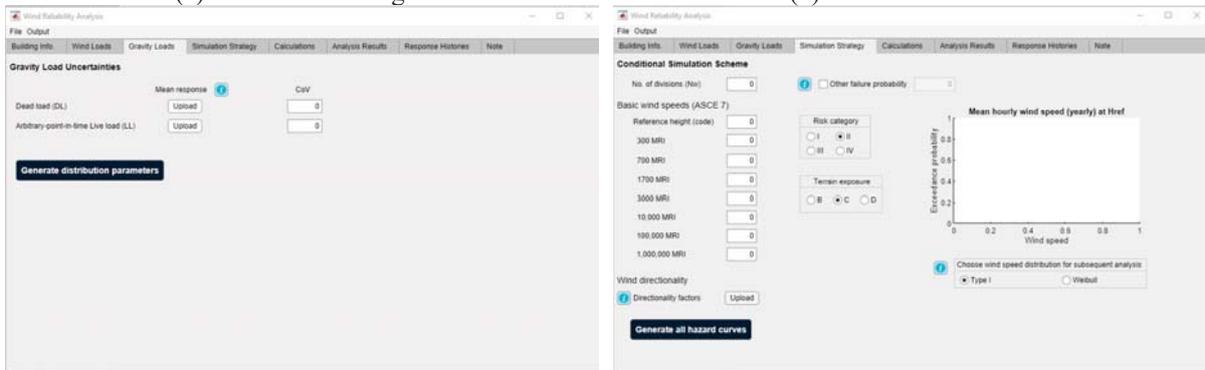
Another key aspect of running reliability analysis through the software is that the user must set up the conditional simulation scheme [1], including the site specific wind hazard curve, for the analysis. Two wind hazard curves will be generated by fitting a Type I or Weibull distributions to a series of basic wind speeds corresponding to various mean recurrence intervals (MRI)

as suggested in ASCE 7 [9] while considering the selected Risk and Terrain Exposure Category [9]. The user then has the option to choose their desired wind speed distribution for subsequent analysis based on the fitted curves displayed on the *Simulation Strategy* tab.



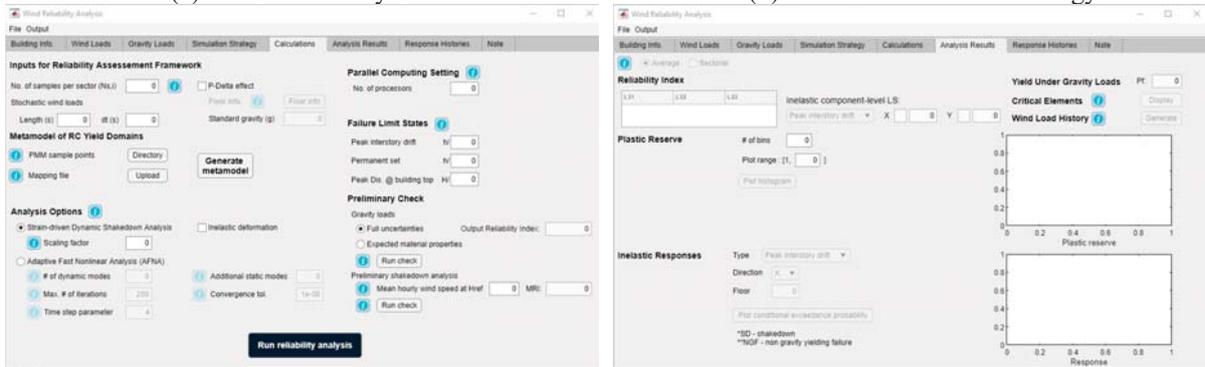
(a) TAB 1: Building Info.

(b) TAB 2: Wind Loads.



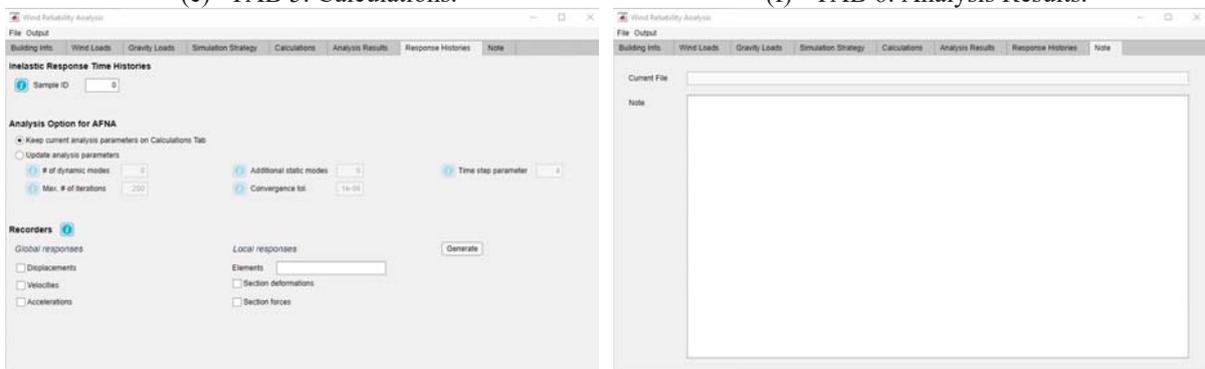
(c) TAB 3: Gravity Loads.

(d) TAB 4: Simulation Strategy.



(e) TAB 5: Calculations.

(f) TAB 6: Analysis Results.



(g) TAB 7: Response Histories.

(h) TAB 8: Notes.

Figure 1: Layouts for all tabs of the software.

Once the problem has been set up, two preliminary checks, namely the gravity check and preliminary shakedown analysis, can be carried out on the *Calculations* tab. Output files summarizing sample information and demand to capacity ratios for all elements will be generated automatically to assist the user in designing or modifying section sizes of the building in order to achieve a target performance.

After the preliminary check, pressing Run reliability analysis button on the *Calculations* tab starts the simulation and a wait bar is created (as a separated window) to update current progress as the analysis moves through each sample. Once the computation is finished, the analysis results are saved automatically and can be displayed on the *Analysis Results* tab. Reliability indexes, critical elements, histogram for plastic reserves and probabilistic distributions for any inelastic response of interest can be plotted selecting the corresponding button on this tab. All outputs on this tab can also be exported from the main menu in *csv* or *xlsx* format. Finally, response time histories can be generated on the *Response Histories* tab for any selected sample of interest using the AFNA approach for further review.

3 CASE STUDY

An example building will be presented in this section to demonstrate the potential of the presented software. All input and files necessary for running the reliability analysis are distributed with the software and described in detail in the user's manual.

3.1 Description

A 45-story archetype building, as shown in Fig. 2(a), is presented to demonstrate the potential of the presented software. The layout consists of 45 levels of office space, floor system composed of 63.5 mm (2.5 in.) of light-weight concrete over 76.2 mm (3 in.) of composite deck supported by steel beams and columns. The columns and braces are wide angle W14 sections except for the corner columns at lower levels, which are square box sections. Each floor is considered to act as a rigid floor diaphragm for horizontal movements. The story height is 4 m for all levels. The overall height of the structure is 180 m. In estimating elastic responses, the first six modes are considered in the modal analysis with damping ratios of 2%. The steel composing the frame is assumed to be elastic perfectly plastic with nominal yield stress $F_y = 345$ MPa. The nominal Young's modulus E_s and shear modulus G_s are taken to be 200 GPa and 77 GPa respectively. Nominal floor loads including self load, superimposed dead load and live load are summarized in Table 1. Uncertainties in the structural system as well as in the gravity and wind loads are considered in the reliability analysis (Table 2).

Self Load	Superimposed Dead Load	Live Load
2.4	0.72	3.1

Table 1: Nominal floor loads of the 45-story archetype building (Units: kN/m²).

	Nominal	Mean/Nominal	Coefficient of Variation	Distribution
F_y	345 (MPa)	1.1	0.06	Normal
E_s	200 (GPa)	1	0.04	Lognormal
ξ	2%	1	0.3	Lognormal
D	-	1.05	0.1	Normal
L_{apt}	-	0.24	0.6	Gamma

Table 2: Description of random variables considered for the 45-story archetype building.

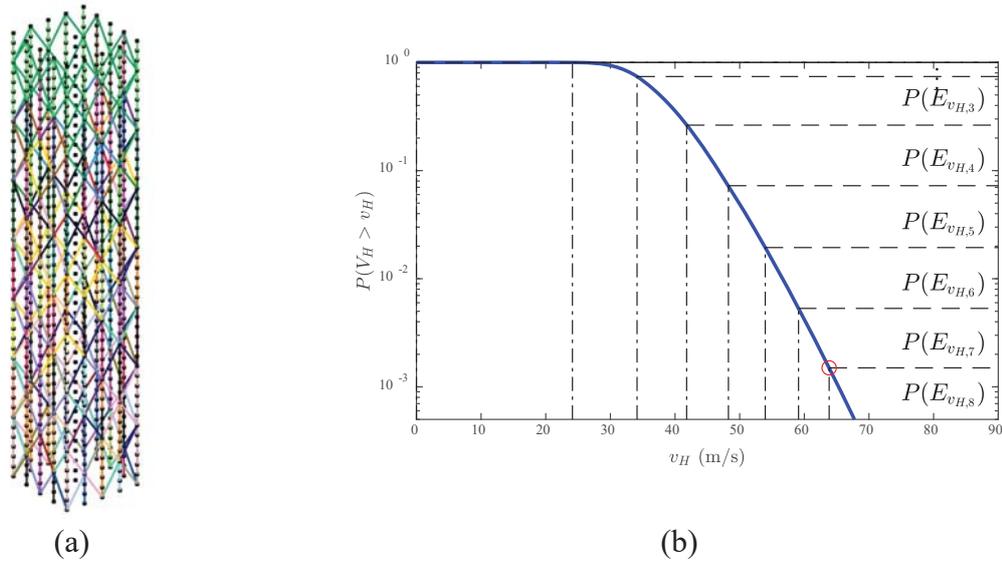


Figure 2: (a) 3D view of the 45-story archetype building and (b) site specific 50-year non-directional mean hourly wind speed at the reference height of the archetype building.

Wind tunnel driven stochastic wind loads of $T = 3600$ s were considered with random wind speeds generated from the site specific hazard curve, as shown in Figure 2(b). In modeling wind directionality, eight sectors, specifically NE, E, SE, S, SW, W, NW and N, were considered in the analysis. To further model the uncertainty in wind direction within each azimuthal sector in the stochastic simulation, the wind direction is assumed in the software to be uniformly distributed between the upper and lower bounds of each azimuthal sector.

3.2 Results

The reliability for the archetype building was determined for four failure limit states outlined in Section 2.1. In particular, peak interstory drift ratio of 1%, peak drift at the building top of 0.5% and permanent set of 0.1% were considered as deformation limits for LS4. The analysis was carried out for a total of 3200 samples, i.e., 400 samples for each wind direction sector. Figure 3 reports the reliability indexes for all limit states in a format that can be exported from the software. By comparing the reliabilities associated with all limit states, it can be observed that this structure is more susceptible to failure due to excessive inelastic deformations, peak displacements at the building top and peak interstory drifts, rather than the inability to shake-down. In addition, figures for plastic reserve of the system and probability distribution of any response parameter of interest can also be plotted on the *Analysis Results* tab, as shown in Figure 4 for the histogram of the plastic reserve of the system and the distribution associated with the peak displacements at the building top in the X-direction.

Limit State	Description	Reliability Index (Average)	Floor (Average)	Reliability Index (Sectorial)	Floor (Sectorial)
LS1	First component yield	= 3.38	-	= 2.84	-
LS2	First system yield	= 3.31	-	= 2.81	-
LS3	Non-shakedown	= 3.90	-	= 3.50	-
LS4	Peak interstory drift-X	= 4.14	43	= 3.63	43
	Peak interstory drift-Y	= 3.97	43	= 3.36	43
	Permanent set-X	= 3.97	11	= 3.52	12
	Permanent set-Y	= 4.23	19	= 3.76	21
	Peak displacement @ Top-X	= 3.46	-	= 2.94	-
	Peak displacement @ Top-Y	= 3.65	-	= 3.03	-

Figure 3: Reliability indexes for the 45-story archetype building exported from the software.

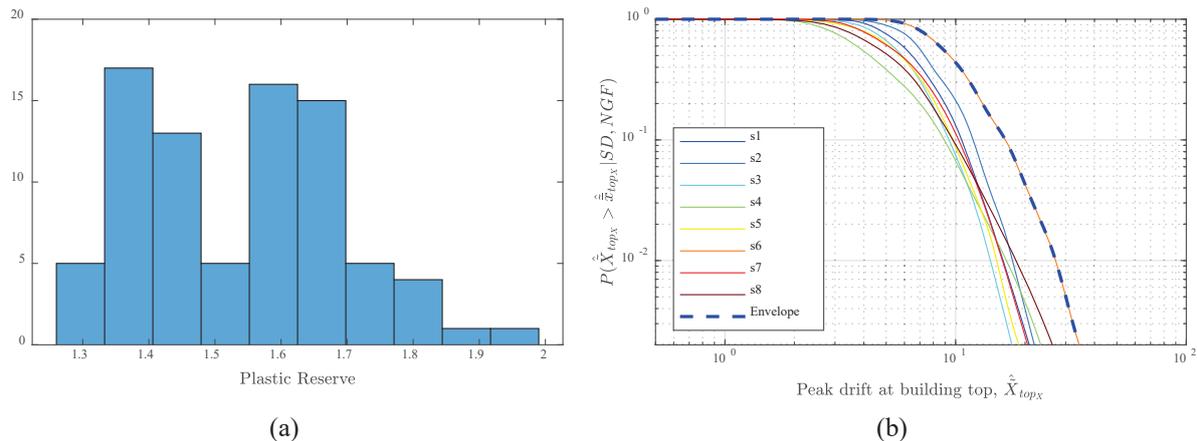


Figure 4: (a) Histogram of plastic reserve and (b) probability of exceedance of the peak displacements at the building top in the X-direction for the 45-story archetype building.

4 CONCLUSIONS

This paper presented a software tool for carrying out reliability analysis of inelastic wind excited systems through direct stochastic simulation with the aim of enabling the full transition from proof-of-concept to practice of reliability estimation considering a full range of uncertainty. The tool is equipped with a GUI that consists in eight tabs that are designed for automatically carrying out all stages of the analysis including: importing the OpenSees finite element model; calibrating the stochastic wind load model; executing the reliability analysis based on efficient conditional stochastic simulation; and finally providing options to display/save the results from the analysis and rerun samples of interest to have a comprehensive understanding of the nonlinear response of the system. The potential of the presented software was demonstrated on a 45-story archetype building subject to stochastic wind loads.

5 ACKNOWLEDGEMENT

This research effort was supported in part by the Magnusson Klemencic Associates (MKA) Foundation through Research Grant Agreement No. 101. This support is gratefully acknowledged.

REFERENCES

- [1] W.-C. Chuang, Innovative Frameworks for the Probabilistic Performance-Based Design of Inelastic Wind Excited Structures, Doctoral dissertation, University of Michigan, 2019.
- [2] W.-C. Chuang, S. M. J. Spence, An efficient framework for the inelastic performance assessment of structural systems subject to stochastic wind loads, *Engineering Structures*, Elsevier, 179, 92–105, 2019.
- [3] W.-C. Chuang, S. M. J. Spence, Probabilistic performance assessment of inelastic wind excited structures within the setting of distributed plasticity, *Structural Safety*, 84, 101923, 2020.
- [4] B. Li, W.-C. Chuang and S. M. J. Spence, An adaptive fast nonlinear analysis (AFNA) algorithm for rapid time history analysis, *8th ECCOMAS Thematic Conference on Computational Methods in Structural Dynamics and Earthquake Engineering*, 2021.

- [5] F. Bartlett, R. Dexter, M. Graeser, J. Jelinek, B. Schmidt, T. Galambos, Updating standard shape material properties database for design and reliability. *Engineering Journal*, 40:2–14, 2003.
- [6] B. Ellingwood, J. G. MacGregor, T. V. Galambos, C. A. Cornell. Probability based load criteria: Load factors and load combinations. *Journal of the Structural Division*, 108:978–997, 1982.
- [7] A. S. Nowak, K. R. Collins. Reliability of Structures. CRC Press, 2013.
- [8] Z. Ouyang, S. M. J. Spence, A performance-based wind engineering framework for envelope systems of engineered buildings subject to directional wind and rain hazards, *Journal of Structural Engineering*, 146(5), 04020049, 2020.
- [9] ASCE 7-16, Minimum design loads and associated criteria for buildings and other structures. American Society of Civil Engineers (ASCE), Reston, VA, 2016.

UNCERTAINTY QUANTIFICATION FOR DEEP LEARNING REGRESSION MODELS IN THE LOW DATA LIMIT

Cristina Garcia-Cardona¹, Yen Ting Lin¹, and Tanmoy Bhattacharya¹

¹Los Alamos National Laboratory
Los Alamos, NM, USA
e-mail: {cgarcia, yentingl, tanmoy}@lanl.gov

Keywords: Uncertainty Quantification, Heteroscedastic Model, Quantile Model.

Abstract. *Deep learning models have contributed to a broad range of applications, but require large amounts of data to learn the desired input-output mapping. Despite the success in developing prediction engines that have high accuracy, much less attention has been given to assessing the error associated with individual predictions. In this work, we study machine-learning models of uncertainty quantification for regression, i.e., methods that are almost purely data driven and use deep learning itself to quantify the confidence in its predictions. We use two approaches, namely the heteroscedastic and quantile formulations, and their extensions to problems with multidimensional output. We focus on the low data limit, where the data sets available are on the order of hundred, not thousands, samples. Through numerical experiments we demonstrate that both heteroscedastic and quantile formulations are robust and good at uncertainty estimation even in this low data limit. We note that the quantile formulation seems to have better performance and is more stable than the heteroscedastic case. Overall, our studies pave the way towards practical design of deep learning models that provide actionable predictions with quantified uncertainty using accessible volumes of data.*

1 INTRODUCTION

Deep learning models have contributed to a broad range of applications including image processing, speech recognition, drug discovery and computational materials science. These models can often vastly accelerate inference, but, being purely data driven with little input about the subject matter, require large amounts of data to learn the desired input-output mapping. Most of the work in the field has been on developing prediction engines that have high accuracy; much less attention has been given to automatically assessing the error associated with individual predictions, but this is no less an essential task when the results are applied in fields such as medicine or engineering [1].

In this work we study machine-learning based models of uncertainty quantification for regression. In contrast with ensemble methods [2], probabilistic machine learning methods [3] or dropout-based approaches [4], that use the mean and variance generated by the dispersion among realizations of models, the strategies that we apply are based on treating the simultaneous prediction of the target and its confidence as a multi-task problem and training the regression models using loss functions that specifically take into account a measure of predictive uncertainty. We specifically evaluate heteroscedastic [5] and quantile [6] formulations and consider their extension to problems with multidimensional output.

The need for uncertainty quantification is especially true in the low data limit, where the density of input points is too low to have replicate measurements in small neighborhoods in feature space, and yet predictions that do not separate the certain from the uncertain are often not actionable! Therefore, in this work, we also study this small data limit. In particular, we study the case where the data sets available are on the order of hundred, not thousands of, samples. Real world data in the biological world are often similarly scarce, due, for example, to the high-cost of experiments, a large number of control parameters, and the high-dimensionality of the poorly-understood feature space, which makes it critical to maximize the predictive value of the trained models.

The document is structured as follows. Sec. 2 introduces the different uncertainty quantification formulations evaluated for regression tasks. Sec. 3 describes the machine learning architectures used. Sec. 4 details the numerical experiments performed, and Sec. 5 finalizes with conclusions drawn from the work.

2 UNCERTAINTY QUANTIFICATION MODELS

We compare two different strategies for deep-learning the uncertainty quantification task, namely the heteroscedastic [5] and quantile [6] formulations. The former models the predictions as normal random variables with parameters that depend on the input. The learning task, then, involves the simultaneous prediction of the mean and the variance of the target output. The quantile formulation, on the other hand, directly learns the various quantile functions for the prediction as a multi-task setting. We also consider extensions of heteroscedastic and quantile formulations for problems with multidimensional output. This section describes both approaches in more detail.

2.1 Heteroscedastic Model

For simplicity, we first consider a heteroscedastic model which regards the one-dimensional observed value as a sample from a univariate normal distribution, with the mean and the variance given by a smooth function of the input features. In the next subsection, we will generalize the method to higher-dimensional observations, for which multivariate Gaussian distributions shall

be adopted. To learn to predict both mean and variance, the machine learning model is trained to minimize the negative log-likelihood (NLL) of the feature-dependent normal distribution [5]. Hence, the heteroscedastic loss over the entire training dataset which consists of N pairs of input \mathbf{x}_i and output y_i , $i = 1 \dots N$, can be expressed as,

$$\mathcal{L}(f, \sigma; \{y^i, \mathbf{x}^i\}_{i=1}^N) = \frac{1}{N} \sum_{i=1}^N \frac{1}{2\sigma(\mathbf{x}^i)^2} \|y^i - f(\mathbf{x}^i)\|^2 + \frac{1}{2} \log \sigma(\mathbf{x}^i)^2, \quad (1)$$

with $f(\mathbf{x})$ and $\sigma(\mathbf{x})^2$ the mean and variance predictions of the model, respectively, and N the samples in the training set. This is similar to the work of Lakshminarayanan et al. [7], but differs in the fact that we do not use ensembles, and also, we guarantee the positivity of the variance by predicting $\log \sigma(\mathbf{x}^i)^2$ as in [5], instead of the softplus function $\log(1 + \exp(\cdot))$ that they use.

Multivariate Approach

For prediction of multiple outputs, a product of one-dimensional normal distributions is not able to capture the correlation in the uncertainty prediction of the different outputs. Therefore, we use the probability density function (PDF) of a multivariate normal distribution to construct the heteroscedastic loss for the multivariate case. The PDF of a multivariate normal distribution can be written as

$$f(\mathbf{z}; \boldsymbol{\mu}, \Sigma) = \frac{\det(\Sigma)^{-1/2}}{2\pi^{k/2}} \exp\left(-\frac{1}{2}(\mathbf{z} - \boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{z} - \boldsymbol{\mu})\right), \quad (2)$$

where $\boldsymbol{\mu} \in \mathbb{R}^k$ stands for the mean, $\Sigma \in \mathbb{R}^{k \times k}$ represents the positive definite covariance matrix, and k is the output space dimension. To learn to predict both the mean and covariance, the machine learning model is trained to minimize the NLL of the multivariate normal PDF, again with parameters chosen as smooth functions on the input domain. The multivariate heteroscedastic loss corresponds to

$$\mathcal{L}(\mathbf{f}, \Sigma; \{\mathbf{y}^i, \mathbf{x}^i\}_{i=1}^N) = \frac{1}{N} \sum_{i=1}^N ((\mathbf{y}^i - \mathbf{f}(\mathbf{x}^i))^T \Sigma(\mathbf{x}^i)^{-1}(\mathbf{y}^i - \mathbf{f}(\mathbf{x}^i)) + \log \det(\Sigma(\mathbf{x}^i))), \quad (3)$$

where $\mathbf{f}(\mathbf{x}) \in \mathbb{R}^k$ stands for the vector of the mean prediction of the model with k outputs, $\Sigma(\mathbf{x}) \in \mathbb{R}^{k \times k}$ represents the predicted covariance matrix, \mathbf{y} the true mean value, and the constants $(2\pi)^{-k/2}$ and $1/2$ have been omitted. To guarantee that the covariance matrix Σ is positive definite, the learning task is specified such that one learns A such that $\Sigma = A^T A$, which is positive by definition.

In general, we need to be careful to avoid the flat-directions of A , i.e., the changes in A that do not change $A^T A$ making the learning task difficult. In the two-output case, with vector $\mathbf{f}(\mathbf{x}) \in \mathbb{R}^2$ and matrix $A(\mathbf{x}) \in \mathbb{R}^{2 \times 2}$, which will be the focus of our study, this problem is easily solved. For simplicity, the explicit \mathbf{x} dependence of A is (mostly) omitted in the following description. Since the 2×2 matrix A can be represented in terms of scalar components a, b, c, d , as

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix},$$

the corresponding covariance matrix can be written as

$$\Sigma = A^T A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}^T \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} a^2 + c^2 & ab + cd \\ ab + cd & b^2 + d^2 \end{pmatrix} \triangleq \begin{pmatrix} \sigma_{11}^2 & \sigma_{12} \\ \sigma_{12} & \sigma_{22}^2 \end{pmatrix}. \quad (4)$$

Note that any simultaneous rotation by the same angle of the vectors given by the columns of matrix A leaves the covariance matrix unchanged. We choose rotation angle $\theta = \arctan(c - b)/(a + d)$, to force $b = c$. With this convention, we can learn the three unconstrained parameters $a, b \equiv c$, and d instead of the three parameters σ_{ij} that need to be constrained to obtain a positive matrix Σ . The remaining freedom in the choice of the parameters turns out to be discrete sign freedoms of the two rows that do not pose difficulties in learning. With Σ in hand, the likelihood function can be explicitly calculated by computing the inverse of the 2×2 covariance matrix analytically,

$$\Sigma^{-1} = \frac{1}{\det(\Sigma)} \begin{pmatrix} \sigma_{22}^2 & -\sigma_{12} \\ -\sigma_{12} & \sigma_{11}^2 \end{pmatrix}, \quad (5)$$

with $\det(\Sigma) = \sigma_{11}^2 \sigma_{22}^2 - \sigma_{12}^2 \neq 0$. Defining: $\mathbf{e} = (e_1, e_2)^T = \mathbf{y} - \mathbf{f}(\mathbf{x})$, allows to write

$$\begin{aligned} \text{NLL} &= \frac{1}{\det(\Sigma)} \begin{pmatrix} e_1 \\ e_2 \end{pmatrix}^T \begin{pmatrix} \sigma_{22}^2 & -\sigma_{12} \\ -\sigma_{12} & \sigma_{11}^2 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \end{pmatrix} + \log \det(\Sigma) \\ &= \frac{1}{\det(\Sigma)} (\sigma_{22}^2 e_1^2 - 2 \sigma_{12} e_1 e_2 + \sigma_{11}^2 e_2^2) + \log \det(\Sigma). \end{aligned} \quad (6)$$

Thus, the loss function for a heteroscedastic approach with two outputs can be written in a simplified form as

$$\begin{aligned} \mathcal{L}(\mathbf{f}, \Sigma; \{\mathbf{y}^i, \mathbf{x}^i\}_{i=1}^N) &= \frac{1}{N} \sum_{i=1}^N \left(\frac{1}{\det(\Sigma(\mathbf{x}^i))} (\sigma_{22}(\mathbf{x}^i))^2 (y_1^i - f_1(\mathbf{x}^i))^2 \right. \\ &\quad - 2 \sigma_{12}(\mathbf{x}^i) (y_1^i - f_1(\mathbf{x}^i)) (y_2^i - f_2(\mathbf{x}^i)) \\ &\quad \left. + \sigma_{11}(\mathbf{x}^i)^2 (y_2^i - f_2(\mathbf{x}^i))^2 + \log \det(\Sigma(\mathbf{x}^i)) \right). \end{aligned} \quad (7)$$

2.2 Quantile Model

The quantile model does not make any assumption about the underlying distribution of the samples. Thus, instead of predicting mean and variance of a normal distribution, the predictions are directly of the various quantiles. A quantile is a set of values that divides a frequency distribution of a variable into equal groups, each containing the same fraction of the whole variable range. To learn a quantile map with a machine learning model, the model is trained to minimize the quantile loss for any given quantile $\alpha \in (0, 1)$. The quantile loss for an individual sample \mathbf{x}_i is defined as [8]

$$\mathcal{L}_\alpha(\xi^i) = \begin{cases} \alpha \xi^i & \text{if } \xi^i \geq 0, \\ (\alpha - 1) \xi^i & \text{if } \xi^i < 0. \end{cases}, \quad (8)$$

where $\xi^i = y^i - f^\alpha(\mathbf{x}^i)$, and, y^i and $f^\alpha(\mathbf{x}^i)$, correspond to the observed value and the predicted quantile α , respectively.

In order to learn to predict several quantiles simultaneously, a loss function composed by the sum over the individual quantile losses of the entire data set can be formulated as follows

$$\mathcal{L}(\{f\}^\alpha; \{y^i, \mathbf{x}^i\}_{i=1}^N) = \sum_{\alpha} \frac{1}{N} \sum_{i=1}^N \mathcal{L}_{\alpha}(y^i - f^{\alpha}(\mathbf{x}^i)) . \quad (9)$$

Specifically, we devise our model to predict three quantiles: the 1st, 5th, and 9th deciles, corresponding to $\alpha = 0.1, 0.5$ and 0.9 , respectively. Note that the quantile for $\alpha = 0.5$ is the median of the distribution.

Multivariate Approach

Extending the quantile formulation to prediction with multiple outputs is not as direct as in the heteroscedastic case, since there is no underlying assumption of the form of the sample distribution, nor a basis for imposing an ordering in multivariate observations as is in the scalar one-output case. Among possible multivariate quantile extensions, we use the generalization given by the geometric quantile formulation proposed in [9]. In the geometric quantile formulation, the d -dimensional multivariate quantiles are indexed by elements of the open unit ball $B^{(d)} = \{\mathbf{u} | \mathbf{u} \in \mathbb{R}^d, |\mathbf{u}| < 1\}$. A function $\Phi(\mathbf{u}, \mathbf{t}) = |\mathbf{t}| + \langle \mathbf{u}, \mathbf{t} \rangle$ is defined for any element $\mathbf{u} \in B^{(d)}$ and $\mathbf{t} \in \mathbb{R}^d$, with $\langle \cdot, \cdot \rangle$ denoting the usual Euclidean inner product. The geometric quantile $\hat{\mathbf{Q}}_n(\mathbf{u})$ corresponding to ('index') \mathbf{u} and based on d -dimensional data points $\mathbf{X}^1, \mathbf{X}^2, \dots, \mathbf{X}^n$ is defined as

$$\hat{\mathbf{Q}}_n(\mathbf{u}) = \arg \min_{\mathbf{Q} \in \mathbb{R}^d} \sum_{i=1}^n \Phi(\mathbf{u}, \mathbf{X}^i - \mathbf{Q}) . \quad (10)$$

Note that for $\mathbf{u} = \mathbf{0}$, the quantile $\hat{\mathbf{Q}}_n(\mathbf{0})$ corresponds to the spatial median. Also, note that a \mathbf{u} with $|\mathbf{u}|$ close to 1 corresponds to an extreme quantile, while close to 0 corresponds to a central quantile. In general, the magnitude of $|\mathbf{u}|$ measures the extent of deviation of $\hat{\mathbf{Q}}_n(\mathbf{u})$ with respect to the center of the cloud formed by the $\{\mathbf{X}^i\}_{i=1}^n$ points, while the direction of \mathbf{u} can be interpreted as providing a notion of how 'out' a point is in a given direction with respect to the center of the cloud, considering the geometry of the cloud itself. Further details can be found in [9].

In our case, we restrict ourselves to directions of $\mathbf{u} = \mathbf{1}$, i.e., the vector with unit components, while adopting the proper normalization to make it an element of $B^{(d)}$. Overloading the alpha notation, to keep a knob $\alpha \in (0, 1)$, we define the following geometric multivariate quantile loss

$$\mathcal{L}_{\alpha}(\boldsymbol{\xi}^i) = \Phi\left(\frac{\mathbf{1}}{\sqrt{d}}(2\alpha - 1), \boldsymbol{\xi}^i\right) , \quad (11)$$

where $\boldsymbol{\xi}^i = \mathbf{y}^i - \mathbf{f}^{\alpha}(\mathbf{x}^i)$, $\mathbf{f}^{\alpha}(\mathbf{x}^i)$ is the predicted multivariate quantile model, \mathbf{y}^i is the observed value and d is the output dimension. Note that $\alpha = 0.5$ corresponds to the spatial median, while for $\alpha < 0.5$ this quantile formulation really uses the $\mathbf{u} = -\mathbf{1}$ direction. Analogously to the 1D case, we learn simultaneously several geometric multivariate quantiles (in the $\mathbf{u} = \pm \mathbf{1}$ direction), by minimizing

$$\mathcal{L}(\{f\}^\alpha; \{\mathbf{y}^i, \mathbf{x}^i\}_{i=1}^N) = \sum_{\alpha} \frac{1}{N} \sum_{i=1}^N \mathcal{L}_{\alpha}(\mathbf{y}^i - \mathbf{f}^{\alpha}(\mathbf{x}^i)) . \quad (12)$$

With this setting, we again use $\alpha = 0.1, 0.5$ and 0.9 .

3 MACHINE LEARNING MODELS

To instantiate the uncertainty quantification formulations discussed and estimate \mathbf{f} , \mathbf{f}^α and Σ , we construct different machine learning architectures and train them for regression tasks. Specifically, we build multi-layer feed-forward neural networks with different number of layers and train them using the loss functions described for heteroscedastic and quantile formulations.

The multi-layer feed-forward neural network that we train is composed of neurons with dense connections. The output o_ν^λ of each artificial neuron ν in layer λ is computed as

$$o_\nu^\lambda = h(\mathbf{w}_\nu^\lambda \cdot \boldsymbol{\xi}^\lambda + b_\nu^\lambda), \quad (13)$$

where $\boldsymbol{\xi}^\lambda$ represents the input vector at layer λ ; \mathbf{w}_ν^λ and b_ν^λ represent neuron parameters: weight vector and bias, respectively; the operator \cdot denotes a dot product; and h , the activation function. We use a rectified linear unit (ReLU): $\text{ReLU}(\beta) = \max(0, \beta)$ as the activation function, except in the output layer where we use a linear activation.

We keep the same architecture in all cases to assess the impact of the loss function. We adapt the number of model outputs according to the uncertainty quantification to evaluate:

- 1D Heteroscedastic: prediction of mean response $f(\mathbf{x})$ and variance $\sigma(\mathbf{x})^2$.
- 2D Heteroscedastic: prediction of mean response $\mathbf{f}(\mathbf{x})$ and covariance matrix $\Sigma(\mathbf{x})$.
- 1D Quantile: prediction of $\alpha = 0.1, 0.5, 0.9$ quantiles, $f^{0.1}(\mathbf{x})$, $f^{0.5}(\mathbf{x})$, $f^{0.9}(\mathbf{x})$, respectively.
- 2D Quantile: prediction of $\mathbf{u} = \pm 1$ $\alpha = 0.1, 0.5, 0.9$ multivariate quantiles, $\mathbf{f}^{0.1}(\mathbf{x})$, $\mathbf{f}^{0.5}(\mathbf{x})$, $\mathbf{f}^{0.9}(\mathbf{x})$, respectively.

Specifically, a heteroscedastic model for one-variable prediction has two outputs: $f(\mathbf{x}^i)$ and $\sigma(\mathbf{x}^i)$, while a heteroscedastic model for two-variable prediction has five outputs: $f_1(\mathbf{x}^i)$, $f_2(\mathbf{x}^i)$, $\sigma_{11}(\mathbf{x}^i)$, $\sigma_{12}(\mathbf{x}^i)$ and $\sigma_{22}(\mathbf{x}^i)$. Analogously, a quantile model for one-variable prediction has three outputs, since we chose to predict three quantiles: $f^{0.5}(\mathbf{x}^i)$, $f^{0.1}(\mathbf{x}^i)$ and $f^{0.9}(\mathbf{x}^i)$. A quantile model for two-variable prediction has six outputs, since we chose to predict three quantiles: $f_1^{0.5}(\mathbf{x}^i)$, $f_2^{0.5}(\mathbf{x}^i)$, $f_1^{0.1}(\mathbf{x}^i)$, $f_2^{0.1}(\mathbf{x}^i)$, $f_1^{0.9}(\mathbf{x}^i)$ and $f_2^{0.9}(\mathbf{x}^i)$.

Regularization To reduce the possibility of model overfitting we apply two well known techniques. We use dropout [10] after each dense layer to randomly drop a fraction of the layer's neurons and avoid co-adaptation. We also make use of the Tikhonov regularization by minimizing the ℓ_2 norm of the weight vectors \mathbf{w}_ν^λ associated to the different neurons in the model.

4 NUMERICAL EXPERIMENTS

We compare the heteroscedastic and quantile formulations in terms of predictive uncertainty for synthetic data, where we know the ground truth, using regression prediction tasks with one-output (1D case) and two-outputs (2D case). We assess the effect of model complexity by training machine learning models with different numbers of layers. To measure the performance we use the following metrics: mean squared error (MSE) eq. (14), the coefficient of determination (R^2) eq. (15) and the negative log-likelihood (NLL) eq. (16).

$$\text{MSE}(y, f) = \frac{1}{N} \sum_{i=1}^N \|y^i - f(\mathbf{x}^i)\|^2, \quad (14)$$

$$R^2(y, f) = 1 - \frac{\sum_{i=1}^N (y^i - f(\mathbf{x}^i))^2}{\sum_{i=1}^N (y^i - \bar{y})^2}, \quad \bar{y} = \frac{1}{N} \sum_{i=1}^N y^i, \quad (15)$$

$$\text{NLL}(y, f, \sigma) = \frac{1}{N} \sum_{i=1}^N \frac{1}{2\sigma(\mathbf{x}^i)^2} \|y^i - f(\mathbf{x}^i)\|^2 + \frac{1}{2} \log \sigma(\mathbf{x}^i)^2. \quad (16)$$

Note that these expressions refer to the 1D case but analogous expressions apply to the 2D case for MSE and NLL. For computing R^2 in the 2D case, the outputs are concatenated in a 1D vector. Note also that for the quantile model the prediction $f(\mathbf{x}^i)$ used is the median, i.e., the 5th decile: $f^{0.5}(\mathbf{x}^i)$. Additionally, note that the NLL expression is valid only for the heteroscedastic model. For measuring NLL of a quantile model, in 1D, we use a normal approximation where we suppose that the interdecile range (IDR), equivalent to the difference between 9th and 1st deciles (i.e., 80% of the samples), corresponds to 80% of the samples of a normal distribution. In a normal distribution this percentage is contained in a band of 1.28σ radius from the mean. Hence we use $\sigma(\mathbf{x}^i)_q = \text{IDR}(\mathbf{x}^i)/2.56$. We do not estimate the likelihood for the 2D quantile model.

Moreover, in order to have a metric that is sensitive to predictive uncertainty but that is independent of the hypothesized sample distribution, we use the 80% coverage. This is estimated in the 1D heteroscedastic case by computing the fraction of samples falling inside the band of 1.28σ radius from the mean. In the 2D heteroscedastic case, by computing the fraction of samples falling inside the ellipsoid with isolevel equal to $-2 \log(0.2)$. In the 1D quantile case, by computing the fraction of samples falling inside the interval between 1st and 9th deciles. And in the 2D quantile case, by computing the fraction of samples falling inside the square defined by using multivariate quantiles $\alpha = 0.1$ and $\alpha = 0.9$ as corners.

4.1 Synthetic Data Generation

Synthetic data has been constructed using the Hill model,

$$H(x; r_1, r_2, k, h) = \begin{cases} r_1 + (r_2 - r_1) \frac{k^h}{k^h + x^h} & \text{for } h \geq 0, \\ r_1 + (r_2 - r_1) \frac{k^{-h}}{k^{-h} + x^{-h}} & \text{for } h < 0, \end{cases} \quad (17)$$

with parameters h and $r_1, r_2, k > 0$.

For the 1D case, i.e., the one-output case, the Hill model with parameters $r_1 = 10$, $r_2 = 30$, $k = 10$ and $h = -6$ is used as the base function, $y = H(x)$. A 1D Gaussian noise is added to it. The Gaussian noise has mean zero and standard deviation given by another Hill model with parameters $r_1 = 5.48$, $r_2 = 3.16$, $k = 10$ and $h = -6$. In this way we simulate a heteroscedastic, i.e., feature-dependent, noise model. This data is plotted in Figure 1 (Left). It can be seen that it corresponds to a monotonically decreasing 1D signal with relatively high dispersion and that the dispersion moderately increases for higher x -coordinate values. Note that the MSE of the noisy data with respect to the clean data is $\text{MSE}=19.03$.

For the 2D case, i.e., the two-outputs case, a Hill model with parameters $r_1 = 10$, $r_2 = 30$, $k = 10$ and $h = 5$ is used as base function for $y_1 = H(x)$ and $y_2 = H(x + 1)$. A 2D

Gaussian noise is added to the outputs. The Gaussian noise has mean zero and covariance matrix $\Sigma = A^T A$ with

$$A(x) = \begin{pmatrix} 5 \times 10^{-3} x^2 & 2 \times 10^{-3} x \\ 2 \times 10^{-3} x & 8 \times 10^{-3} x^2 \end{pmatrix}.$$

This data is plotted in Figure 1 (Right). It can be seen that it corresponds to a monotonically increasing 2D signal with relatively low dispersion and that the dispersion increases for higher x -coordinate values. Note that the MSE of the noisy data with respect to the clean data is MSE=0.19.

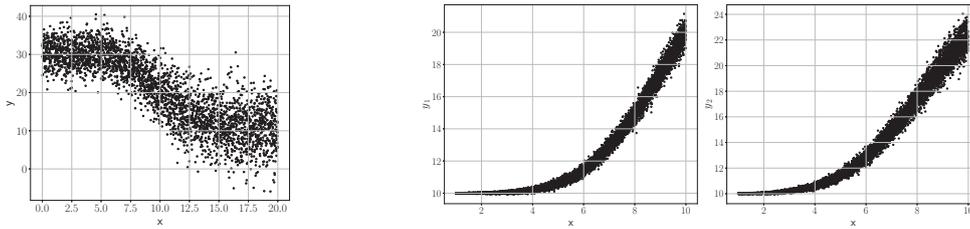


Figure 1: Synthetic data generated for model training. **Left: 1D, N=3000. Right: 2D, N=10000.**

4.2 Results

For the numerical experiments, we build sets of 3000 samples for the 1D case and 10000 samples for the 2D case. For the 1D case, we construct uncertainty estimator models with 50 neurons per layer, and, 2, 3, 5 and 10 layers. For the 2D case, we construct uncertainty estimator models with 100 neurons per layer and, 1, 2, 3, 5 and 10 layers. We use low and high levels of regularization. These correspond to: dropout=0.01 and Reg $\ell_2 = 1 \times 10^{-8}$ (i.e., the factor used for weighting the regularization term given by the ℓ_2 norm of the network weight parameters), for low regularization and dropout=0.2 and Reg $\ell_2 = 1 \times 10^{-5}$ for high regularization. We split the data in 80% training and 20% testing. We train these models with the Adam optimizer [11] during 500 epochs for 1D models and 1000 epochs for 2D models, since those are the approximate number of epochs for stabilized loss function. All the models are implemented in Keras [12].

Box plots for the 1D results obtained for training with N=2400 samples evaluated over the test set (600 samples) for 20 repetitions are shown in Figure 2. Note that both type of models exhibit good performance not only with respect to the predicted value (MSE comparable to training data, median $R^2 > 0.75$) but also in terms of the uncertainty estimations, with similar NLL and coverage close to the expected 80%. Overall, metrics are slightly better for the quantile model, even when the data is a heteroscedastic noisy data. Interestingly, note that there is no significant change in performance for the different number of layers evaluated (2, 3, 5 and 10 layers) or for the low and high regularization levels used.

Box plots for the 2D results obtained for training with N=8000 samples evaluated over the test set (2000 samples) for 20 repetitions are shown in Figure 3. Note that both type of models exhibit good predictive accuracy (MSE comparable to training data, median $R^2 > 0.9$) and the uncertainty estimations are close to the 80% coverage specified, with slight under-prediction for the heteroscedastic case and the quantile with low regularization and small over-prediction

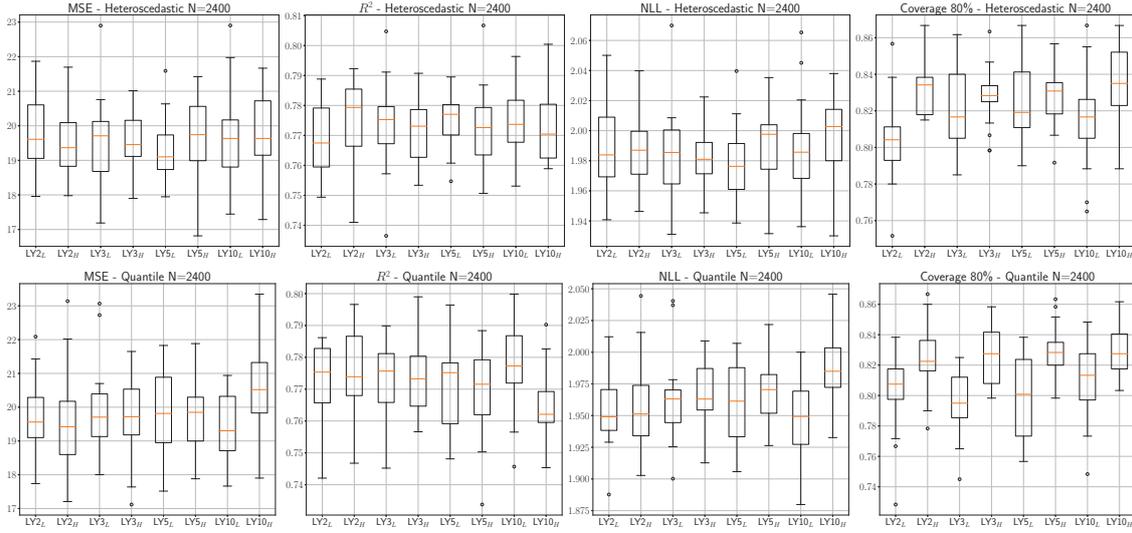


Figure 2: Statistics for models with low and high regularizations, for 1D data and $N=2400$ samples. LY denotes the number of layers, while the L or H sub-index is used to denote regularization level. **Top: Heteroscedastic. Bottom: Quantile.**

for the quantile with high regularization. Additionally, performance is similar for models between 1 and 3 layers, with performance degrading a little for models with 5 layers and high regularization or 10 layers.

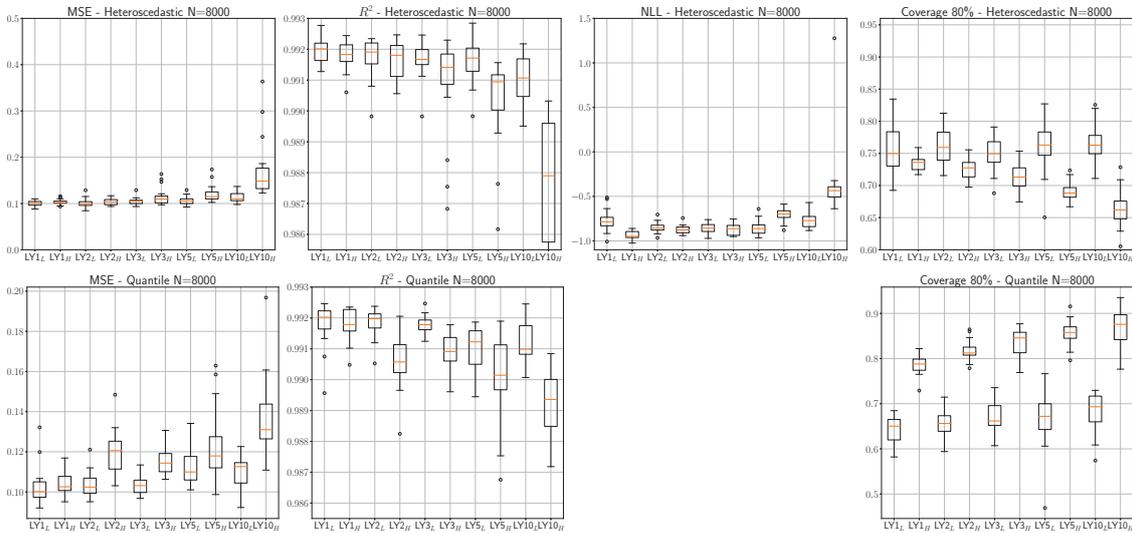


Figure 3: Statistics for models with low and high regularizations, for 2D data and $N=8000$ samples. LY denotes the number of layers, while the L or H sub-index is used to denote regularization level. **Top: Heteroscedastic. Bottom: Quantile.**

4.2.1 Low Data Limit in 1D

For evaluating the consistency between models trained with enough data and models trained in the low data limit, we repeat the training procedure but using $N = 30$ samples for training.

Note that in this case we let models with 10 layers to train during 1000 epochs.

Box plots for the 1D results obtained for training with $N = 30$ samples evaluated over the test set (2970 samples) for 20 repetitions are shown in Figure 4. MSE and R^2 for the heteroscedastic model have been framed in ranges comparable to quantile results so some boxes for the 2 layers neural network are cut. Note that despite the substantial reduction of the data for training, the model performance only narrowly degrades with respect to the case where enough samples are available. Also, some differences between the performance of the two main uncertainty quantification paradigms start to emerge, as well as some more notorious deviations with respect to the complexity of the architecture of the network or the regularization level.

The performance of the quantile formulation degrades less than the heteroscedastic one, with lower MSEs and NLLs, higher R^2 and coverage closer to the 80% expected value. Also, the quantile performance is consistent for 2 to 5 layers deteriorating for the 10-layers network with low regularization (labeled LY10_L in the plots). In general, the regularization seems to play a minor role, with little differences in the 80% coverage prediction between low and high regularization. In contrast, the heteroscedastic model with 2 layers has poor performance. Between 3 to 5 layers, low regularization seems moderately better than high regularization, but the opposite seems to apply for 10 layers. These results seem to indicate that the quantile formulation is more robust for the low data limit, requiring only marginal regularization, unless the network has a high complexity, in which case stronger regularization helps. The heteroscedastic formulation needs an architecture that has medium complexity to be effective in the low data limit, with slightly better performance for low regularization.

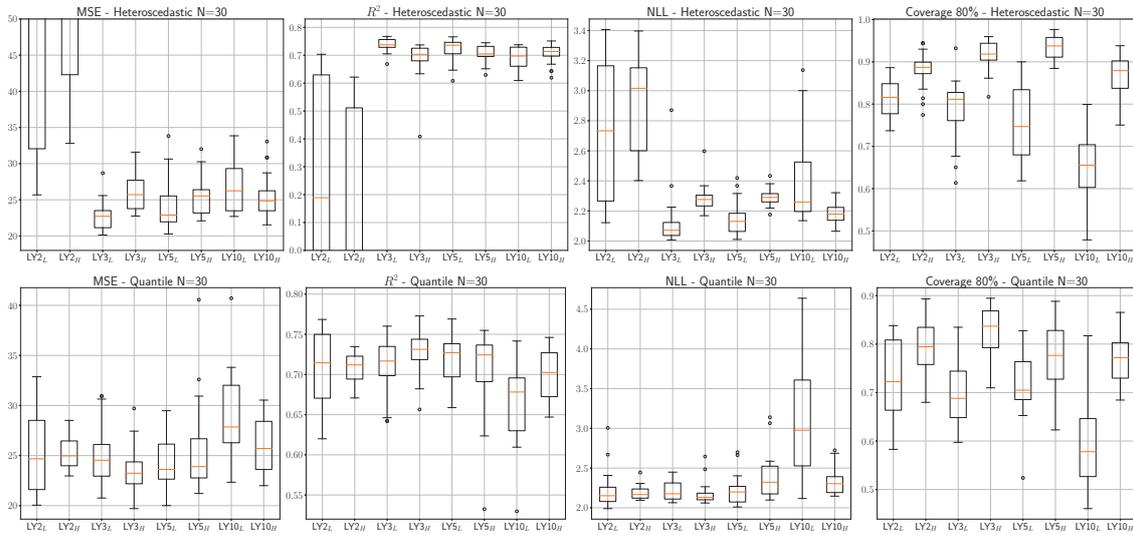


Figure 4: Statistics for models with low and high regularizations, for 1D data and $N=30$ samples. LY denotes the number of layers, while the L or H sub-index is used to denote regularization level. **Top: Heteroscedastic. Bottom: Quantile.**

Additionally, we include results in the low data limit for models trained to minimize the MSE loss, i.e., without considering any uncertainty quantification formalism. Box plots for the 1D results obtained for training with $N = 30$ samples evaluated over the test set (2970 samples) for 20 repetitions are shown in Figure 5. The homoscedastic label denotes that no specific noise model is used for training. Note again that the degradation observed for heteroscedastic and quantile formulations, is in the order of the one for the homoscedastic case, in the small data

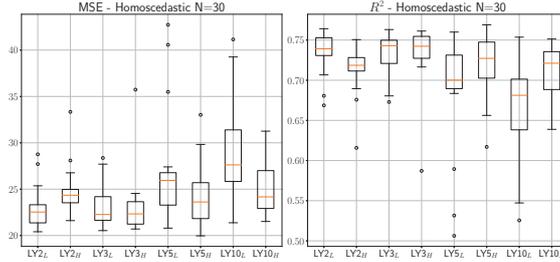


Figure 5: Statistics for models with low and high regularizations, for 1D data and $N=30$ samples, trained without uncertainty quantification. LY denotes the number of layers, while the L or H sub-index is used to denote regularization level.

limit. Therefore, adding uncertainty quantification does not sacrifice model performance, even in the low data limit. On the contrary, it provides a complementary level of information and a small reduction of the model dispersion, specially for cases where the MSE loss has worse performance.

Nevertheless, in the low data limit it makes more sense to also assume that the amount of data for testing is low. Hence, a more practical comparison seems to be to also evaluate model performance in a small testing set. We evaluate this criterion in the models with 3 layers. In this comparison, we randomly select 40 samples and split them randomly into two subset: $N = 30$ samples for training and the remainder 10 samples for testing. We sweep over a logarithmic 10×10 grid of the regularization parameters. For dropout in the range: $[1 \times 10^{-3}, 2 \times 10^{-1}]$ and for Reg ℓ_2 in the range $[1 \times 10^{-8}, 1 \times 10^{-4}]$. For each of the different regularization combinations in the grid, we train a model for 500 epochs and evaluate the performance with respect to all the metrics in the testing set of 10 samples. We compute 20 repetitions over the grid sweeping and report the test median in Figure 6. Not all the metrics are completely consistent with each other, but it seems that the performance for the heteroscedastic model is slightly better on the lower left corner (i.e., very low dropout and ℓ_2 regularization), while there does not seem to be a clear trend in the quantile model.

4.2.2 Low Data Limit in 2D

We proceed as in the 1D case and train models for the 2D data in the low data limit using $N = 60$ and low and high regularizations. Note than in this case the heteroscedastic model requires more iterations to achieve reasonable performance (i.e., about 10000 epochs). Likewise, we train the highly regularized quantile model for 3000 epochs due to slight oscillations. Box plots for the 2D results obtained for training with $N = 60$ samples evaluated over the test set (9940 samples) for 20 repetitions are shown in Figure 7. Overall the performance of the low data limit models is consistent with the models trained with enough samples. The main differences can be found in the 80% coverage results, with both formulations exhibiting better performance for models with low regularization. As a reference, box plot results for training with MSE loss are included in Figure 8. Again, uncertainty quantification formulations, do not degrade performance while slightly reducing model dispersion and providing a measure of the confidence in the machine learning model predictions.

Analogously to the 1D case, we also evaluate the model performance in a small testing set using models with 3 layers. Therefore, we randomly select 80 samples and split them randomly into two subset: $N = 60$ samples for training and the remainder 20 samples for testing. We

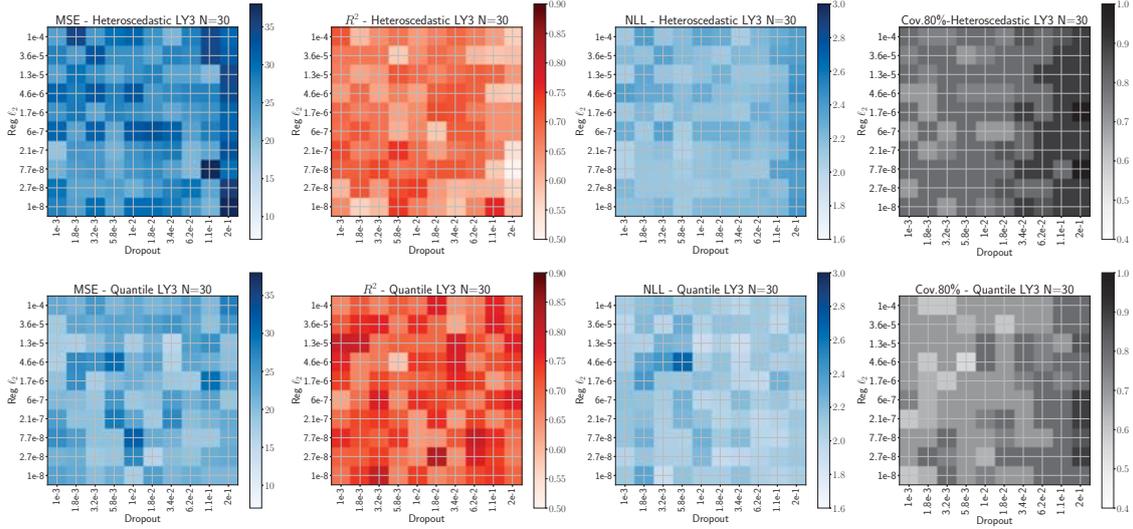


Figure 6: Statistics for models with 3 layers over a grid of different dropouts and ℓ_2 norm regularizations for 1D data and $N=30$ samples. When blue colormap is used, lower values correspond to better performance; when red colormap is used, higher values correspond to better performance; when gray colormap is used, values closer to 0.8 correspond to better performance. **Top: Heteroscedastic. Bottom: Quantile.** Same scale used in both cases.

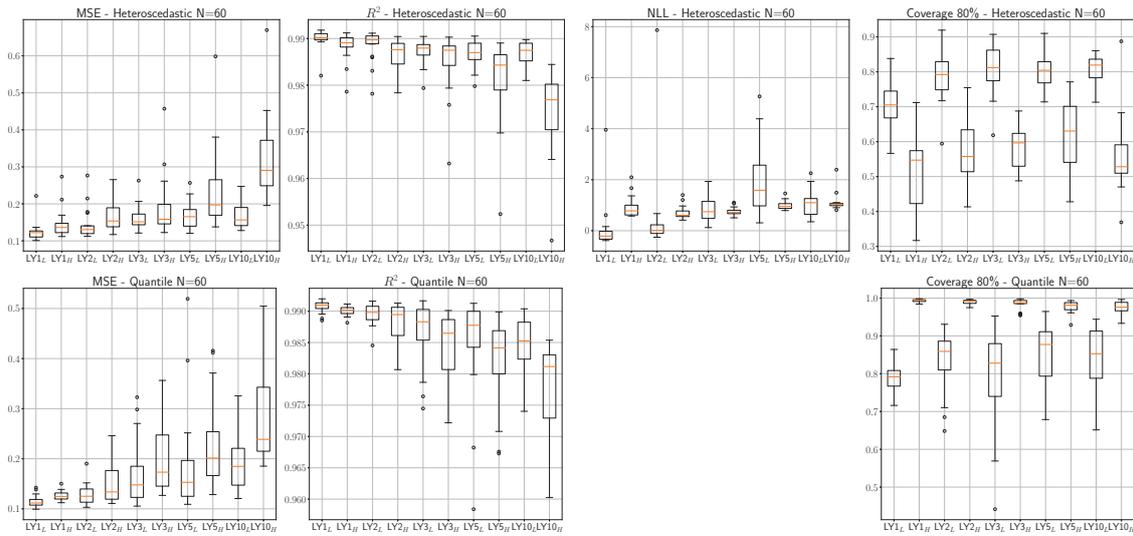


Figure 7: Statistics for models with low and high regularizations, for 2D data and $N=60$ samples. LY denotes the number of layers, while the L or H sub-index is used to denote regularization level. **Top: Heteroscedastic. Bottom: Quantile.**

sweep over the same 10×10 logarithmic grid of the regularization parameters than in the 1D case. For each of the different regularization combinations in the grid, we train a model for 5000 epochs for the heteroscedastic case and 1000 epochs for the quantile case, and evaluate the performance with respect to all the metrics in the testing set of 20 samples. We compute 20 repetitions over the grid sweeping and report the test median in Figure 9. It seems clear that in both formulations better results are achieved for small dropout regularization.

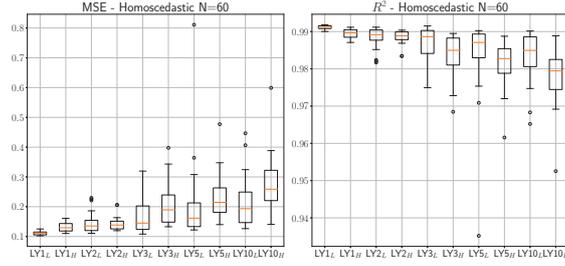


Figure 8: Statistics for models with low and high regularizations, for 2D data and $N=60$ samples, trained without uncertainty quantification. LY denotes the number of layers, while the L or H sub-index is used to denote regularization level.

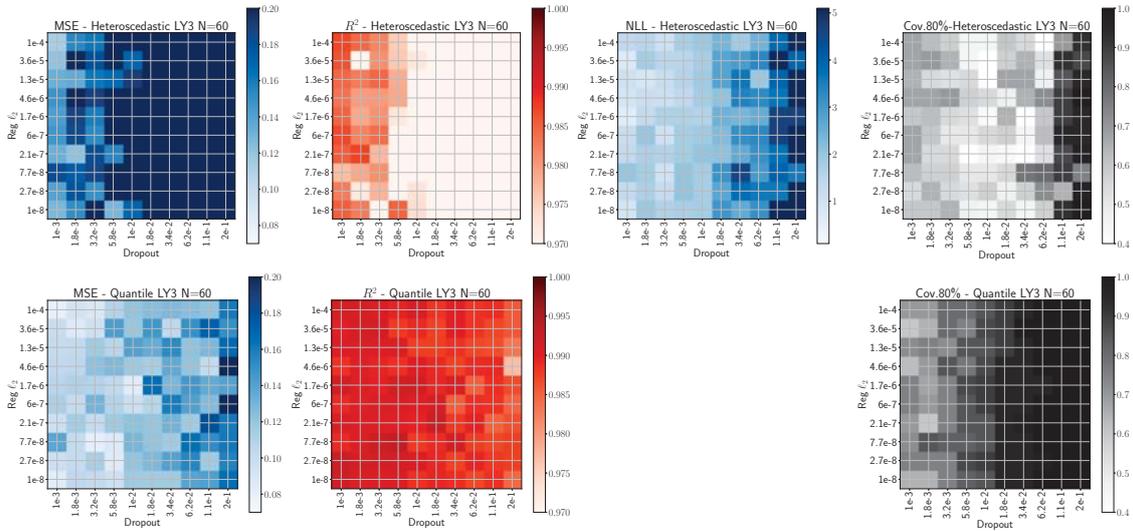


Figure 9: Statistics for models with 3 layers over a grid of different dropouts and ℓ_2 norm regularizations for 2D data and $N=60$ samples. When blue colormap is used, lower values correspond to better performance; when red colormap is used, higher values correspond to better performance; when gray colormap is used, values closer to 0.8 correspond to better performance. **Top: Heteroscedastic. Bottom: Quantile.** Same scale used in both cases, losing some resolution.

5 CONCLUSIONS

In this work, we applied uncertainty quantification models, namely heteroscedastic and quantile formulations, to synthetic data with one and two-outputs. We trained neural-network models with different complexities and evaluated their performance in the low data limit, where just a handful of data (tens of samples) is available. Through numerical experiments we demonstrate that both heteroscedastic and quantile formulations are robust and good at uncertainty estimation even in the low data limit. Furthermore, we find that in these formulations, very small ‘true regularization’ strategies, such as dropout or weighting of the ℓ_2 -norm of the parameters, are required to produce good results even for complex models. Also, we note that the quantile formulation seems to have better performance and is more stable than the heteroscedastic case, i.e., the quantile models do not degrade as fast when model complexity is increased. Overall, our studies pave the way towards practical design of deep learning models that provide actionable predictions with quantified uncertainty using accessible volumes of data.

ACKNOWLEDGEMENTS

This work has been supported in part by the Joint Design of Advanced Computing Solutions for Cancer (JDACS4C) program established by the U.S. Department of Energy (DOE) and the National Cancer Institute (NCI) of the National Institutes of Health, and was performed under the auspices of the U.S. Department of Energy by Los Alamos National Laboratory under Contract DE-AC5206NA25396. YTL acknowledges partial support from LDRD (Laboratory Directed Research and Development) program under project 20210043DR (Uncertainty Quantification for Robust Machine Learning). Approved for public release LA-UR-21-22482.

REFERENCES

- [1] E. Begoli, T. Bhattacharya, and D. Kusnezov, “The need for uncertainty quantification in machine-assisted medical decision making,” *Nature Machine Intelligence*, vol. 1, pp. 20–23, Jan. 2019.
- [2] S. Jain, G. Liu, J. Mueller, and D. Gifford, “Maximizing overall diversity for improved uncertainty estimates in deep ensembles,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, pp. 4264–4271, Apr. 2020. [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/5849>
- [3] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, “Weight uncertainty in neural networks,” in *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ser. ICML’15. JMLR.org, 2015, p. 1613–1622.
- [4] Y. Gal and Z. Ghahramani, “Dropout as a Bayesian approximation: Representing model uncertainty in deep learning,” in *Proceedings of Machine Learning Research*, M. F. Balcan and K. Q. Weinberger, Eds., vol. 48. New York, New York, USA: PMLR, 20–22 Jun 2016, pp. 1050–1059. [Online]. Available: <http://proceedings.mlr.press/v48/gal16.html>
- [5] A. Kendall and Y. Gal, “What uncertainties do we need in Bayesian deep learning for computer vision?” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017, pp. 5574–5584. [Online]. Available: <https://proceedings.neurips.cc/paper/2017/file/2650d6089a6d640c5e85b2b88265dc2b-Paper.pdf>
- [6] S. Abeywardana, “Deep quantile regression,” 2018, <https://towardsdatascience.com/deep-quantile-regression-c85481548b5a>.
- [7] B. Lakshminarayanan, A. Pritzel, and C. Blundell, “Simple and scalable predictive uncertainty estimation using deep ensembles,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017, pp. 6402–6413. [Online]. Available: <https://proceedings.neurips.cc/paper/2017/file/9ef2ed4b7fd2c810847ffa5fa85bce38-Paper.pdf>
- [8] T. S. Ferguson, *Mathematical Statistics: A Decision Theoretic Approach*. New York and London: Academic Press, 1967.
- [9] P. Chaudhuri, “On a geometric notion of quantiles for multivariate data,” *Journal of the American Statistical Association*, vol. 91, no. 434, pp. 862–872, Jun. 1996.

- [10] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.
- [11] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [12] F. Chollet, *Keras documentation*, 2018. [Online]. Available: <https://faroit.com/keras-docs/2.1.2>

UNCECOMP 2021

**Proceedings of the
4th International Conference on
Uncertainty Quantification in Computational Sciences and Engineering**

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

First Edition, September 2021

ISBN: 978-618-85072-6-5



**Institute of Structural Analysis and Antiseismic Research
National Technical University of Athens, Greece**