Universitat Politècnica de Catalunya
E.T.S. d'Enginyers de Camins, Canals i Ports de Barcelona
in collaboration with SEAT S.A.

Programa de Doctorat d'Enginyeria Civil

Quantifying uncertainty in complex automotive crashworthiness computational models: development of methodologies and implementation in VPS/Pamcrash.

Author:
Marc Rocas

Supervisors:
Dr. Pedro Díez,
Dr. Alberto García,
Dr. Xabier Larráyoz

*A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy
in the*
Laboratory of Computational Methods and Numerical Analysis
(LaCàN)

June 22, 2021

Declaration of Authorship

I, Marc ROCAS, declare that this thesis titled, "Quantifying uncertainty in complex automotive crashworthiness computational models: development of methodologies and implementation in VPS/Pamcrash." and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Abstract

Universitat Politècnica de Catalunya
E.T.S. d'Enginyers de Camins, Canals i Ports de Barcelona
in collaboration with SEAT S.A.

Doctor of Philosophy

Quantifying uncertainty in complex automotive crashworthiness computational models: development of methodologies and implementation in VPS/Pamcrash.

by Marc Rocas

The automotive industry is constantly involved in the development of new methodologies and projects with the aim of reducing costs. During the vehicle design process, one of the most significant cost arises from building and testing prototypes for a valid crashworthiness performance. Mathematical crash models play an important role to get a solid knowledge of the structure, aiming to achieve a successful Euro NCAP test. However, the complex nature of a crash model hinders to obtain a robust design to guarantee a good performance. Currently, in the context of crashworthiness models, particular attention is focused to uncertainties affecting the design process. Despite important improvements in modeling uncertainty quantification, theoretical simulations and experimental models are not still in perfect correlation. Starting from a computational crash model that reproduces the behaviour of the structure system, the aim of uncertainty quantification is modeling the sources of uncertainty (lack of knowledge and natural variability) from the input parameters to the output responses.

This doctoral thesis presents an uncertainty quantification methodology for complex crashworthiness models, assessing the robustness of the models and supporting decision making. Due to the high computational cost of crash models (around 18 hours for a full VPS/pamcrash model), the use of raw Monte Carlo methods for uncertainty quantification is often unaffordable. To overcome this limitation, in the first part of the thesis a review of the state-of-the-art is presented. The most relevant methods are implemented for a benchmark problem of interest for SEAT. However, some weaknesses are detected for classic approaches to deal with complex crash models. Input variability leads to nonlinear problems with high dimensional outputs. In addition, the behaviour of crash structures may have multiple hidden structure modes that can be a challenging task to be predicted. Detecting and describing these behaviours to quantify probabilities, statistics and sensitivity analysis (among other measures) can provide a potential tool for robust analysis for the SEAT portfolio.

To overcome this problem, the use of metamodels (surrogate models) is a well established approach, substituting the full order model (based on a limited number of training runs of the full order model at selected points of the input variables) for uncertainty quantification. In this doctoral thesis several techniques are studied, Ordinary Kriging, Polynomial Response Surface and a new novel surrogate strategy based on the Proper Generalized Decomposition denoted by Separated Response Surface. However, uncertainty inputs, nonlinear behaviours and large number of degrees of freedom for the outcome leads to solve high dimensional problems where the metamodel jeopardizes efficiency. Thus, previous to define a metamodel, a dimensionality reduction technique (for this thesis, kernel Principal Component Analysis) presents advantages to simplify the outcome description with the aim of building an a posteriori efficient metamodel.

This thesis develops a methodology combining dimensionality reduction and surrogate modeling for uncertainty quantification of crash problems, aiming to perform a minimum number of full order simulations, using a data-driven adaptive approach. The proposed methodology is tested for an industrial benchmark problem, demonstrating its performance for obtaining robust information of the system for multi-purpose analyses.

Acknowledgements

Th research work presented in this Doctoral thesis has been carried out at the department of Technical Computing of SEAT S.A. and at the Laboratory of Computational Methods and Numerical Analysis (LaCàN) of the Universitat Politècnica de Catalunya(UPC), both located in Barcelona. This collaboration could be done thanks to the the *Doctorats Industrials* program of the Generalitat de Catalunya.

Firtsly, I would like to express my gratitude and admiration towards my three supervisors (Xabi Larrayoz, Berto García y Pedro Díez) for their continued support and help throughout this thesis. They always inspired me by their expertise, knowledge, motivation and professionalism to tackle any problems that have arisen during this time. They consistently maintained my motivation with long talks in the most difficult moments. Their leadership skills, as good friends and bosses, made everything flow smoothly. I am completely sure that this Ph.D. thesis is much better due to the rich ideas given by them. I really want to thank them for these three years of supervision.

I would also express my gratitude to all my colleges of EK-13 department at SEAT that helped me with open arms. Specially Armando Perez, Jaime Pedret, Sara Hernandez and Daniel Dorribo for all the moments of laughter, conversations and opinions during my stay. I would like also to thank all the other Ph.D. candidates of SEAT and LaCàN. Specially my good friends Pedro Herruzo and Hector Rallo for long conversations, coffees, lunch time, workout trainings and formations.

Last but not least, I would like to thank my extended family and friends for their continuous support that gave me the strength to reach where I could not. Every conversation, every moment, every doubt, every hug of support, every travel, every trail run, every dive, every climb, every trekking, every paddle, every workout and every shared moment with endless laughs. All of them have trusted me and encouraged me day after day. I am very lucky to have them in my life and I have infinite love for them.

I especially dedicate this thesis to my lovely parents and brothers Pau and Lluís. Since I was a little boy they have listened and supported me in every moment of my life. Their love, advice and words of encouragement have helped

me to grow as a person and fight for what I want. Without them, nothing would have been possible. They are everything in my life.

Contents

List of Figures

List of Tables

List of Abbreviations

FEM	Finite Element Method
UQ	Uncertainty Quantification
VPS	Virtual Performance Solution
QoI	Quantity of Interest
SFEM	Stochastic Finite Element Method
PC	Polynomial Chaos
MC	Monte Carlo
QMC	Quasi Monte Carlo
MLMC	Multi Level Monte Carlo
OK	Ordinary Kriging
PRS	Polynomial Response Surface
SRS	Separated Response Surface
PGD	Proper Generalized Decomposition
PCA	Principal Component Analysis
kPCA	Kernel Principal Component Analysis
PDF	Probability Density Function
CDF	Cumulative Density Function
StD	Standard Deviation
KL	Kullback Leibler divergence

DR	Dimensionality Reduction
SM	Surrogate Modeling
SpC	Spearman Correlation
EK	Entwicklung Karosserie
DoE	Design of Experiments
KL-Div	Kullback Leibler divergence value between two histograms

For my parents and brothers

Chapter 1

Introduction

1.1 Crashworthiness: historical background

In the automotive industry, it is of utmost importance to carry out research and development quick and efficiently for its highly competitive market. Thus, car companies are constantly investigating for improving tools, methodologies and processes to optimize the effectiveness. The field of crashworthiness is a critical influential parameter for the marketability of a new car, specifically legal requirements and different regulations are requisites on new car projects. In European Union, the vehicles are tested with the Euro NCAP normative. Therefore, an efficient and robust crashworhtiness design is an enormous advantage over their competitors. In the beginnings, automakers focused on destructive physical testing of prototypes to analyze crashworhtiness properties. Nowadays, destructive testing of automobiles has been drastically reduced and replaced by hundreds of crash simulations.

Automotive crashworthiness simulations have their origins in the army, introduced in the 60s. Laboratories in the United State developed the explicit Finite Element Method and implemented it to crash projects in 1970 with the growth of supercomputers (Spethmann, Thomke, and Herstatt, 2006). The project was related to the impact of a aircraft at a high velocity of 200m/s on the safety containment of a nuclear power plant. The computational simulation took 33 hours to solve a model of 60 elements of the safety containment for 22 milliseconds of simulation (Haug, Scharnhorst, and Du Bois, 1986). The automotive industry observed the potential of this technology applied to crashworthiness tests. The first approaches consisted of modeling single car components. In 70s, engineering journals reported articles about numerical methods for crash simulations in the automotive field (Haug, Scharnhorst, and Du Bois, 1986). This was associated with the availability of supercomputers and the necessity of car industry to simulate components of the car at early project stage to reduce the prototype costs. One of the first projects was the simulation of a side member of a Porsche in 1983 to investigate the absorption of the kinetic energy by elastic and plastic deformations. A number of 96 elements were modeled with the Abaqus software on a supercomputer. In contrast with respect to a destructive test, virtual simulations could analyze the exact values of plastic and elastic results for each element of the model. However, with the current knowledge, this Porsche simulation was oversimplified in the geometry mesh and, consequently, unrealistic results were obtained.

In the 80s, the vectorized supercomputer appeared with significant improvements in processor speed. At this point, the first full vehicle crash simulation was simulated with the solver Pamcrash from ESI Group in the course of the Forschungsgemeinschaft Automobiltechnik working group (Du Bois et al., 2004). The conglomerate group was composed for all the seven German automakers (Audi, BMW, Ford, Mercedes-Benz, Opel, Porsche and Volkswagen).

In 1986, the greatest achievement was performed with the Volkswagen Polo model, a full frontal crash of the passenger structure was recreated. The model was meshed with 5661 finite elements, 105 beams and 5100 nodes (the maximum available for the hardware), see Fig.1.1. The simulation required 4 hours for 60 milliseconds of crash simulation. This project was an inflection point in automotive crash simulations.

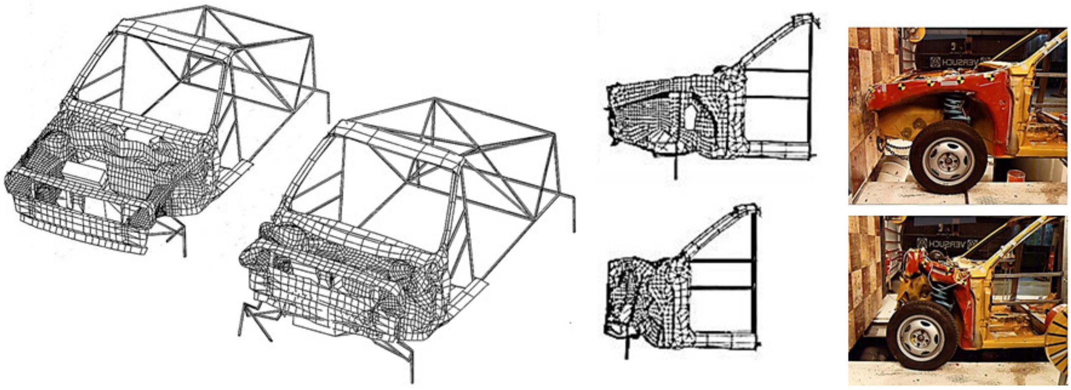


Figure 1.1 The 1986 Volkswagen Polo model for a frontal crash. Pamcrash model and real crash test. Source: (Haug, Scharnhorst, and Du Bois, 1986)

In 90s, the German magazine *Auto, Motor und Sport* created a new crash test for the automobile companies. The test demanded a frontal crash with 55km/h and 50 percent of impact for the rigid barrier. This test induced a significant increase in the load on the vehicle structure.. In consequence, the engineers employed crash simulation to achieve this goal. The Opel Astra was the first vehicle to succeed for this test with virtual simulation of a crash model of 70000 elements and 2 days of computational time for 110 milliseconds of simulation, see Fig. 1.2. At that moment, virtual simulation started to obtain the benefits of simulations compared with the traditional experimental tests, allowing fast, cheap and better understanding of crashworthiness (Böttcher, Frik, and Gosolits, 2005).

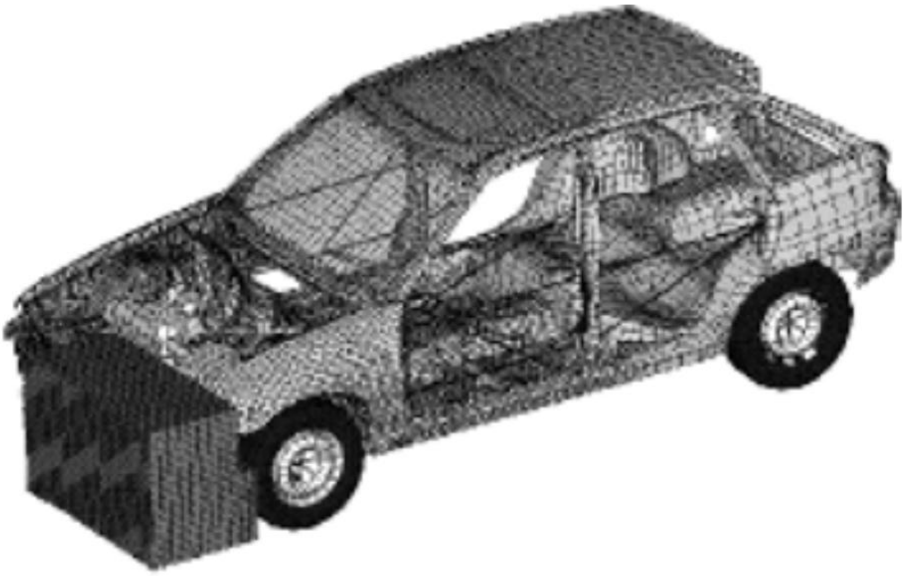


FIGURE 1.2 The 1990 Open Astra front impact model. Source: (Böttcher, Frik, and Gosolits, 2005)

In 2000s, a computational revolution appears by combining CAD, CAM and CAE as a powerful tool to develop craswhorthiness. The computational models started to become more complex, considering randomness, dummies, different parameters settings and parametrizing interiors. Fig. 1.3 shows the crash model of the 2003 Opel Astra. For this model 1,398,435 elements were considered. This model was solved in a Linux cluster instead of using a supercomputer. With the new developments in crash models, the companies significantly reduced the number of experimental prototypes for new projects.

The growth of computing power made it possible to model more complex and realistic cars, allowing hundreds of simulations with different combinations of parameters. Currently, car manufacturers work with millions of elements and hundreds of parts. However, the complexity of the models caused a problem of correlation between simulation and experimentation.

At the end of the first decade of the 2000s, an accelerated rise of artificial intelligence, machine learning and data science, brings a new field with great potential. Within a short time, large companies began to implement the scientific

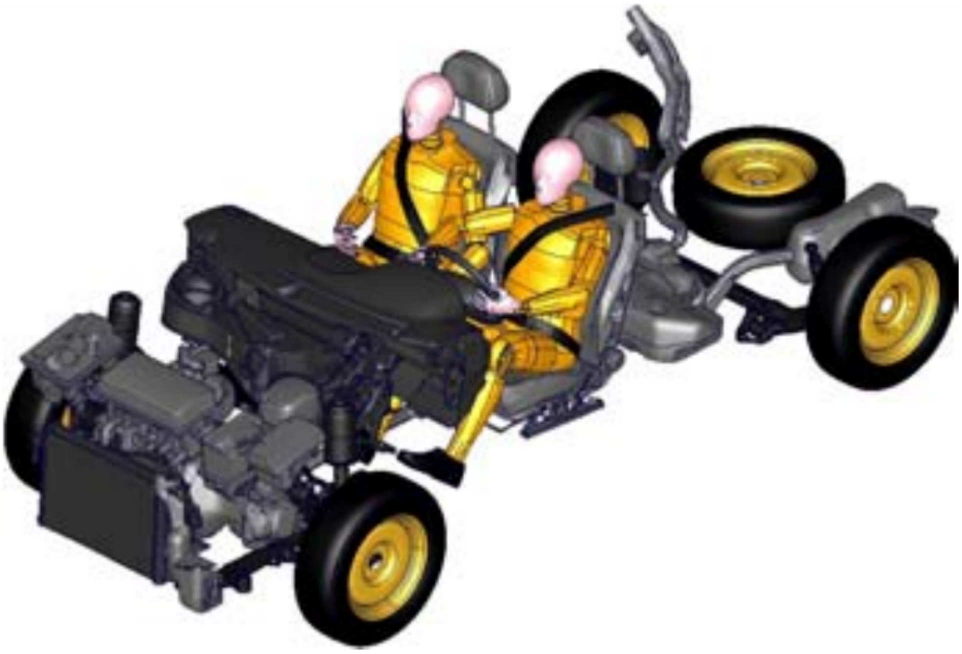


Figure 1.3 The 2003 Opel Astra model with dummies, controls, restraint systems and fuel tank. Source: (Böttcher, Frik, and Gosolits, 2005)

advances in the field of data, recognizing the enormous potential that this would bring. It drastically revolutionized the decision making processes in companies and turned data into an extremely valuable asset for predicting markets, facial recognition, language processing and predictive modeling among others. Car industries started to investigate and implement data science to their numerical problems to obtain robust models to reduce the physical tests (prototypes).

Nowadays, artificial intelligence and machine learning are combined to study thousands of data, with the aim objective of obtaining relevant information and a better understanding of the behaviour of the structures. This concept presents a new revolution for the automotive industry. The fusion between CAD, CAM, CAE and artificial intelligence foresees a very promising future for new car projects. This fusion would allow to obtain a very robust information to quantify the uncertainty from the simulations. However, this merge between these different fields presents challenging difficulties to be solved.

1.2 Scope of the thesis

In SEAT portfolio, this Industrial Doctoral thesis is proposed to tackle the aforementioned problem between simulation an experimentation. The computational models are approximations of the real world and thus may present inaccuracy results between these two correlated fields. Wrong simulation predictions are attributed to different sources of uncertainty depending on the model application. Numerical errors (discretization error in space and time for the FEM), oversimplification of the input variables (not all the parameters are taken into account) and variability in the model parameters (uncertainty knowledge of the inputs) are the main source of uncertainty. To mitigate this problem Uncertainty Quantification (UQ) has played an important role in the last years allowing to become a new field of research to address this issue. UQ field is intended to propagate the uncertainty inputs to the output responses, providing uncertainty responses of the quantity of interest. Thus, the outcome of the computational model is complemented with robust information.

The uncertainty quantification field and the increase of computational resources play an important role to manage data analysis with source of uncertainty. However, computational models are highly expensive (e.g. 18 hours for realistic crash) and not always is feasible to develop an uncertainty quantification approach. Fortunately, stochastic analysis, machine learning and artificial intelligence can deal with uncertainty quantification problems for computational models. this drives to create new methodologies and algorithms that allow to deal with expensive computational models and large datasets of information.

UQ aims to describe and understand how the uncertainty inputs propagates the variability to the output model by merging the field of applied mathematics (e.g. stochastic analysis, statistics, probability theory, mathematical models), physics (e.g. civil engineering) and data science (e.g. machine learning, artificial intelligence). However, UQ in the field of crashworthiness can present complex issues. Variability in the inputs, nonlinear behaviours for the responses and high dimensional problems leads to a complex UQ problem statement. In that cases, classic approaches as Monte Carlo method is computationally unaffordable and other techniques need to be proposed to deal with all these difficulties.

With all the aforementioned, this doctoral thesis explores the communication bridge between the fields of crash simulation, stochastic analysis, uncertainty quantification and data science to obtain robust information from simulation models.

1.3 Objectives and outline of the Doctoral thesis

The goal of this Doctoral thesis is to propose a new uncertainty quantification methodology applied for the field of crashworthiness. This goal is divided in different objectives:

- Implementation of the state-of-the-art UQ methodologies for a crashworthiness benchmark problem.
- Implementation of a nonlinear dimensionality reduction technique for nonlinear datasets.
- Development of surrogate models for crash problems with uncertainty inputs, high dimensional outputs and nonlinear behaviours.
- Implementation of advanced statistic, clustering detection (structure hidden modes) and sensitivity analysis (influence of parameters) for crash models.
- Propose a new algorithm/methodology for robust analysis for expensive crash models with nonlinear behaviours.
- Test the proposed methodology to a benchmark crash problem.
- Implementation of the novelty methodology to a realistic industrial problem for SEAT portfolio.

The thesis is organised in five chapters including this introduction. In Chapter 2 it is presented a crashworthiness benchmark problem (simplified B-Pillar model) from SEAT portfolio as the starting point for the thesis. Also, a state-of-the-art of some stochastic methods for crashworhtiness to deal with uncertainty quantification is developed. This section is divided in intrusive and non intrusive solvers. For each class, a brief theoretical explanation of the methods is given. The methods that present the most advantages for the benchmark problem are applied. Specifically Monte Carlo, Quasi Monte Carlo and Polynomial Chaos are tested. Some weaknesses are identified from the different state-of-the-art methods. To deal with that, in Chapter 3 a novel approach combining dimensionality reduction and surrogate modeling is proposed. Also, different metamodel techniques are compared, where a novel response surface based in Proper Generalized Decomposition is developed. The performance of the methodology is tested using the benchmark problem presented in Chapter 2. In addition to the presented methodology, a new adaptive approach based on dimensionality reduction and surrogate modeling for multi-purpose analysis is presented in Chapter 4. The methodology is based on an adaptive compact approach evaluating sensitivity analysis and hidden structure modes with the minimum number of evaluations of the full order model. The adaptive approach is tested with an industrial benchmark problem and compared with a Monte Carlo vademecum of simulations. A detailed summary of this thesis with the most relevant conclusions and discussion is presented in Chapter 5.

1.4 Publications and conferences

Articles in indexed journals:

- **Rocas, M.**, García-González, A., Larráyoz, X., Díez, P.: Nonintrusive stochastic finite elements for crashworthiness with VPS/Pamcrash. Archives of Computational Methods in Engineering pp. 1–26. (2020). *Impact factor=7.36.*

- **Rocas, M.**, Zlotnik, S., García-González, A., Larráyoz, X., Díez, P.: Nonintrusive Uncertainty Quantification for automotive crash problems with VPS/Pamcrash. Finite Elements in Analysis and Design 193, p. 103556. (2021). *Impact factor=2.949.*

- **Rocas, M.**, García-González, A., Larráyoz, X., Díez, P.: Adaptative surrogates fo crashworthiness models for multi-purpose engineering analyses accounting for uncertainty. Submitted (2021).

International conference contributions:

- **Rocas, M.**, S., García-González, A., Larráyoz, X., Díez, P.: Nonintrusive Uncertainty Quantification for crashworthiness simulations. ADMOS (International Conference on Adaptive Modelling and Simulation). Alicante (Spain), (2019).

Internal/Industrial talks, workshops, seminars and posters:

- **Rocas, M.**: AI-methoden für CAE. Volkswagen, Audi, Porche, Skoda, SEAT. (2020). Workshop group.

- **Rocas, M.**: Robustheit in der Berechnung. Wolfsburg, Germany (2020). Workshop group.

- **Rocas, M.**, García-González, A., Larráyoz, X., Díez, P.: Uncertainty quantification for crashworhtiness models. AK-Versagen. Volkswagen. Wolfsburg, Germany (2019).Talk.
- **Rocas, M.**: Nonintrusive Stochastic Finite Element Method for crashworthiness with VPS Pamcrash. NumROM LaCan. Barcelona, Spain (2018), Scientific Seminar.
- **Rocas, M.**: Uncertainty quantification with stochastic Finite Element Methods for crashworthiness. SEAT Future Mobility Day. Barcelona, Spain (2018). Poster.
- **Rocas, M.**: Uncertainty quantification with stochastic Finite Element Methods for crashworthiness. ProMotion Volkswagen. Wolfsburg, Germany (2018). Poster.

This Doctoral thesis is presented in the Industrial Doctorate framework funded by *Generalitat de Catalunya*, *Ministerio de Economía y Empresa* and *Ministerio de Ciencia, Innovación y Universidades.* This research was proposed and developed in collaboration between the Laboratory of mathematical and computational modelling (LaCàN) of the School of Civil Engineering (Universitat Politècnica de Catalunya) and the Entwicklung Karosserie (EK) department of structure calculation of SEAT (Martorell).

Chapter 2

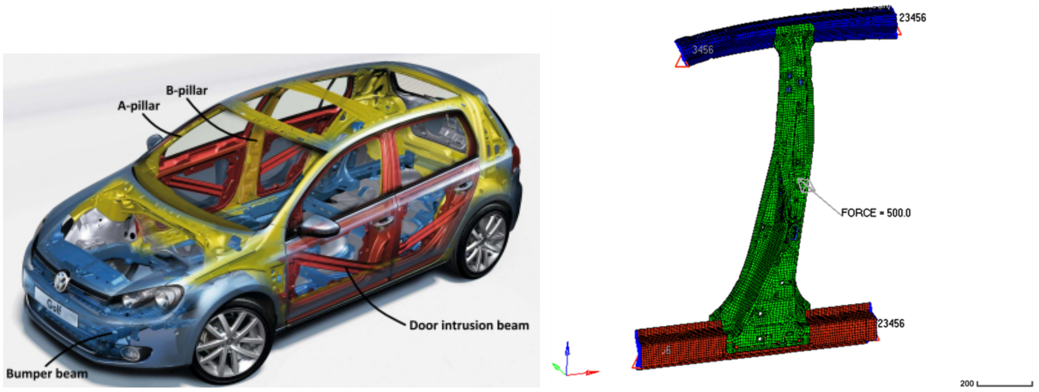
Uncertainty Quantification for crashwhortiness

This chapter presents a state-of-the-art of the most suitable techniques for stochastic analysis in the field of crashworhtiness. The different techniques are divided in intrusive and non intrusive solvers, that is the nature of the two group of techniques. This Chapter is structured as follows: section 2.1 describes the benchmark crash problem used for different methods. The stochastic approaches are reviewed, both the intrusive and non-intrusive strategies, in Sections 2.2 and 2.3 respectively. Next, Section 2.4 shows a comparative analysis of the SFEM solvers. Section 2.5, illustrates the numerical results obtained using the aforementioned stochastic UQ techniques. Finally, with Section 2.6 the chapter closes with a discussion and conclusions.

The content of this chapter has been published in *Archives of Computational Methods in Engineering* (Rocas et al., 2020).

2.1 Benchmark for a B-Pillar crash model

This chapter discusses the problem statement presented by SEAT for UQ analyses as a starting point for the thesis. The benchmark problem presented makes reference to a B-pillar problem from a vehicle. In Fig. 2.1 a realistic B-pillar is illustrated. The B-pillar provides structural support for the vehicle's roof panel and is designed for blocking the front door and absorb the energy from a side crash. This component is one of the most sophisticated parts of the vehicle body presenting unpredictable nonlinear behaviours. It is demonstrated that small changes in the input variables present big changes in the output response. This leads to obtain different modes of the structure and in consequence a loss of confidence with the model. This lack of knowledge with poor robustness leads to a lack of trust to move on to prototyping and experimentation.



(A) Full vehicle render.

(B) B-pillar model.

FIGURE 2.1 Volkswagen Golf VI with highlighted structural components (Volkswagen AG, 2012)

2.1.1 Model description

The benchmark crash problem under consideration is illustrated in Fig. 2.2. It corresponds to a reduced test model, ideally reproducing the main characteristics of the simulation of a B-pillar, a well-known structural component of cars. This particular benchmark test is used for different research studies in the Volkswagen Group. For the sake of saving computational cost and time, this model is often used to test new materials, adhesives, welding spots or other conditions because, due to the simplicity of the model, the numerical response requires a computation of approximately 20 minutes.

The driving force in the model is provided by the impactor (green zone in Fig. 2.2), that crashes at a speed of 50 mm/s against the vertical profile (red zone) during one second. The three structural parts are plates made of laminated steel sheet manufactured by cold folding. All the parts are joined with a structural adhesive bond, its material properties are characterized by Volkswagen with a confidential character.

The complete structure is modelled with shell elements. The impactor is considered to be a rigid body. The whole model has a grand total of $N = 13908$ nodes (with 6 degrees of freedom). The Quantity of Interest (QoI) to be analysed is the final plastic strain average of the 142 shell elements of the area depicted in black in Fig. 2.2.

The numerical solver is implemented in VPS/Pamcrash (*PAM-SCL - Theory Notes Manual* 2000), with the shell finite element discretization mentioned above and an explicit time stepping scheme to solve the dynamical problem. The displacements of the points at the ends of the horizontal profile are prescribed

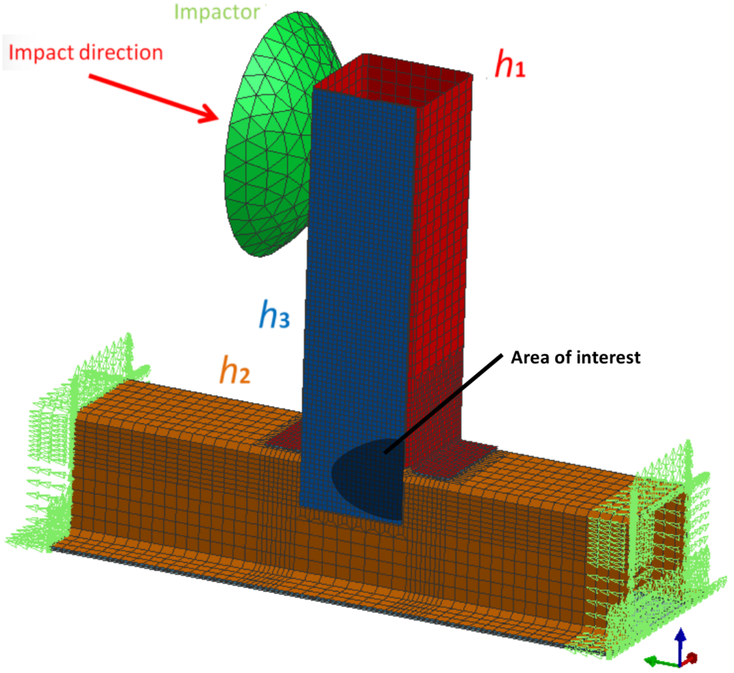


FIGURE 2.2 Crash benchmark. Thicknesses h_1, h_2 and h_3 are the three input random parameters corresponding to the vertical profile (red), horizontal profile (orange) and plate profile (blue). The impactor (green), and the area of elements of interest (black) are also depicted.

to zero (points marked with green arrows in Fig. 2.2). The contact between the different components of the structure are treated with the surface-surface model defined in VPS/Pamcrash.

Thicknesses h_1, h_2 and h_3 of the three parts of the structure are considered to be stochastic parameters, that random variables are collected in vector $\boldsymbol{h} = [h_1, h_2, h_3]^\mathsf{T}$. Their aleatory character is associated with the imperfections produced during the manufacturing process. Random variables h_1, h_2 and h_3 are assumed to be normal and uncorrelated, that is $h_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$ and $\mathrm{cov}(h_i, h_j) = 0$, for $i, j = 1, 2, 3$. In each of the three parts, the corresponding thickness is considered to be constant. Besides, the three thicknesses h_1, h_2 and h_3 are modelled as having the same mean $\mu_1 = \mu_2 = \mu_3 = 1.2\,\mathrm{mm}$ and standard deviation $\sigma_1 = \sigma_2 = \sigma_3 = 0.12\,\mathrm{mm}$.

In order to build a training set, and as a first assessment of the stochastic behavior of the system, a number of $\mathtt{n_s}$ Monte Carlo realizations (or samples) are performed. Thus, $\mathtt{n_s}$ values of the input parameters $\boldsymbol{h}^i$, for $i = 1, 2, \ldots, \mathtt{n_s}$ are generated with a random number generator and the corresponding VPS/Pamcrash solutions are obtained. These solutions (in particular the vectors containing the plastic strain in the $d = 142$ elements of the zone of interest) are collected in a training set matrix $\mathbf{X} = [\mathbf{x}^1 \mathbf{x}^2 \cdots \mathbf{x}^{\mathtt{n_s}}] \in \mathbb{R}^{d \times \mathtt{n_s}}$, where each column $\mathbf{x}^i = [x_1^i \ldots x_d^i]^\mathsf{T}$ is the VPS/Pamcrash solution corresponding to input $\boldsymbol{h}^i$. The actual QoI is the average plastic strain in the specific area plotted in black in Fig. 2.2, is represented by a form $l^0(\cdot)$, and for each $\boldsymbol{h}^i$ and $\mathbf{x}^i$ reads

$$Y_{\mathtt{QoI}} = l^0(\mathbf{x}^i) = \frac{1}{d} \sum_{j=1}^{d} x_j^i.$$

The QoI is introduced as an essential indicator for decision making. The QoI summarizes the information contained in $\mathbf{x}$. Quantifying the uncertainty of the QoI is sufficient to take some decisions. For instance, to verify the crashworthiness response of the structural design. Note that the Monte Carlo process with $\mathtt{n_s}$ samples is considered for the thesis as a reference method, and it is only obtained in the academic example under consideration. The number of full-scale computations affordable for a real problem in the automotive industrial practice is much lower.

2.2 Intrusive SFEM solvers

This section summarizes the state-of-art of the most significant intrusive Stochastic Finite Element Method (SFEM) applied in crashworthiness for UQ analysis.

Intrusive methods reformulate the deterministic finite element matrix scheme into a stochastic model SFEM by including the randomness of the variables. Despite the computational complexity, SFEM techniques are used for solving stochastic partial differential equations.

2.2.1 Formulation and notation

For an UQ analysis on the proposed benchmark, the comercial software VPS solves the equilibrium equation of transient dynamics (*PAM-SCL - Theory Notes Manual* 2000)

$$\mathbf{M}\ddot{\mathbf{U}} + \mathbf{C}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F}_{ext}(t), \tag{2.1}$$

where $\mathbf{M}$ (Mass matrix), $\mathbf{C}$ (damping matrix), $\mathbf{K}$ (Stiffness matrix), $\mathbf{F}_{ext}$ (external force) and t (time).

Nevertheless, for the sake of simplicity, a linear static problem is developed to illustrate the used notation of the different stochastic techniques. The compact equation corresponds to:

$$\mathbf{K}(\boldsymbol{h})\mathbf{U}(\boldsymbol{h}) = \mathbf{F}(\boldsymbol{h}) \tag{2.2}$$

where $\mathbf{K}(\boldsymbol{h})$ is the stiffness matrix, $\mathbf{F}(\boldsymbol{h})$ the load vector and $\mathbf{U}(\boldsymbol{h})$ the vector of unknowns, and $\boldsymbol{h}$ is the vector containing a random discretization, where the number of realizations is determined by $\mathtt{n_s}$ samples. The global stiffness matrix $\mathbf{K}(\boldsymbol{h})$ is obtained after assembling the elemental matrices $\mathbf{K}^e$,

$$\mathbf{K}^e = \int_{\Omega_e} \mathbf{B}^T\mathbf{D}\mathbf{B}d\Omega_e \tag{2.3}$$

where $\mathbf{D}$ corresponds to the elasticity matrix which depends on the Lamé parameters, and $\mathbf{B}$ is the matrix that relates the components of the stress with the nodal displacements. Assuming that matrix $\mathbf{D}$ has a stochastic behaviour, the elasticity matrix is given by:

$$\mathbf{D}((x,y,z),\boldsymbol{h}) = R((x,y,z),\boldsymbol{h})\mathbf{D}_0 \tag{2.4}$$

where $R((x,y,z),\boldsymbol{h})$ is the random field and $\mathbf{D}_0$ is the mean value of the elasticity matrix. The behavior of the random field $R((x,y,z),\boldsymbol{h})$ is described by the mean μ and a fluctuation function $P(\boldsymbol{h})$ such that $R((x,y,z),\boldsymbol{h}) = \mu((x,y,z)) + P((x,y,z),\boldsymbol{h})$, thus the stiffness matrix becomes:

$$\begin{aligned} \mathbf{K}((x,y,z),\boldsymbol{h}) &= \int_{\Omega} R((x,y,z),\boldsymbol{h})\mathbf{B}^T\mathbf{D}_0\mathbf{B}d\Omega \qquad (2.5) \\ &= \int_{\Omega} \big(\mu(x,y,z) + P((x,y,z),\boldsymbol{h})\big)\mathbf{B}^T\mathbf{D}_0\mathbf{B}d\Omega. \end{aligned}$$

The purpose of the stochastic analysis is to determine reliable statistical information of a QoI response from the solution $\mathbf{U}(\boldsymbol{h})$, which is a random field. Recall that for our dynamic analysis, the QoI is chosen to be the mean of the plastic strain of the area of interest illustrated in Fig. 2.2.

2.2.2 Perturbation Method

The Perturbation Method was introduced in 1970 to solve a large number of problems with uncertainty inputs. The main idea is to propagate the uncertainty by Taylor series (Kleiber and Hien, 1992; Arregui-Mena, Margetts, and Mummery, 2016), this technique has been used in structural engineering to solve nonlinear dynamic problems (Liu, Belytschko, and Mani, 1986b; Liu, Belytschko, and Mani, 1986a).

The stochastic parameters used to construct the stifness matrix $\mathbf{K}(\boldsymbol{h})$ are varying around their mean. In this approach the stochastic field $R((x,y,z),\boldsymbol{h})$ is discretized into $\mathtt{n_s}$ zero-mean random variables $(\boldsymbol{h}^i)_{i=1}^{\mathtt{n_s}}$. Hence, expanding the stiffness matrix $\mathbf{K}((x,y,z),\boldsymbol{h})$ as a Taylor series around their mean $\mathbf{K}_0$ reads (Stefanou, 2009):

$$\mathbf{K}((x,y,z),\boldsymbol{h}) = \mathbf{K}_0 + \sum_{i=1}^{\mathtt{n_s}} \mathbf{K}_i^I \boldsymbol{h}^i + \frac{1}{2}\sum_{i=1}^{\mathtt{n_s}}\sum_{j=1}^{\mathtt{n_s}} \mathbf{K}_{ij}^{II} \boldsymbol{h}^i \boldsymbol{h}^j + ..., \tag{2.6}$$

where $\mathbf{K}_0 = \int_\Omega \mu \mathbf{B}^T \mathbf{D}_0 \mathbf{B} d\Omega$ is the mean value of the stiffness matrix. Furthermore, $\mathbf{K}_i^I$, $\mathbf{K}_{ij}^{II}$ are the first and second order derivatives respectively, evaluated at $\boldsymbol{h} = 0$, and represent the fluctuation part of the stiffness matrix:

$$\mathbf{K}_i^I = \frac{\partial \mathbf{K}((x,y,z),\boldsymbol{h})}{\partial \boldsymbol{h}^i}|_{\boldsymbol{h}=0}. \tag{2.7}$$

$$\mathbf{K}_{ij}^{II} = \frac{\partial^2 \mathbf{K}((x,y,z),\boldsymbol{h})}{\partial \boldsymbol{h}^i \partial \boldsymbol{h}^j}|_{\boldsymbol{h}=0}. \tag{2.8}$$

For solving of the system $\mathbf{KU} = \mathbf{F}$, the propagated Taylor expansion of vector $\mathbf{F}$ corresponds to

$$\mathbf{F}(\boldsymbol{h}) = \mathbf{F}_0 + \sum_{i=1}^{\mathtt{n_s}} \mathbf{F}_i^I \boldsymbol{h}^i + \frac{1}{2}\sum_{i=1}^{\mathtt{n_s}}\sum_{j=1}^{\mathtt{n_s}} \mathbf{F}_{ij}^{II} \boldsymbol{h}^i \boldsymbol{h}^j + ..., \tag{2.9}$$

and assuming the external forces $\mathbf{F}$ as deterministic (no random behaviour), then the first and second derivatives are $\mathbf{F}_i^I = \mathbf{F}_{ij}^{II} = 0$, thus $\mathbf{F} = \mathbf{F}_0$.

Besides, the propagated Taylor expansion of vectors $\mathbf{U}$ reads

$$\mathbf{U}(\boldsymbol{h}) = \mathbf{U}_0 + \sum_{i=1}^{\mathtt{n_s}} \mathbf{U}_i^I \boldsymbol{h}^i + \frac{1}{2}\sum_{i=1}^{\mathtt{n_s}}\sum_{j=1}^{\mathtt{n_s}} \mathbf{U}_{ij}^{II} \boldsymbol{h}^i \boldsymbol{h}^j + ... \tag{2.10}$$

where the terms U_0, U_i^I and U_{ij}^{II} can be calculated by substituting (2.10) and the deterministic $\mathbf{F}_0$ into (2.2) and identifying the similar order coefficients on both sides of the equation, is obtained the following iterative scheme:

$$\mathbf{U}_0 = \mathbf{K}_0^{-1}\mathbf{F}_0 \tag{2.11}$$

$$\mathbf{U}_i^I = -\mathbf{K}_0^{-1}\mathbf{K}_i^I\mathbf{U}_0 \tag{2.12}$$

$$\mathbf{U}_{ij}^{II} = -\mathbf{K}_0^{-1}(\mathbf{K}_i^I\mathbf{U}_j^I + \mathbf{K}_i^I\mathbf{U}_i^I + \mathbf{K}_{ij}^{II}\mathbf{U}_0) \tag{2.13}$$

Equation (2.11) gives the deterministic nodal displacement, equation (2.12) and (2.13) give the first and second order perturbations, respectively, of the displacement vector. Additionally, the statistics of $\mathbf{U}(\boldsymbol{h})$ with the mean $\mathbb{E}(\mathbf{U}(\boldsymbol{h}))$ and the covariance matrix $Cov[\mathbf{U}(\boldsymbol{h}), \mathbf{U}(\boldsymbol{h})]$ read:

$$\mathbb{E}(\mathbf{U}(\boldsymbol{h})) \approx \mathbf{U}_0 + \frac{1}{2}\sum_{i=1}^{\mathtt{n_s}}\sum_{j=1}^{\mathtt{n_s}} \mathbf{U}_{ij}^{II} Cov[\boldsymbol{h}^i, \boldsymbol{h}^j], \tag{2.14}$$

$$Cov[\mathbf{U}(\boldsymbol{h}), \mathbf{U}(\boldsymbol{h})] \approx \sum_{i=1}^{\mathtt{n_s}}\sum_{j=1}^{\mathtt{n_s}} \mathbf{U}_i^I (\mathbf{U}_j^I)^T Cov[\boldsymbol{h}^i, \boldsymbol{h}^j]. \tag{2.15}$$

Increasing the number of terms in the expansion will improve the accuracy of the perturbation method, but affecting the computational cost.

2.2.3 Galerkin Polynomial Chaos Method

This section is devoted to review the formulation of Galerkin Polynomial Chaos (PC) method for UQ. By considering Gaussian random inputs, the main idea of Polynomial Chaos is to propagate the uncertainty through a Hermite polynomials basis (Ghanem and Spanos, 2003). This stochastic method involves two basic steps:

- Step 1: Implementing the polynomial Hermite expansion into the the stochastic formulation of the model.
- Step 2: Applying Galerkin projection basis to get the polynomial chaos coefficients.

Applying Hermite polynomials for the terms of the equilibrium equation (2.2), allow to extend $\mathbf{K}(\boldsymbol{h})$, $\mathbf{U}(\boldsymbol{h})$ and $\mathbf{F}(\boldsymbol{h})$ as polynomial expansions in the form:

$$\begin{aligned} \mathbf{K}(\boldsymbol{h}) &= \sum_{i=0}^{\mathtt{n_{KC}}} \mathbf{K}_i \Psi_i(\boldsymbol{h}), \\ \mathbf{U}(\boldsymbol{h}) &= \sum_{i=0}^{\mathtt{n_{PC}}} \mathbf{U}_i \Psi_i(\boldsymbol{h}), \\ \mathbf{F}(\boldsymbol{h}) &= \sum_{i=0}^{\mathtt{n_{FC}}} \mathbf{F}_i \Psi_i(\boldsymbol{h}), \end{aligned} \tag{2.16}$$

where $\mathbf{K}_i, \mathbf{U}_i, \mathbf{F}_i$ are the polynomial chaos coefficients (known as Fourier coefficients) and $\Psi_i(\boldsymbol{h})$ are orthogonal basis. Where, a main condition for using this technique is that the random inputs have to be independent variables. For the sake of simplicity, the polynomial basis of the equation (2.16) corresponds to $\Psi_i(\{h_k\}_{k=1}^3)$. The number of terms $\mathtt{N}_{\mathtt{KC,PC,FC}}$ are defined as

$$\mathtt{N}_{\mathtt{KC,PC,FC}} + 1 = \frac{(M + \mathtt{d}_{\mathtt{KC,PC,FC}})!}{M!\,\mathtt{d}_{\mathtt{KC,PC,FC}}!}, \tag{2.17}$$

being $\mathtt{d}_{\mathtt{KC}}$, $\mathtt{d}_{\mathtt{PC}}$ and $\mathtt{d}_{\mathtt{FC}}$ the order of the polynomial expansions, and M the number of stochastic variables.

To construct a proper orthogonal polynomial basis is necessary to know the Probability Density Function (PDF) of the inputs in the called Askey-Wilson scheme (Askey and Wilson, 1985; Mathelin, Hussaini, and Zang, 2005) to guarantee a good convergence. For example, Hermite polynomial basis for gaussian distribution, and Legendre polynomials for uniform distribution (Xiu and Karniadakis, 2002; Zhang, 2013; Xiu, 2010).

In the case of our benchmark problem, three Gaussian input variables are considered $\boldsymbol{h} = [h_1, h_2, h_3]^T$. The multivariate Hermite polynomials are created by the tensor product of the univariate polynomials of each random input (Feinberg and Langtangen, 2015). Table 2.1 shows the corresponding relation between multi-indexes and single-indexes for the calculation of the multivariate basis of Hermite polynomial basis of order 2.

Multi-index	Multi-polynomial	i	$\Psi_j(\boldsymbol{h})$
(0 0 0)	$\Psi_0(h_1)\Psi_0(h_2)\Psi_0(h_3)$	0	1
(1 0 0)	$\Psi_1(h_1)\Psi_0(h_2)\Psi_0(h_3)$	1	h_1
(0 1 0)	$\Psi_0(h_1)\Psi_1(h_2)\Psi_0(h_3)$	2	h_2
(0 0 1)	$\Psi_0(h_1)\Psi_0(h_2)\Psi_1(h_3)$	3	h_3
(2 0 0)	$\Psi_2(h_1)\Psi_0(h_2)\Psi_0(h_3)$	4	$h_1^2 - 1$
(1 1 0)	$\Psi_1(h_1)\Psi_1(h_2)\Psi_0(h_3)$	5	$h_1 h_2$
(1 0 1)	$\Psi_1(h_1)\Psi_0(h_2)\Psi_1(h_3)$	6	$h_1 h_3$
(0 2 0)	$\Psi_0(h_1)\Psi_2(h_2)\Psi_0(h_3)$	7	$h_2^2 - 1$
(0 1 1)	$\Psi_0(h_1)\Psi_1(h_2)\Psi_1(h_3)$	8	$h_2 h_3$
(0 0 2)	$\Psi_0(h_1)\Psi_0(h_2)\Psi_2(h_3)$	9	$h_3^2 - 1$

TABLE 2.1 Multi-index Hermite Polynomials of three dimension. Note that the first column describes the degree of the univariate polynomials.

As in the Perturbation method, it is assumed that the external force $\mathbf{F}$ is deterministic. Therefore the equilibrium equation 2.2 takes the form:

$$\left(\sum_{i=0}^{\mathtt{n}_{\mathtt{KC}}} \mathbf{K}_i \Psi_i(\boldsymbol{h})\right)\left(\sum_{j=0}^{\mathtt{n}_{\mathtt{PC}}} \mathbf{U}_j \Psi_j(\boldsymbol{h})\right) = \mathbf{F} \tag{2.18}$$

becoming,

$$\sum_{i=0}^{\mathtt{n_{KC}}}\sum_{j=0}^{\mathtt{n_{PC}}}\mathbf{K}_i\mathbf{U}_j\Psi_i(\boldsymbol{h})\Psi_j(\boldsymbol{h}) = \mathbf{F}. \tag{2.19}$$

The above equation is projected onto n^{th} polynomial basis with $n = (0, 1, .., \mathtt{n_{PC}})$. To simplify the formulation, It is assumed that $\mathtt{n_{KC}} = \mathtt{n_{PC}}$, and therefore, making the inner product, it is obtained the expression:

$$\left\langle \sum_{i=0}^{\mathtt{n_{PC}}}\sum_{j=0}^{\mathtt{n_{PC}}}\mathbf{K}_i\mathbf{U}_j\Psi_i(\boldsymbol{h})\Psi_j(\boldsymbol{h}), \Psi_n(\boldsymbol{h}) \right\rangle = \langle \mathbf{F}, \Psi_n(\boldsymbol{h})\rangle \,, \tag{2.20}$$

renaming $\Psi_i(\boldsymbol{h})$ as Ψ_i to relax and compact the notation, the previous expression can be written such that

$$\sum_{i=0}^{\mathtt{n_{PC}}}\sum_{j=0}^{\mathtt{n_{PC}}}\mathbf{K}_i\mathbf{U}_j \langle \Psi_i\Psi_j, \Psi_n\rangle = \langle \mathbf{F}, \Psi_n\rangle \,. \tag{2.21}$$

Describing the inner product of two random functions, $f(v)$ and $g(v)$, as:

$$\big\langle f(v), g(v)\big\rangle = \mathbb{E}(f(v)g(v)) = \int_R f(v)g(v)p_v(v)dv, \tag{2.22}$$

being $p_v(v)$ the probability density function of v. Therefore, let us introduce the following notation:

$$C_{ijn} = \mathbb{E}[\Psi_i\Psi_j\Psi_n] = \langle \Psi_i\Psi_j, \Psi_n\rangle \,, \tag{2.23}$$

$$\mathbf{F}_n = \langle \mathbf{F}, \Psi_n\rangle \,. \tag{2.24}$$

Then, equation (2.21) can be rewritten as:

$$\left(\sum_{i=0}^{\mathtt{n_{PC}}}\sum_{j=0}^{\mathtt{n_{PC}}}\mathbf{K}_iC_{ijn}\right)\mathbf{U}_j = \mathbf{F}_n \tag{2.25}$$

For the sake of simplicity, let us define

$$\mathbf{K}_{jn} = \sum_{i=0}^{\mathtt{n_{PC}}}\mathbf{K}_iC_{ijn} \tag{2.26}$$

Thus the equation (2.25) reads

$$\left(\sum_{j=0}^{\mathtt{n_{PC}}}\mathbf{K}_{jn}\right)\mathbf{U}_j = \mathbf{F}_n \tag{2.27}$$

which means:

$$\begin{pmatrix} \mathbf{K}_{00} & \dots & \mathbf{K}_{0,\mathtt{n_{PC}}} \\ \mathbf{K}_{10} & \dots & \mathbf{K}_{1,\mathtt{n_{PC}}} \\ \vdots & \ddots & \vdots \\ \mathbf{K}_{\mathtt{n_{PC}},0} & \dots & \mathbf{K}_{\mathtt{n_{PC}},\mathtt{n_{PC}}} \end{pmatrix} \begin{pmatrix} \mathbf{U}_0 \\ \mathbf{U}_1 \\ \vdots \\ \mathbf{U}_{\mathtt{n_{PC}}} \end{pmatrix} = \begin{pmatrix} \mathbf{F}_0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$\mathbf{U}_j$ is a N-dimensional vector (N is the number of nodes of the finite element model), and $\mathbf{K}_{jn}$ is a matrix of size (N x N). Therefore, the global linear system will have $(N \cdot \mathtt{n_{PC}})$ x $(N \cdot \mathtt{n_{PC}})$ size. In this case, where $\mathbf{F}$ is deterministic, only $\mathbf{F}_0$ is non-zero. The computational cost to solve the linear system is thus much greater with respect to a deterministic approach (Ghanem and Spanos, 2003). Additionally, the first vector of coefficients $\mathbf{U}_0$ corresponds to the mean of $\mathbf{U}$, that is $\mathbb{E}[\mathbf{U}] = \mathbf{U}_0$.

At this point, for post-processing analysis in polynomial chaos, it is useful to define the covariance matrix of all the components of $\mathbf{U}$, which read:

$$Cov[\mathbf{U}, \mathbf{U}] = \sum_{i=0}^{\mathtt{n_{PC}}} \mathbb{E}[\Psi_i^2]\mathbf{U}_i \cdot \mathbf{U}_i^T. \tag{2.28}$$

2.2.4 Spectral Stochastic Finite Element Method

This section presents a briefly review of the Spectral Stochastic Finite Element Method developed by Ghanem and Spanos in 1991 (Ghanem and Spanos, 2003). The essence of this approach is to combine Karhunen-Loeve expansion (for the stochastic input parameters) with Polynomial Chaos (for the response variability(Sudret and Der Kiureghian, 2000)). In the literature, different developments have been applied combining Karhunen-Loeve and PC (Ghanem and Spanos, 2003; Ghanem and Kruger, 1996; Nouy, 2009; Doostan, Ghanem, and Red-Horse, 2007).

For this method, we suppose that each random input h_i is described as a random field $H_i((x, y, z))$. Then, each field can be discretized as a finite number of uncorrelated random variables by the truncated Karhunen-Loéve decomposition (Shinozuka and Deodatis, 1991; Liu, Liu, and Peng, 2017; Stefanou and Papadrakakis, 2007). Aiming to reduce the dimensionality of the problem to deal with an stochastic analysis. To relax and simplify the notation, for this approach it is considered one random field to illustrate the theoretical concepts of the method. The Karhunen-Loève decomposition allows representing a random field by a sum of mutually uncorrelated (zero-mean) scalar random variables multiplied by deterministic functions (Grigoriu, 2006), namely

$$H((x,y,z),\boldsymbol{h}) = \mu(x,y,z) + \sum_{i=1}^{\mathtt{n_{KL}}} \sqrt{\lambda_i}\xi_i(\boldsymbol{h})\varphi_i(x,y,z), \tag{2.29}$$

where $\mu(x,y,z)$ is the mean of the random field $H((x,y,z),\boldsymbol{h})$, λ_i and $\varphi_i(x,y,z)$ are the deterministic eigenvalues and eigenfunctions of to the covariance function respectively. To relax and simplify the notation for this approach it is considered only a random field. The eigenfunction $\varphi_i(x,y,z)$ are obtained through a spectral covariance decomposition. The stochasticity of the system is approximated by $\mathtt{n_{KL}}$ uncorrelated standard Gaussian random variables $\xi_i(\boldsymbol{h})$ are the random variables knowns as the Fourier coefficients, with $i = 1, 2, ...\mathtt{n_{KL}}$. The readers are referred to (Ghanem and Spanos, 2003; Ghanem and Kruger, 1996) for deeper theoretical details on Karhunen-Loéve technique.

The stiffness matrix $\mathbf{K}$ is computed by Karhunen-Loeve substituting in (2.3) equations (2.4) and (2.29), becoming the elemental matrix $\mathbf{K}^e(\boldsymbol{h})$

$$\mathbf{K}^e(\boldsymbol{h}) = \mathbf{K}_0^e + \sum_{i=1}^{\mathtt{n_{KL}}} \mathbf{K}_i^e \xi_i(\boldsymbol{h}), \tag{2.30}$$

where $\mathbf{K}_0^e$ is the mean value $\int_{\Omega_e} \mu((x,y,z))\mathbf{B}^T\mathbf{D}_0\mathbf{B}d\Omega_e$ and $\mathbf{K}_i^e$ are deterministic matrices that describe the fluctuation part of the stiffness matrix, given by:

$$\mathbf{K}_i^e = \sqrt{\lambda_i}\int_{\Omega_e} \varphi_i(x,y,z)\mathbf{B}^T\mathbf{D}_0\mathbf{B}d\Omega_e. \tag{2.31}$$

Assuming $\mathbf{F}$ is deterministic and expanding the unknown vector $\mathbf{U}(\boldsymbol{h})$ by PC expansion, the finite element equilibrium system reads:

$$\left(\sum_{i=0}^{\mathtt{n_{KL}}} \mathbf{K}_i \xi_i(\boldsymbol{h})\right) \cdot \left(\sum_{j=0}^{\mathtt{n_{PC}}} \mathbf{U}_j \Psi_j(\boldsymbol{h})\right) = \mathbf{F}. \tag{2.32}$$

Physically, $\mathbf{K}_0$ refers to the mean stiffness, and $\mathbf{K}_i$ to the random fluctuation around the mean. After some algebraic manipulations (analogously to Galerkin PC method), the equilibrium system is given by:

$$\mathbf{K}_{ij} = \sum_{i=0}^{\mathtt{n_{KL}}} C_{ijk} \cdot \mathbf{K}_i; \quad k = 0, ..., \mathtt{n_{PC}}, \tag{2.33}$$

where $C_{ijk} = \mathbb{E}[\xi_i \Psi_j \Psi_k]$, and finally obtaining

$$\left(\sum_{i=0}^{\mathtt{n_{PC}}} \mathbf{K}_{ij} \right) \mathbf{U}_j = \mathbf{F}_n. \tag{2.34}$$

From above expressions, the statistics of the coefficient vector $\mathbf{U} = [\mathbf{U}_0, \mathbf{U}_1, ..., \mathbf{U}_{\mathtt{n_{PC}}}]^T$ are described as in Galerkin PC, since the stochastic response propagation in Spectral Stochastic Finite Element Method is developed by a PC expansion.

2.3 Non intrusive SFEM solvers

Non-intrusive techniques does not require reformulation of the source code, which facilitates the statistical analyses by direct pre and post processing methods. This makes that techniques highly recommended to be applied in a wide range of fields from integrated circuits to computational fluid dynamics (Kaintura, Dhaene, and Spina, 2018; Phoon, Huang, and Quek, 2005). In this section, it is reviewed the most suitable methods for crashworthiness.

2.3.1 Non-intrusive Polynomial Chaos expansion

Non-intrusive Polynomial Chaos (Eldred, 2009) is based on a decomposition of a random function $Y(\mathbf{U}(\boldsymbol{h}))$ into deterministic and stochastic components in a separable manner. Thus, the quantity of interest $Y_{\mathtt{QoI}} = Y(\mathbf{U}(\boldsymbol{h}))$ considered for the computational crash problem is represented as a Polynomial Chaos expansion by the expression:

$$Y_{\mathtt{QoI}} = Y(\mathbf{U}(\boldsymbol{h})) = \sum_{n=0}^{\mathtt{n_{PC}}} c_n \cdot \Psi_n(\boldsymbol{h}), \tag{2.35}$$

where c_n are the deterministic Fourier coefficients and $\Psi_n(\boldsymbol{h})$ are the random basis functions (orthogonal polynomials chosen in the Askey-Wilson scheme (Askey and Wilson, 1985)). Recall that for using polynomial chaos, the inputs have to be independent, however, if there are dependencies between them, it is necessary additional methods. For more details the reader is referred to (Feinberg and Langtangen, 2015).

In the following, an overview of the two main techniques to estimate Fourier Coefficients are described: Pseudo Spectral Projection (quadrature based) and Point Collocation (least square minimisation).

Pseudo Spectral Projection Polynomial Chaos

Pseudo Spectral Projection method (or quadrature polynomial chaos) is based on a quadrature scheme to obtain the coefficients c_n (Phoon, Huang, and Quek, 2005; Hosder, Walters, and Perez, 2014). Each coefficient is calculated by projecting equation (2.35) onto the n^{th} basis, with $n = 0, 1, .., \mathtt{n_{PC}}$ such that

$$\left\langle Y_{\mathtt{QoI}}, \Psi_n(\boldsymbol{h}) \right\rangle = \left\langle \sum_{n=0}^{\mathtt{n_{PC}}} c_n \Psi_n(\boldsymbol{h}), \Psi_n(\boldsymbol{h}) \right\rangle . \tag{2.36}$$

Using the orthogonality properties of the basis functions:

$$\left\langle Y_{\mathtt{QoI}}, \Psi_n(\boldsymbol{h}) \right\rangle = c_n \left\langle \Psi_n^2(\boldsymbol{h}) \right\rangle , \tag{2.37}$$

then,

$$c_n = \frac{\langle Y_{\mathtt{QoI}}, \Psi_n(\boldsymbol{h}) \rangle}{\langle \Psi_n^2(\boldsymbol{h}) \rangle} . \tag{2.38}$$

Recalling the definition (2.22), the previous equation becomes:

$$c_n = \frac{\mathbb{E}(Y_{\mathtt{QoI}}, \Psi_n)}{\mathbb{E}(\Psi_n^2)} = \frac{1}{\mathbb{E}(\Psi_n^2)} \int_{\Omega} Y_{\mathtt{QoI}} \Psi_n(\boldsymbol{h}) \mathtt{PDF}(\boldsymbol{h}) d\boldsymbol{h} \tag{2.39}$$

The goal of Pseudo Spectral Projection is to evaluate the multi-dimensional integral of (2.39) with numerical quadrature techniques (Mathelin, Hussaini, and Zang, 2005; Eldred, Webster, and Constantine, 2008; Jäckel, 2005; Sraj et al., 2017; Gilli et al., 2013), obtaining:

$$c_n = \frac{1}{\mathbb{E}(\Psi_n^2)} \sum_{k=0}^{K-1} Y_{\mathtt{QoI}}(\mathbf{h}^k) \Psi_n(\mathbf{h}^k) w_k , \tag{2.40}$$

where w_k are weights, $\mathbf{h}^k$ are quadrature points, K is the number of evaluations for the model, determined by $(q+1)^{\mathtt{n_d}}$, being q the number of quadrature points and $\mathtt{n_d}$ the number of stochastic inputs ($\mathtt{n_d} = 3$ for the benchmark under consideration). $Y_{\mathtt{QoI}}(\mathbf{h}^k)$ is the QoI evaluated in the quadrature point $\mathbf{h}^k$ and $\Psi_n(\mathbf{h}^k)$ are the basis functions. Finally, $\mathbb{E}(\Psi_n^2)$ can be computed analytically for multivariate polynomials (Le Maıtre et al., 2002; Matthies and Keese, 2005).

Point Collocation Polynomial Chaos

Point collocation Polynomial Chaos, is another non-intrusive technique to calculate the Fourier coefficients c_n for the equation 2.35 (Hosder, Walters, and Balch, 2007; Berveiller, Sudret, and Lemaire, 2004). The goal of this method is to extend the polynomial chaos expansions to be equal to each black box evaluation $Y_{\mathtt{QoI}}(\mathbf{h}^k)$ at a set of "collocation" sampling points $\mathbf{h}^k$. Hence, the polynomial chaos reads

$$Y_{\mathtt{QoI}}(\mathbf{h}^k) = \sum_{n=0}^{\mathtt{n_{PC}}} c_n \cdot \Psi_n(\mathbf{h}^k), k = 0, 1, .., K. \tag{2.41}$$

The corresponding linear system reads:

$$\underbrace{\begin{pmatrix} \Psi_0(\mathbf{h}^0) & \dots & \Psi_{\mathtt{n_{PC}}}(\mathbf{h}^0) \\ \Psi_0(\mathbf{h}^1) & \dots & \Psi_{\mathtt{n_{PC}}}(\mathbf{h}^1) \\ \vdots & \dots & \vdots \\ \Psi_0(\mathbf{h}^K) & \dots & \Psi_{\mathtt{n_{PC}}}(\mathbf{h}^K) \end{pmatrix}}_{\mathbf{M}} \cdot \underbrace{\begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{\mathtt{n_{PC}}} \end{pmatrix}}_{\mathbf{c}} = \underbrace{\begin{pmatrix} Y_{\mathtt{QoI}}(\mathbf{h}^0) \\ Y_{\mathtt{QoI}}(\mathbf{h}^1) \\ \vdots \\ Y_{\mathtt{QoI}}(\mathbf{h}^K) \end{pmatrix}}_{\mathbf{z}}$$

This establishes a system of K equations and $\mathtt{n_{PC}}$ unknowns, where generally $K \geq \mathtt{n_{PC}}$, being therefore an overdetermined system. The least-square minimisation approach consists in finding a set of coefficients which minimises the mean square error, obtaining the solution

$$\mathbf{c} = (\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\mathbf{z}. \tag{2.42}$$

The choice of the collocation points $\mathbf{h}^k$ highly influences computational cost and also the accuracy of the results. Various sampling methods to define $\mathbf{h}^k$ are proposed in the literature such as: Pseudo-Random values, Latin hypercube, Hammersley samples, Halton sequences, and Sobol sequences among others (Hammersley, 1960; Rifkin and Lippert, 2007; Feinberg and Langtangen, 2015; Wong, Luk, and Heng, 1997).

Post processing polynomial chaos

From the orthonormal of the basis functions, it is easily computed the mean and standard deviation of a polynomial chaos expansion using the coefficients c_n. The mean of the random solution is given by

$$\mathbb{E}[Y_{\mathtt{QoI}}] = \mathbb{E}\left[\sum_{n=0}^{\mathtt{n_{PC}}} c_n \cdot \Psi_n(\boldsymbol{h})\right] = c_0, \tag{2.43}$$

which indicates that the zero coefficient of the expansion corresponds to the expected value. Similarly, the variance reads,

$$\sigma^2 = Var[Y_{\mathtt{QoI}}] = \mathbb{E}\left[(Y_{\mathtt{QoI}} - \mathbb{E}(Y_{\mathtt{QoI}}))^2\right] = \left\langle \left(\sum_{i=0}^{\mathtt{n_{PC}}} c_n \mathit{\Psi}_i(\boldsymbol{h}) - c_0\right)^2 \right\rangle = \sum_{i=1}^{\mathtt{n_{PC}}} c_i^2 \left\langle \mathit{\Psi}_i^2(\boldsymbol{h})\right\rangle . \tag{2.44}$$

Additionally, the PC expansion can be used as a response surface for UQ. Thus, the PDF of $Y_{\mathtt{QoI}}$ is obtained by evaluating the PC expansion with new random points $\boldsymbol{h}$ (Sudret and Mai, 2015). A framework scheme of the non-intrusive Polynomial Chaos procedure is shown in Fig. 2.3.

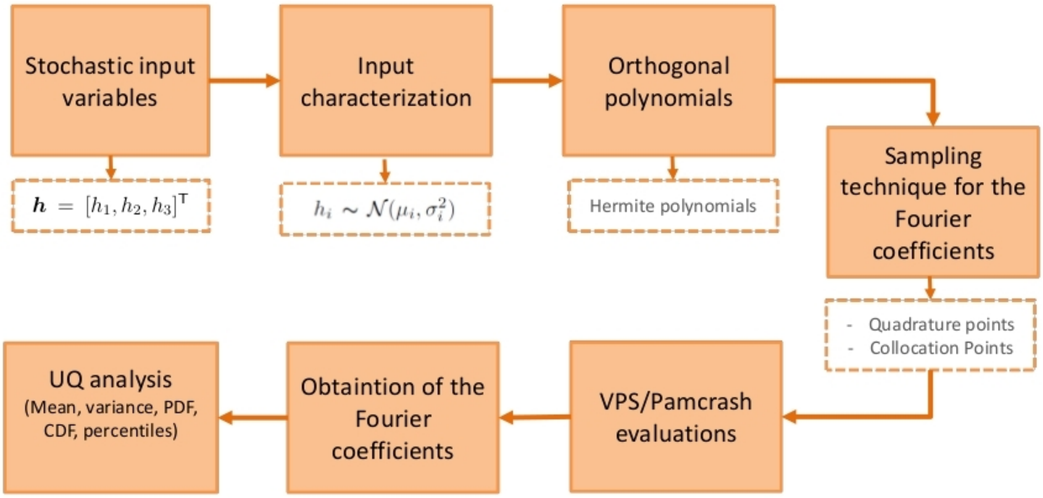

Stochastic input variables
$\boldsymbol{h} = [h_1, h_2, h_3]^{\mathsf{T}}$
Input characterization
$h_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$
Orthogonal polynomials
Hermite polynomials
Sampling technique for the Fourier coefficients
- Quadrature points
- Collocation Points
VPS/Pamcrash evaluations
Obtaintion of the Fourier coefficients
UQ analysis
(Mean, variance, PDF, CDF, percentiles)

FIGURE 2.3 Non-intrusive Polynomial Chaos framework

2.3.2 Monte Carlo Method

The Monte Carlo (MC) method is a well-know technique for propagating the uncertainty in complex systems (Wasserstein, 1997). This probabilistic technique generates a finite number of random samples $\boldsymbol{h}^i, i = 1, 2, ..., \mathtt{n_s}$ to propagate the uncertainty by evaluating the model in each sample point $\boldsymbol{h}^i$. Fig. 2.4 and 2.5 show the main idea and the steps of MC methodology for the B-pillar crash problem.

The input space is typically parametrized by a large number of random samples ($\mathtt{n_{MC}}$=$\mathtt{n_s}$), leading to a high dimensional problem. The expected value

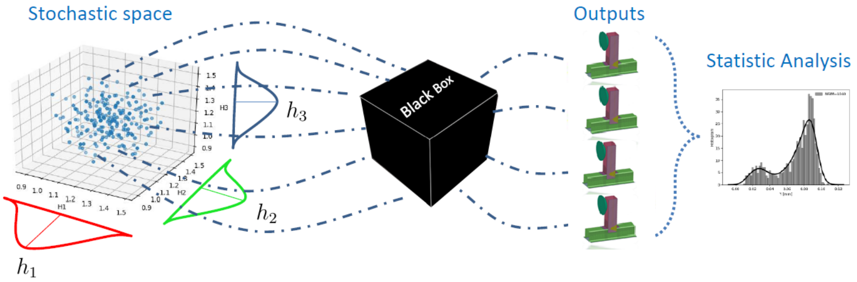


FIGURE 2.4 Monte Carlo crash problem scheme.

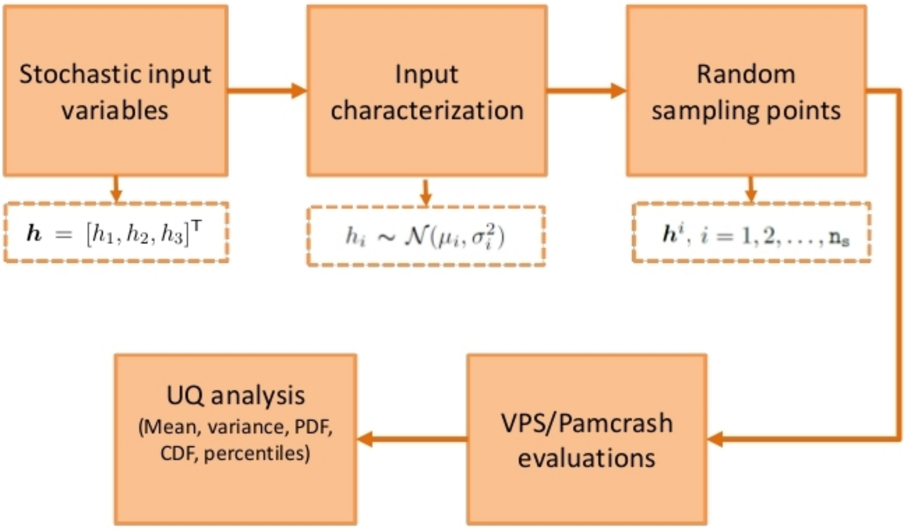

Stochastic input variables
Input characterization
Random sampling points
$\boldsymbol{h} = [h_1, h_2, h_3]^\mathsf{T}$
$h_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$
$\boldsymbol{h}^i,\ i = 1, 2, \ldots, \mathtt{n_s}$
UQ analysis
(Mean, variance, PDF, CDF, percentiles)
VPS/Pamcrash evaluations

FIGURE 2.5 Monte Carlo method framework

$\mathbb{E}[Y_{\mathtt{QoI}}]$ of the training set is defined by:

$$Y_{\mathtt{MC}} = \mathbb{E}[Y_{\mathtt{QoI}}] = \frac{1}{\mathtt{n}_{\mathtt{MC}}} \sum_{i=1}^{\mathtt{n}_{\mathtt{MC}}} Y_{\mathtt{QoI}}(\boldsymbol{h}^i), \tag{2.45}$$

where $\boldsymbol{h}^i$ corresponds to an input sample realization, and $\mathtt{n}_{\mathtt{MC}}$ the number of Monte Carlo evaluations. Given the expected value $Y_{\mathtt{MC}}$, the variance (σ^2) and the standard deviation (σ) are determined by,

$$\sigma^2[Y_{\mathtt{QoI}}] = \frac{1}{\mathtt{n}_{\mathtt{MC}}} \sum_{i=1}^{\mathtt{n}_{\mathtt{MC}}} (Y_{\mathtt{QoI}}(\boldsymbol{h}^i) - Y_{\mathtt{MC}})^2, \tag{2.46}$$

$$\sigma = \sqrt{\sigma^2}. \tag{2.47}$$

It is important to empathize that the Standard Deviation (StD) is measured in the same units as the training set and the variance is measured in squared units.

The main disadvantage of MC method is its slow convergence rate, determined by the order $\frac{1}{\sqrt{\mathtt{n}_{\mathtt{MC}}}}$. In crashworthiness analysis, each simulation requires a large time to be solved, consequently, the required computational cost to implement this stochastic technique is generally unaffordable. In such cases, different variants from MC can be implemented based in smart sampling and variance reduction techniques.

2.3.3 Quasi Monte Carlo Method

Quasi Monte Carlo (QMC) method is a variant of the MC technique based on a reduction of the model evaluations. Thus, the input parameters $\boldsymbol{h}^i$ are discretized for $i = 1, 2, .., \mathtt{n}_{\mathtt{QMC}}$, where the number of samples are reduced with respect to MC ($\mathtt{n}_{\mathtt{QMC}} < \mathtt{n}_{\mathtt{MC}}$). QMC is based in smart sampling where the convergence rate is improved with respect to the classical MC, close to $\frac{1}{\mathtt{n}_{\mathtt{QMC}}}$ (Graham, Parkinson, and Scheichl, 2018).

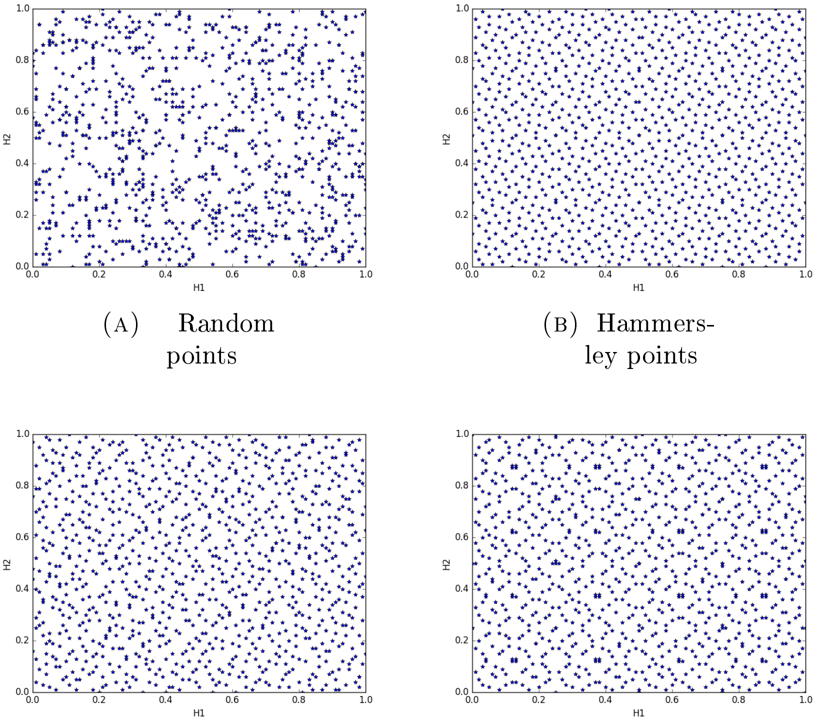
The choice of QMC points are based on low discrepancy sequences, also called quasi-random or sub-random sequences (Niederreiter, 1978). Sub-random numbers have an advantage over MC random points, they cover the domain of interest quickly and evenly. Fig. 2.6 shows a comparative plot with three schemes to generate QMC points in a uniform 2D space. It illustrates: random points, Hammersley, Halton and Sobol sequences (Feinberg and Langtangen, 2015; Hosder, Walters, and Balch, 2007). Fig. 2.7 shows the QMC scheme procedure. Analogously to MC, the QMC expected value of the QoI is given by

$$Y_{\mathtt{QMC}} = \frac{1}{\mathtt{n}_{\mathtt{QMC}}} \sum_{i=1}^{\mathtt{n}_{\mathtt{QMC}}} Y_{\mathtt{QoI}}(\tilde{\boldsymbol{h}}^i), \tag{2.48}$$

where $\tilde{\boldsymbol{h}}^i$ represents the input parameters discretized with a discrepancy sequence technique.

2.3.4 Multi level Monte Carlo Method

Multilevel Monte Carlo (MLMC) is a variant of MC that has been implemented in different fields (Graham, Parkinson, and Scheichl, 2018; Barth, Schwab, and Zollinger, 2011). The method is based in a hierarchy numerical approach for different levels of accuracy. For each level, the model becomes progressively accurate with more computational cost. The strategy consist in evaluate a



(A) Random points

(B) Hammersley points

(C) Halton points

(D) Sobol points

FIGURE 2.6 Comparison of various quasi random sequences (B),(C),(D) with respect to random points (A).

large number of simulations for a low computational model (giving a sense of average behaviour) and few number of simulations for a high accurate model (giving a sense of precision) (Aslett, Nagapetyan, and Vollmer, 2017). This hierarchy method generally provides better results for models that the mesh or the time step can be changed easily for each level to create a training set of models. If the problem allows this condition, then the computational cost is drastically improved with respect to MC.

Here the quantity of interest $Y_{\texttt{QoI}}$ is approximated for a sequence of hierarchical levels $Y^0_{\texttt{QoI}}, ..., Y^l_{\texttt{QoI}}$ with different accuracies. Being $Y^0_{\texttt{QoI}}$ the less accurate level, thus requiring less computing cost. On the contrary, $Y^\ell_{\texttt{QoI}}$ is the most accurate and therefore computationally costly (Giles, 2008). The sense of the multilevel method is based in the telescoping sum defined by

$$\mathbb{E}[Y_{\texttt{MLMC}}] = \mathbb{E}[Y^0_{\texttt{QoI}}] + \sum_{\ell=1}^{L} \mathbb{E}[Y^\ell_{\texttt{QoI}} - Y^{\ell-1}_{\texttt{QoI}}]. \tag{2.49}$$

Where the first term $\mathbb{E}[Y^0_{\texttt{QoI}}]$ it is a low estimator of the QoI and the terms $\sum_{\ell=1}^{L} \mathbb{E}[Y^\ell_{\texttt{QoI}} - Y^{\ell-1}_{\texttt{QoI}}]$ improves the accuracy. Subtituying the MC equation 2.45

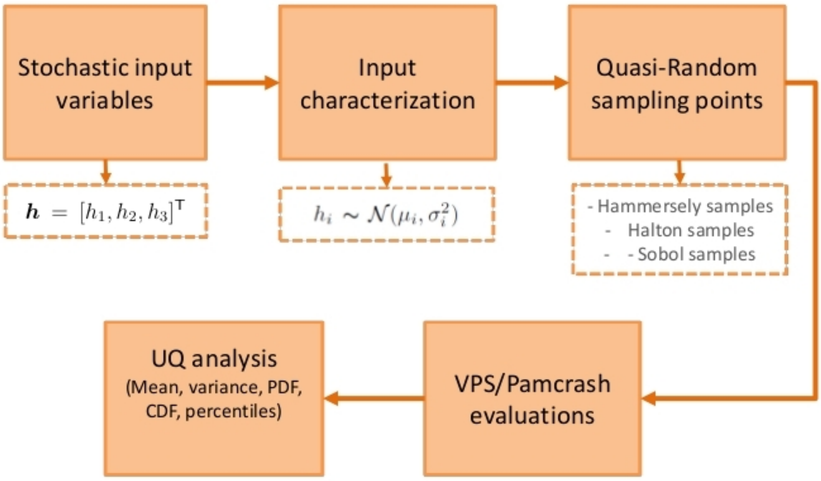

Stochastic input variables
$\boldsymbol{h} = [h_1, h_2, h_3]^{\mathsf{T}}$
Input characterization
$h_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$
Quasi-Random sampling points
- Hammersely samples
- Halton samples
- - Sobol samples
VPS/Pamcrash evaluations
UQ analysis
(Mean, variance, PDF, CDF, percentiles)

FIGURE 2.7 Quasi Monte Carlo method framework

into 2.49 for each level leads to

$$\begin{aligned}\mathbb{E}[Y_{\texttt{MLMC}}] = \frac{1}{\texttt{n}_0} \sum_{n=1}^{\texttt{n}_0} Y^0_{\texttt{QoI}}(\boldsymbol{h}^n) + \\ \sum_{\ell=1}^{L} \left[\frac{1}{\texttt{n}_\ell} \sum_{n=1}^{\texttt{n}_\ell} (Y^\ell_{\texttt{QoI}}(\boldsymbol{h}^n) - Y^{\ell-1}_{\texttt{QoI}}(\boldsymbol{h}^n)) \right],\end{aligned} \tag{2.50}$$

where $\texttt{n}_0$ is the number of simulations in level 0 and $\texttt{n}_\ell$ is the number of simulations in level ℓ, where $\texttt{n}_0 < \texttt{n}_1 < \cdots < \texttt{n}_{\ell-1} < \texttt{n}_\ell$. The different levels of accuracy for each model basically are obtained by two different manners: (i) increasing the time step, (ii) refining the mesh grid. In crashworthiness analysis, refining the mesh can be unaffordable, thus a better option is to increase the time step, keeping the mesh fixed. Fig. 2.8 illustrates the MLMC scheme of a 2D geometry with a telescoping increasing mesh. Analogously to MC and QMC, Fig. 2.9 shows the framework of the main steps to implement MLMC method.

2.3.5 Taguchi Method

Design of Experiments (DoE) is based on the implementation of sampling strategies using a specific number of simulations. One of the first DoE approaches is the Taguchi method, based on studying the variability of the input with a small number of simulations. Taguchi is applied in several fields such as automotive

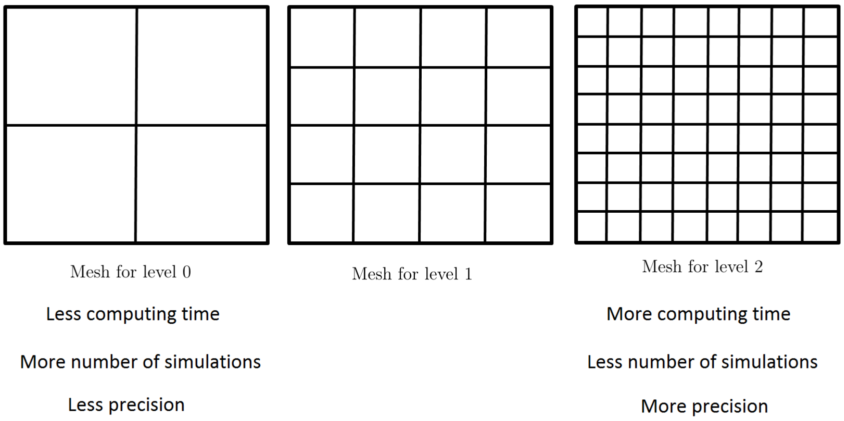


FIGURE 2.8 Schematic example of a Multi Level Monte Carlo telescoping strategy using different mesh grid levels.

engineering, biology, chemistry to evaluate the minimum number of model evaluations (Gopalsamy, Mondal, and Ghosh, 2009; Tsui, 1992; Fratila and Caizar, 2011). The key of this technique is on the use of orthogonal arrays by columns. That is, for any pair of columns, all combinations of input levels occur, and they occur an equal number of times (Taguchi and Konishi, 1987; Zang, Friswell, and Mottershead, 2005; Lin et al., 2005; Al-Momani and Rawabdeh, 2008). Table 2.2 shows an example of parameters combination using Taguchi method. It consists on a total 9 simulations to be conducted with three parameters (in our case they would be the thicknesses h_1, h_2, h_3), each one of them discretized with three values:

$$h_1 = [h_1^1, h_1^2, h_1^3]$$

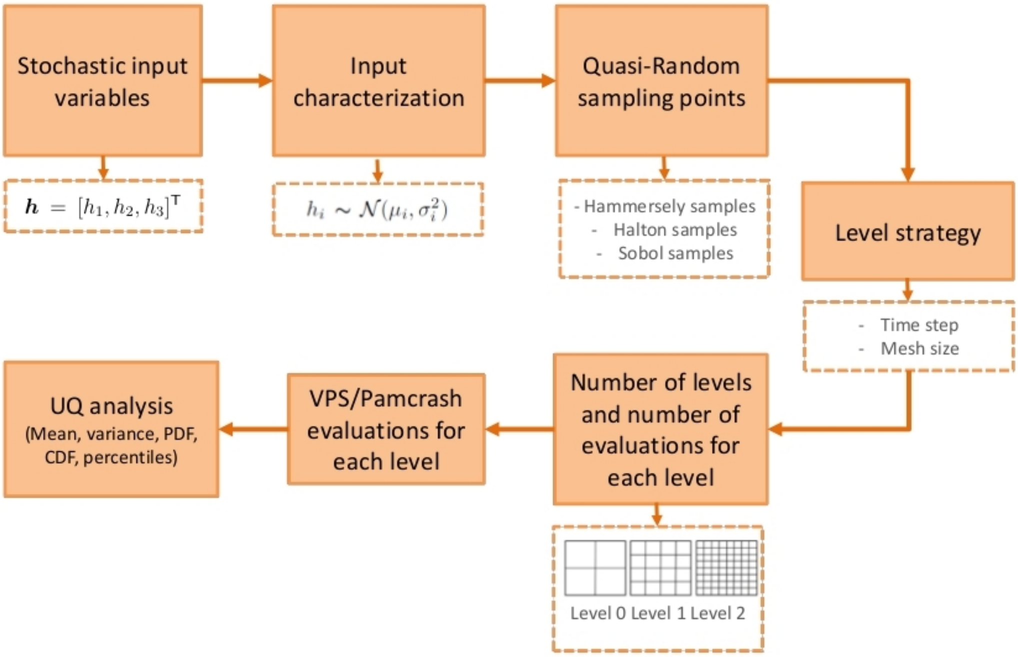
$$h_2 = [h_2^1, h_2^2, h_2^3]$$

$$h_3 = [h_3^1, h_3^2, h_3^3]$$

For more specific details on Taguchi method and orthogonality properties, the reader is refereed to (Taguchi and Konishi, 1987).

The main steps used in Taguchi methodology (Fei, Mehat, and Kamaruddin, 2013; Roy, 2001) are:

- Select the random variables.
- Select of number of samples for each variable.
- Construct the orthogonal array.
- Conduct the simulations with respect to the array.


Stochastic input variables

$\boldsymbol{h} = [h_1, h_2, h_3]^{\mathsf{T}}$

Input characterization

$h_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$

Quasi-Random sampling points

- Hammersely samples
- Halton samples
- Sobol samples

Level strategy

- Time step
- Mesh size

Number of levels and number of evaluations for each level

VPS/Pamcrash evaluations for each level

UQ analysis (Mean, variance, PDF, CDF, percentiles)

Figure 2.9 Multi Level Monte Carlo framework

Simulation number	Input h_1	Input h_2	Input h_3
1	h_1^1	h_2^1	h_3^3
2	h_1^1	h_2^2	h_3^2
3	h_1^1	h_2^3	h_3^1
4	h_1^2	h_2^1	h_3^2
5	h_1^2	h_2^2	h_3^1
6	h_1^2	h_2^3	h_3^3
7	h_1^3	h_2^1	h_3^1
8	h_1^3	h_2^2	h_3^3
9	h_1^3	h_2^3	h_3^2

Table 2.2 Orthogonal Array of combination inputs

- Analyse data results.

To post process the outcomes from Taguchi method, a signal-to-noise ratio (Atkinson, Donev, and Tobias, 2007) and an analysis of variance is commonly used to calculate and improve the variability of the samples (Fei, Mehat, and Kamaruddin, 2013; Datta, Bandyopadhyay, and Pal, 2008; Taguchi and Konishi, 1987).

2.4 Comparison of SFEM solvers for crashworthiness

In this section, it is shown a comparative analysis between intrusive and non-intrusive sampling methods to implement the most suitable UQ approaches for the crashworthiness benchmark problem explained in section 2.1.

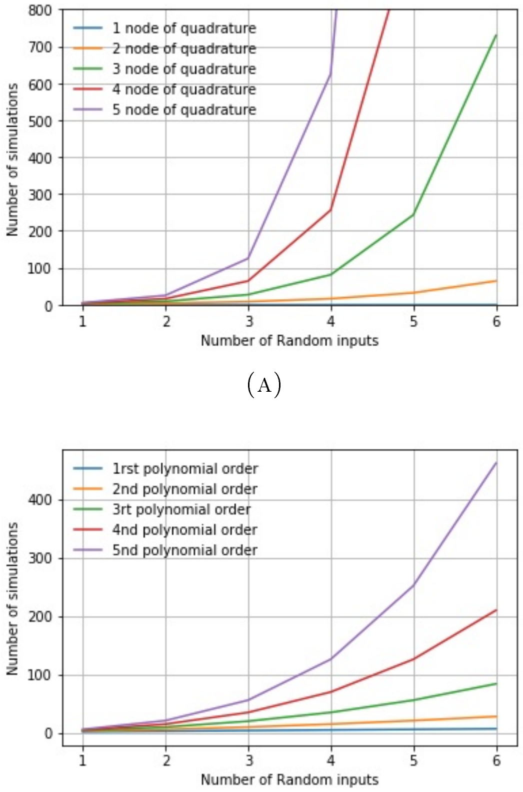
Intrusive solvers such as Perturbation method, Galerkin Polynomial Chaos and Spectral Stochastic Finite Element Method described in above sections have important strengths to take into account. Intrusive techniques give the full random responses in the whole time and space domains. Also, in general cases it requires fewer simulation compared with non-intrusive approaches. However, these methodologies are not trivial to be implemented or even not possible (Eiermann, Ernst, and Ullmann, 2007). Furthermore, computational complexity increases with the number of random inputs and the order of the expansion (Arregui-Mena, Margetts, and Mummery, 2016). In the case where the QoI have a non-linear response, high expansion orders are required and instabilities can jeopardize the problem (Bergman et al., 1997).

On the other side, the main advantage of non intrusive methods, Pseudo Spectral Polynomial Chaos, Point Collocation Polynomial Chaos, Monte Carlo and variants (non intrusive methods), is the use of commercial softwares to obtain deterministic outputs without inferring in the source code (Eiermann, Ernst, and Ullmann, 2007).

MC technique is considered the most general and robust method of uncertainty quantification. For that reason, it is used as a reference method for comparison purposes. This approach allows to tackle linear and nonlinear problems in a wide range of engineering and science fields (Sudret and Der Kiureghian, 2000; Sudret, 2008). One main drawback lays on the large number of simulations necessary to obtain good results, which in crashworthiness means unaffordable computational cost. QMC and MLMC present improvements to reduce dimensionality, thus in cases where simulations require a high computing power, its implementation is recommended (Graham, Parkinson, and Scheichl, 2018).

Non intrusive PC present advantages with respect to MC methods in terms of dimensionality, since it allows a reduction of the number of simulations. Unlike MC approach, non intrusive PC expansions suffer from the curse of dimensionality, the number of terms grow exponentially with the number of random inputs (Xiu, 2009; Stefanou, 2009) and the polynomial degree. In pseudo spectral projection PC, the number of simulations increases by $N_{\mathrm{psp}} = (q)^{\mathtt{n_d}}$, where q is the number of quadrature nodes and $\mathtt{n_d}$ the number of random variables. For

Point Collocation PC the number of simulations is defined by $N_{\mathrm{pc}} = n_f \frac{(\mathrm{n_d}+p)!}{\mathrm{n_d}!p!}$, being p the order of the polynomial expansion and n_f a proportional parameter that increases the number of collocation points, thus oversampling the number of simulations. If the parameter $n_f > 1$ the system is overdetermined and least squares technique is used to reduce the residual error of the response. This proportional parameter is used to improve the response accuracy. In the literature, different authors consider indispensable oversampling the training set of simulations for a good performance (Hadigol and Doostan, 2018; Eldred, 2009). Fig. 2.10 shows the computational cost associated to Pseudo Spectral Projection and Point Collocation. It is clearly visible the lower computational cost of Point Collocation with respect to Pseudo Spectral Projection with large number of stochastic inputs (Feinberg and Langtangen, 2015). However, with low number of random inputs the cost is similar.



(A)

(B)

FIGURE 2.10 Number of black box simulations using (A) Pseudo Spectral Projection Polynomial Chaos, and (B) Point Collocation Polynomial Chaos with $n_f = 1$.

To conclude, for one or two random variables, the number of simulations for both methods have similar computational cost. However, when the number of random variables increases, Point Collocation is a better option for uncertainty quantification (Stefanou, 2009).

2.5 Benchmark B-Pillar results

In this section it is presented the results of the benchmark problem described in section 2.1.

The benchmark model is solved with a deterministic approach as a starting point. It is considered deterministic thicknesses $h_1 = h_2 = h_3 = 1.2$ mm. In Fig. 2.11 it is shown the snapshots of the local plastic strain evolving in time for the explicit simulation. The response for the QoI (plastic strain average on the the area of interest in the last step of time) through this deterministic approach corresponds to $Y_{\texttt{QoI}} = 0.0798$.

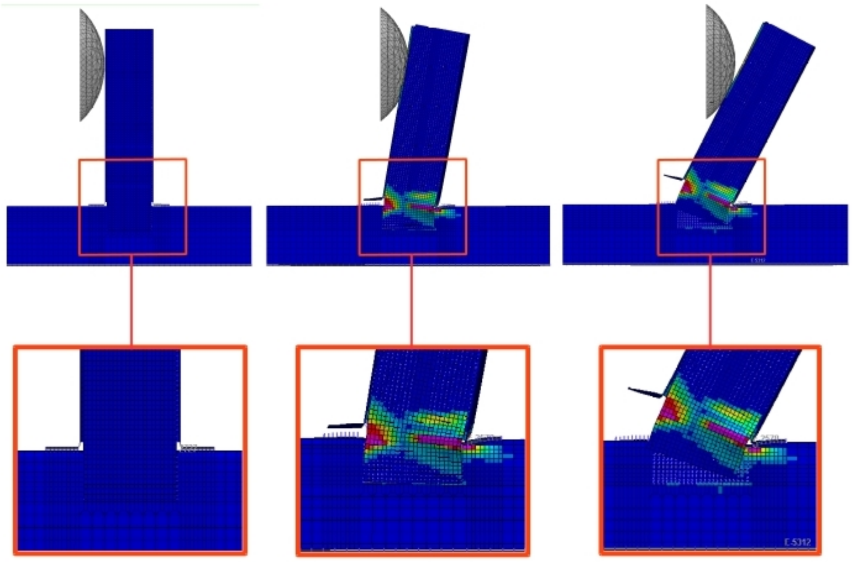


FIGURE 2.11 Snapshots of the model response (0s, 0.5s and 1s from left to right).

In the next section the B-Pillar problem is considered stochastic with random thicknesses.

2.5.1 Framework

The benchmark crash problem is analyzed with different non-intrusive UQ methods. The deterministic evaluations of the full order model are conducted with the commercial solver VPS/Pamcrash 2015.4. Intrusive methods have been avoided for the need to reformulate the source code of Pamcrash.

As aforementioned, the computational time is mainly the most critical drawback in crashworthiness studies. Thus, the non intrusive stochastic techniques, PC and QMC, are implemented and compared with the MC results.

A MC vademecum of $\mathtt{n}_{\mathtt{MC}} = 2466$ simulations is performed for the training set. MC is used as a reference methodology for comparison purpose. Computing the solutions of the final training set with 2466 samples used here required around 822 hours (approximately 35 days) of computational time in one of the SEAT clusters. Besides, the computational time required for a standard full crash model is around one day per simulation. Thus, any effort in devising strategies to build a reliable training set with the minimum number of full-order solutions is worthwhile. Therefore, Quasi Monte Carlo and Point Collocation Polynomial Chaos are implemented with Hammersley sampling (Hammersley, 1960) for the discretization of the input space $\boldsymbol{h}$. Fig. 2.12 illustrates the stochastic space $\boldsymbol{h}$ of the three random thickness using Hammersley technique.

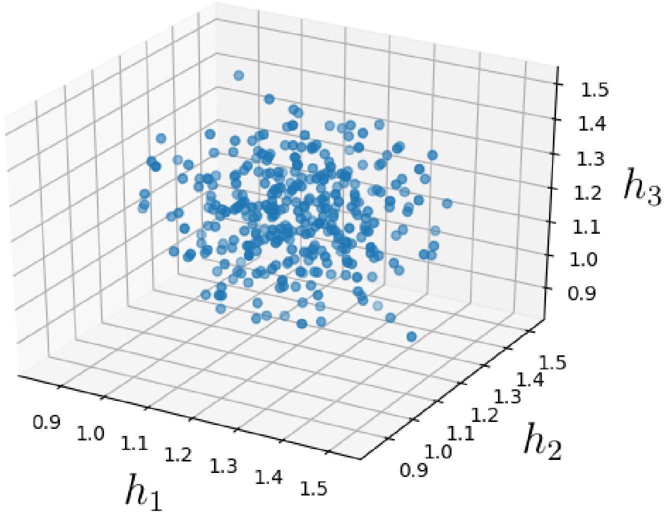


FIGURE 2.12 Stochastic input space of h_1, h_2, h_3 with 200 Hammersley sample points.

For the study of Point Collocation PC, different oversampling values n_f and polynomial orders p are conducted. Aiming to obtain the optimal combination of these two parameters involved for the method.

The deterministic black box simulations were launched by parallel computing with two different machines: i) A cluster with 16 CPUs of 3.40 GHz and

252.3 GiB of RAM memory. ii) A Workstation laptop with 4 CPUs of 3.40 Ghz and 32 GiB of RAM. The computational cost for a single simulation was in the order of 20 minutes, whereas all the set of simulations are in the order of weeks.

2.5.2 Monte Carlo

The Monte Carlo method is implemented as a reference approach to obtain a vademecum of the QoI. In Fig. 2.13 the convergence plots of the mean $Y_{\mathtt{MC}}$ and the standard deviation $StD_{\mathtt{MC}}$ with respect the number of simulations $\mathtt{n_s}$ are illustrated. In Table 2.3 it is represented the statistical values up to $\mathtt{n_s} = 2466$ simulations. The MC approach convergence to the values of $Y_{\mathtt{MC}} = 0.0695$ and $StD_{\mathtt{MC}} = 0.0239$, which will be considered as reference results for further analysis. Accordingly, the PDF evolution is illustrated in Fig. 2.14 with different sampling size $\mathtt{n_s}$. A clear bimodal behaviour is observed. The *Mode 2* shows a predominant area of probability with respect to *Mode 1*. This phenomenon of multi modality usually occurs when small perturbations in the system cause some changes on the output. The prediction of different modes give an important key to understand the nature of the problem.

$\mathtt{n_s}$	1000	1500	2066	2466
$Y_{\mathtt{MC}}$	0.0697	0.0699	0.0695	0.0695
$StD_{\mathtt{MC}}$	0.0234	0.0237	0.0240	0.0239

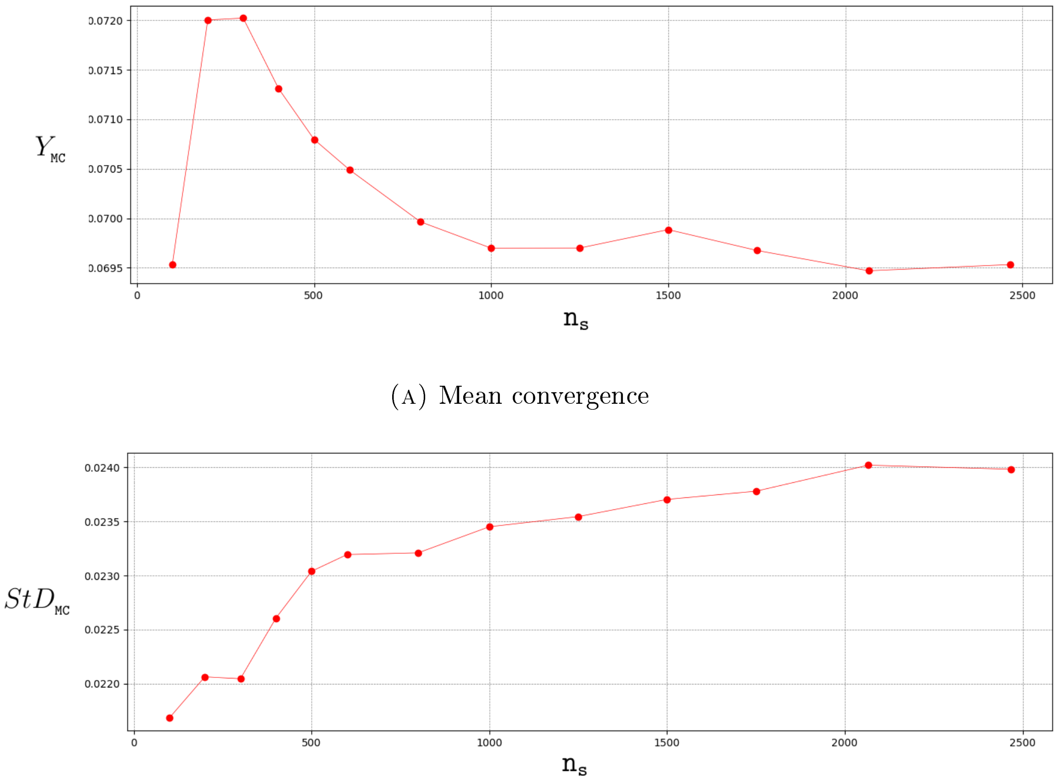
TABLE 2.3 Mean and standard deviation results with MC.

To analyze the bimodality, two simulations are selected from the PDF functions coinciding with each peak in Fig. 2.16. Fig. 2.15 shows the two simulations respectively.

In Table 2.4 it is illustrated the two simulations with its corresponding stochastic inputs $h_1, h_2,\ h_3$. To analyze which parameter or combination of parameters generate the bimodal distribution, in Fig. 2.16 there are plotted the samples that appears at left (red) and right (blue) of the histogram.

	h_1	h_2	h_3	$Y_{\mathtt{QoI}}$
Simulation (A)	1.31	1.47	1.37	0.0294
Simulation (B)	1.22	1.08	1.12	0.0852

TABLE 2.4 Stochastic inputs h_1, h_2, h_3 and $Y_{\mathtt{QoI}}$ for each peak of the PDF.



(A) Mean convergence

(B) Standard deviation convergence

FIGURE 2.13 Convergence plots of the expected value $Y_{\texttt{MC}}$ and standard deviation $StD_{\texttt{MC}}$ with respect to the number of evalutions $\texttt{n}_\texttt{s}$

Fig. 2.17 shows the stochastic space of the thickness inputs $\boldsymbol{h}$ for each simulation according to the previous bimodal color condition. It is observed two differentiated domains regarding the stochastic thickness h_3. It seems to be an evidence that parameter h_3 has a direct correlation with the QoI. However, h_1 and h_2 from Fig. 2.17c shows a mixed distribution of the samples in all the stochastic space, therefore a not clear influence with the output is here visible.

2.5.3 Quasi Monte Carlo

In this section QMC is implemented with the Hammersley technique for the input discretization. Fig. 2.18 shows the mean and standard deviation convergence plots with respect to the reference values from the MC approach (red line). Similar results with respect to MC were obtained with 330 simulations (recall that for MC a total of 2466 simulations were used), leading to a QMC values $Y_{\texttt{QMC}} = 0.06944$ and a $StD_{\texttt{QMC}} = 0.02391$.

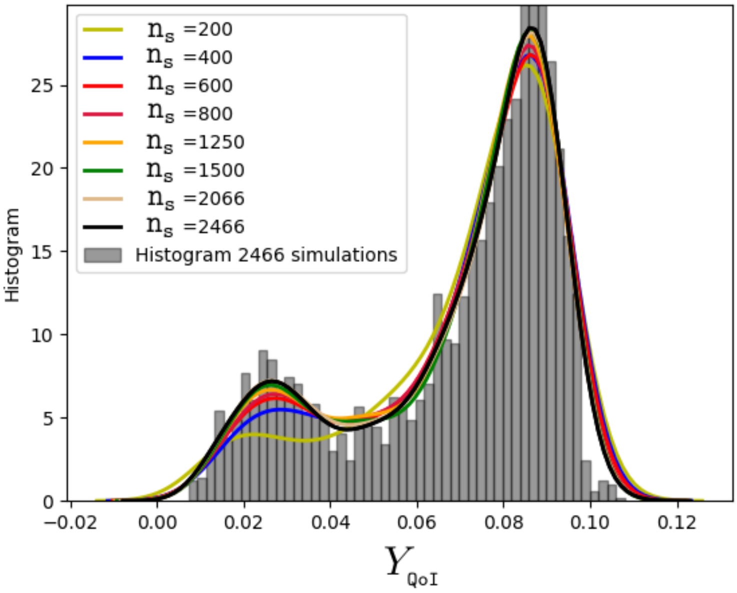
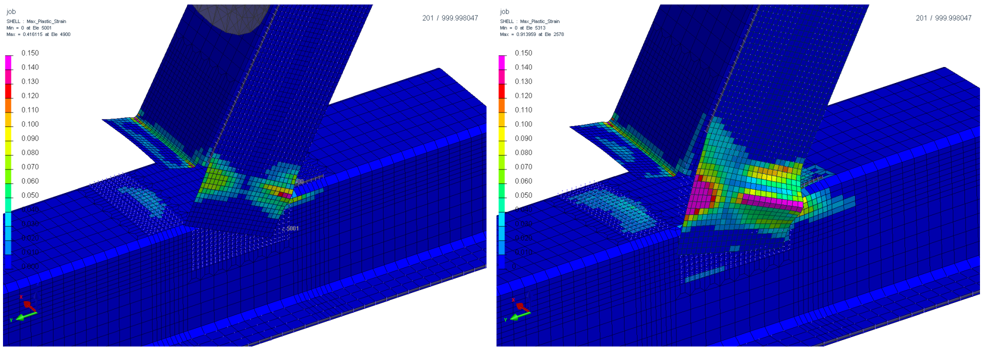


FIGURE 2.14 Probability density functions for the QoI with different sampling size $\mathtt{n_s}$ for the training set.



(A) $Y_{\mathtt{QoI}} = 0.0294$

(B) $Y_{\mathtt{QoI}} = 0.0852$

FIGURE 2.15 (A) Simulation sample corresponding to peak 1 and (B) simulation sample corresponding to peak 2.

2.5.4 Non intrusive Polynomial Chaos

In this section the results and capabilities of the Point Collocation Polynomial Chaos are shown for different combinations of the polynomial order p and the oversampling parameter n_f. Fig. 2.19, 2.20, 2.21 and 2.22 show the results for the different cases. It is illustrated the probability density functions for MC (blue), QMC (green) and PC (red). For each case, there are evaluated the same number of VPS/Pamcrash simulations for QMC and PC for comparison purposes.

According to these illustrations, using polynomial orders higher than four,

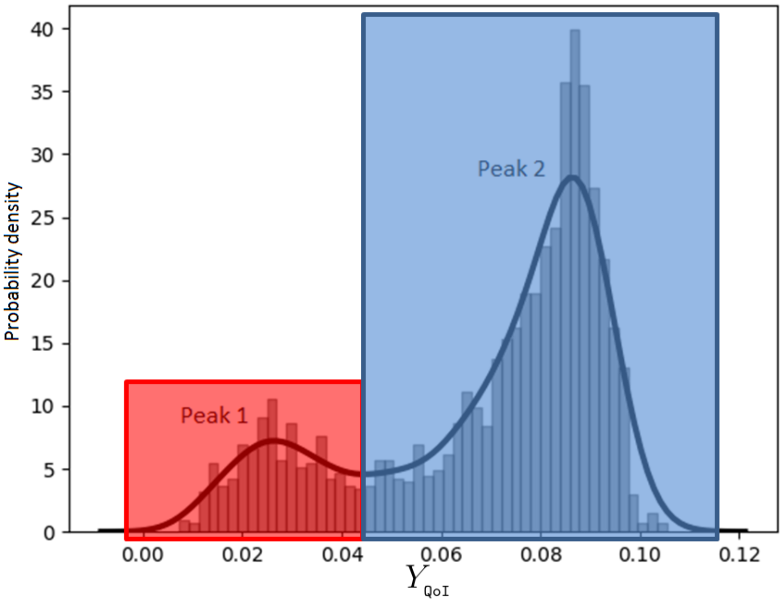
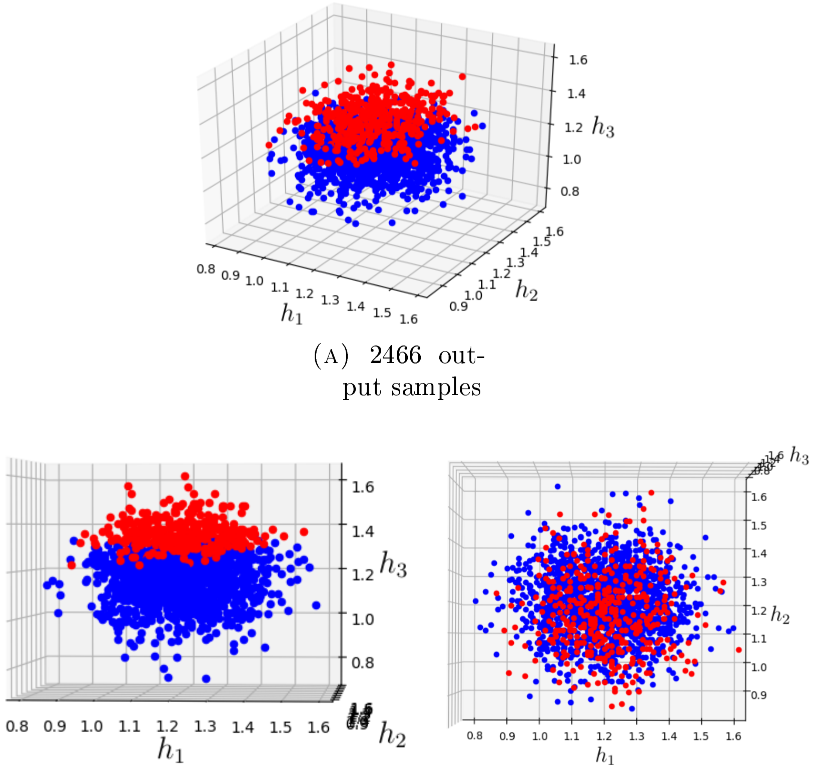


FIGURE 2.16 Probability density function for the QoI with two coloured areas corresponding to each structure mode.

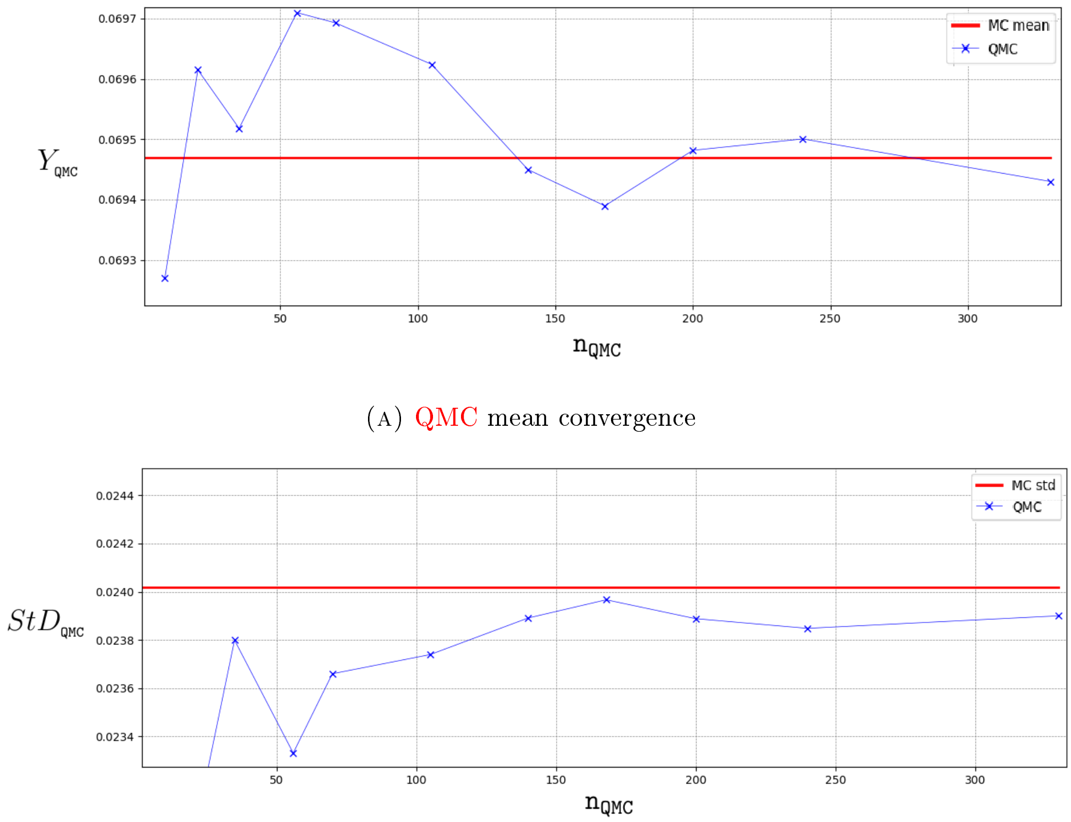


(A) 2466 output samples

(B) Frontal view

(C) Vertical view

FIGURE 2.17 Scattered plots of the input space with respect to the bimodal behaviour. Red samples (Mode 1), blue samples (Mode 2).



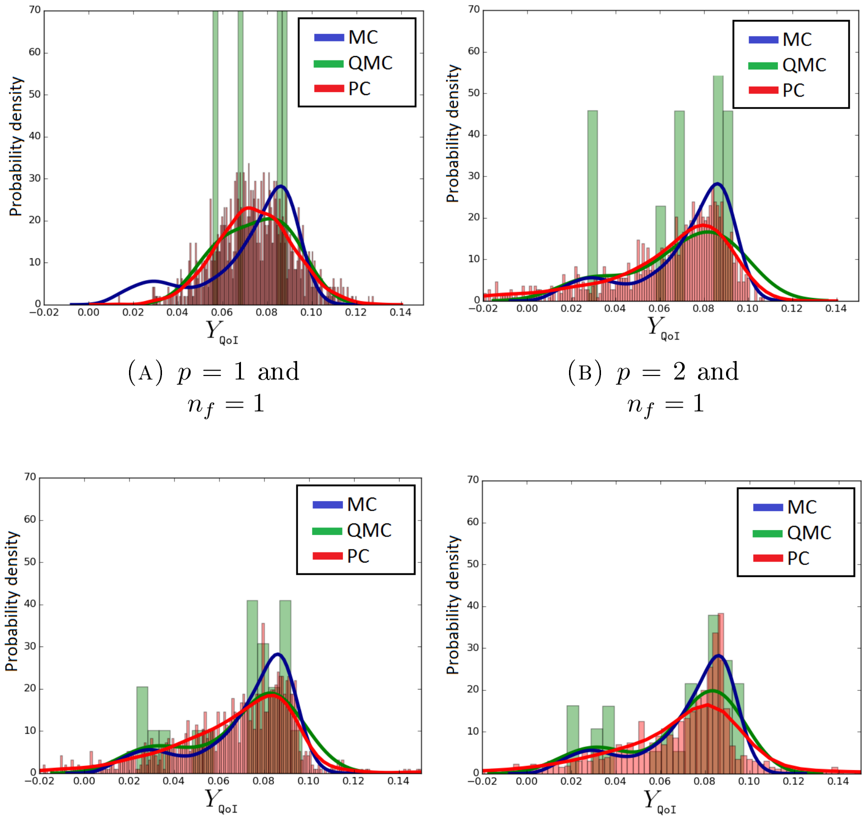
(A) QMC mean convergence

(B) QMC standard deviation convergence

FIGURE 2.18 Quasi Monte Carlo mean and standard deviation convergence plots for different sampling size $\mathtt{n_{QMC}}$.

the accuracy of the results tends to diverge, apart from the case of $n_f = 2$ (double of training set evaluations with respect the unknowns) with $p = 5$ and $p = 6$ (Fig. 2.23), where the probability density function are better captured. This effect is clearly seen in Fig. 2.24, illustrating the behaviour of the mean and standard deviation, for the different cases described above, increasing the polynomial order.

In terms of the number of evaluations, Fig. 2.25 illustrates the mean and standard deviation of MC, QMC and PC methods with the different samplig size $\mathtt{n_s}$. All configurations of PC show less accurate results with respect to QMC, that maintains a highly stable results with very low number of simulations. Also, when the number $\mathtt{n_s}$ increases it is observed that the results of PC with $n_f = 1$ and $n_f = 2$ shows inaccurate results for any combination of the polynomial order p.



(A) $p = 1$ and $n_f = 1$

(B) $p = 2$ and $n_f = 1$

(C) $p = 3$ and $n_f = 1$

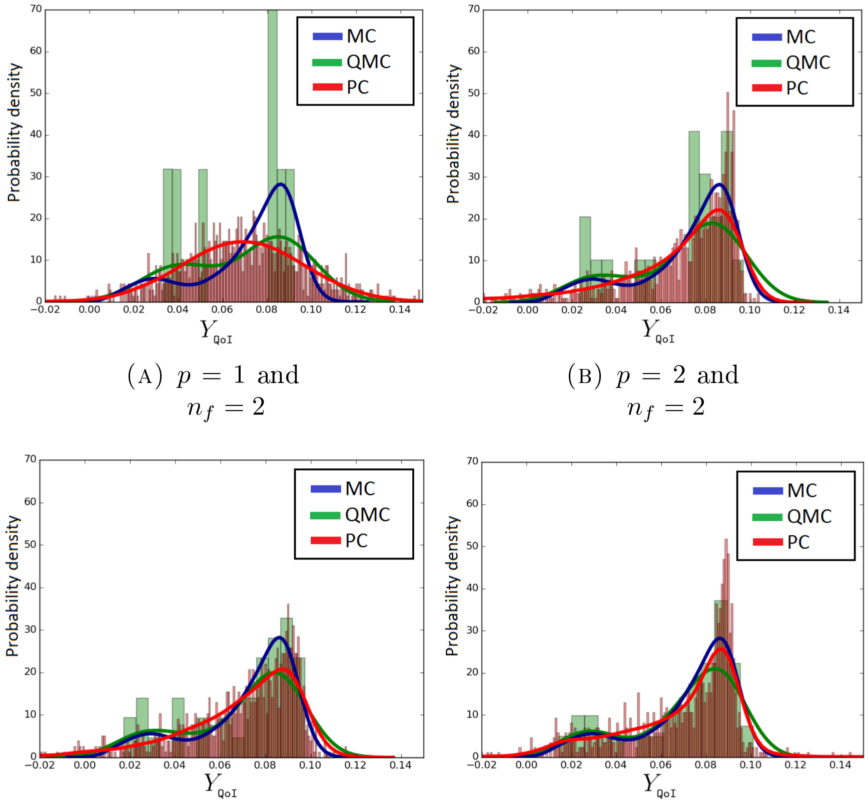
(D) $p = 4$ and $n_f = 1$

FIGURE 2.19 Probability density function evolution of Point Collocation Polynomial Chaos ($n_f = 1$, Hammersley sampling), QMC (Hammersley sampling) and MC (random sampling). Approaches launched with different polynomial order p and $\mathtt{n_{QMC}}$.

2.6 Conclusions

The most suitable SFEM solvers for crashworhtiness UQ analysis were introduced in this chapter. In order to avoid the cumbersome task of an intrusive approach, non-intrusive methods are implemented for the benchmark problem. The VPS/Pamcrash is used as a solver for the training evaluations of the full order model.

The MC results of the stochastic problem offers a sound framework for uncertainty propagation, however its low efficiency precludes its use for analyses involving crash models. In this case, MC is implemented as a reference method to be compared with other techniques. It is detected a bimodal behaviour of the QoI with a high predominant probability area.



(A) $p = 1$ and $n_f = 2$

(B) $p = 2$ and $n_f = 2$

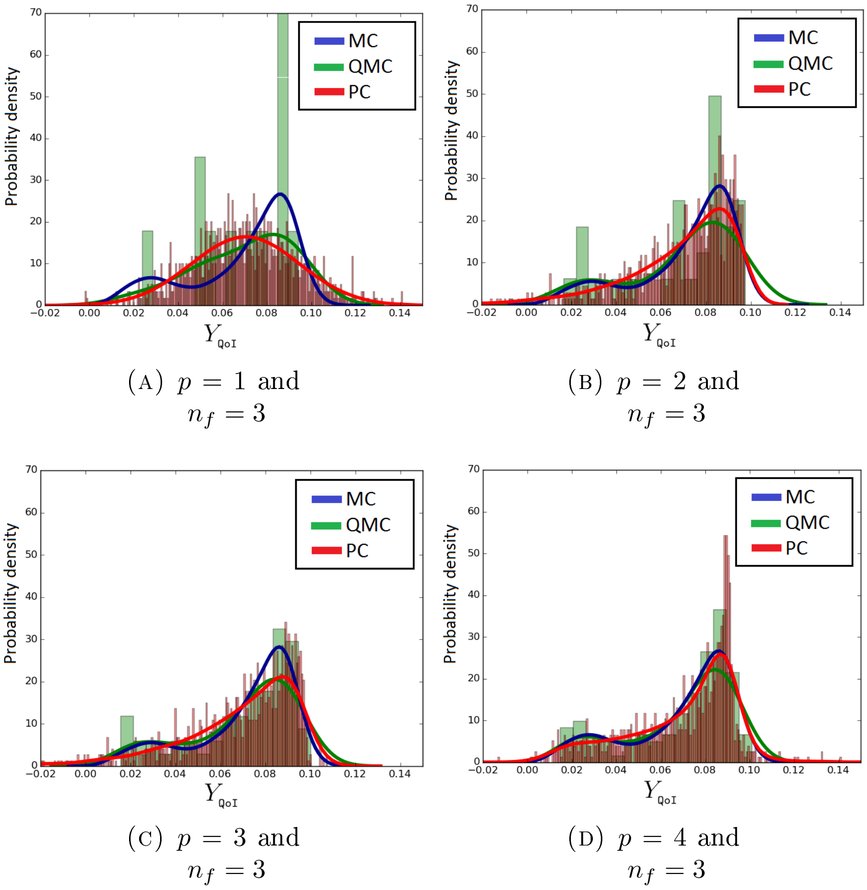
(C) $p = 3$ and $n_f = 2$

(D) $p = 4$ and $n_f = 2$

FIGURE 2.20 Probability density function evolution of Point Collocation Polynomial Chaos ($n_f = 2$, Hammersley sampling), QMC (Hammersley sampling) and MC (random sampling). Approaches launched with different polynomial order p and $\mathtt{n}_{\mathtt{QMC}}$.

To demonstrate the viability of decreasing the number of simulations (training set), two non-intrusive techniques are studied and compared with respect to MC: Quasi Monte Carlo and Point Collocation Polynomial Chaos, both implemented for the crash problem. Compared with the classical MC method, QMC effectively reduces the required number of simulations to achieve a similar accuracy results. The so-called Point Collocation Polynomial Chaos (based on a least-square minimization) shows that in order to have a good accuracy it requires three times greater the number of simulations ($n_f = 3$), leading to an overdetermined problem. The PC results capture worse the bimodal behaviour of the probability density function. However, the highest pick of the PDF shows good precision. In counterpart, the results of QMC illustrates a good performance to capture the bimodal behaviour but the pick with highest probability is worse described.

For the present state-of-the-art review, some weaknesses are observed to



(A) $p = 1$ and $n_f = 3$

(B) $p = 2$ and $n_f = 3$

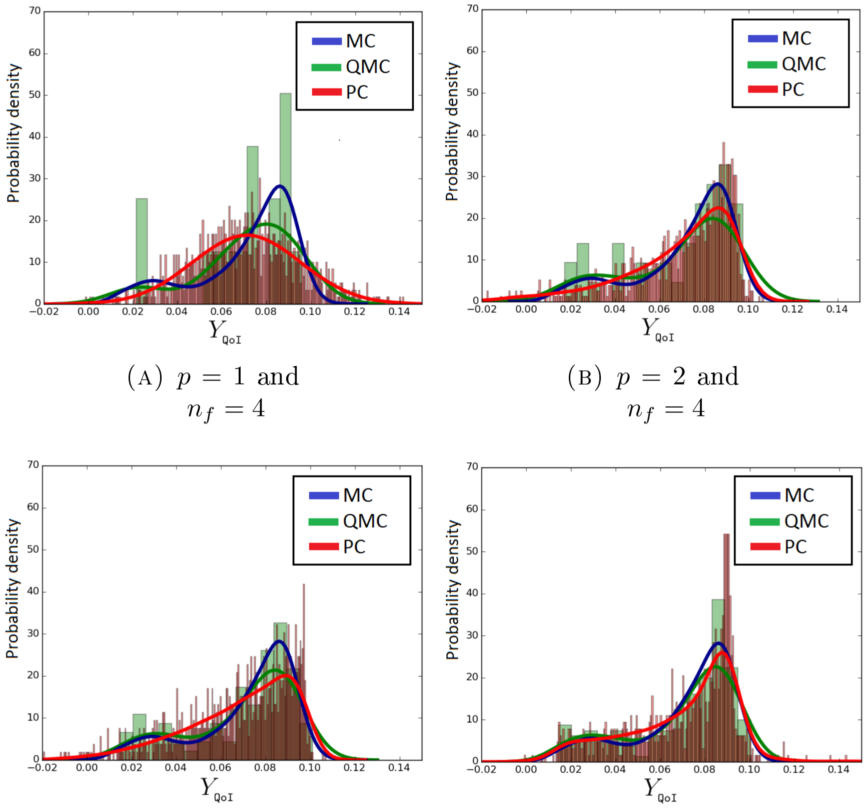
(C) $p = 3$ and $n_f = 3$

(D) $p = 4$ and $n_f = 3$

FIGURE 2.21 Probability density function evolution of Point Collocation Polynomial Chaos ($n_f = 3$, Hammersley sampling), QMC (Hammersley sampling) and MC (random sampling). Approaches launched with different polynomial order p and $\mathtt{n_{QMC}}$.

deal with high dimensional outputs and data analysis. The high dimensional outputs of the model evaluations brings a rich high dimensional dataset $\mathbf{X} = [\mathbf{x}^1\mathbf{x}^2 \cdots \mathbf{x}^{\mathtt{n_s}}] \in \mathbb{R}^{d \times \mathtt{n_s}}$ where each column $\mathbf{x}^i = [x_1^i \dots x_d^i]^\intercal$ is a VPS/Pamcrash solution storing the maximum plastic strain in the last step of time in the area of interest. This leads to a high dimensional problem to post process and metamodel. The state-of-the-art methods present limitations in terms of efficiency, data behaviours (clustering) and sensitivity analysis. However, it has been opened a new research way of possibilities combining stochastic modelling with data analysis techniques (machine learning) (such as dimensionality reduction techniques and surrogate modelling) to optimize the computation resources required for these analysis through reducing the dimensionality of the problem.

In the next chapters, dimensionality reduction and surrogate modelling will be combined for a new proposed methodology for the benchmark crash problem.



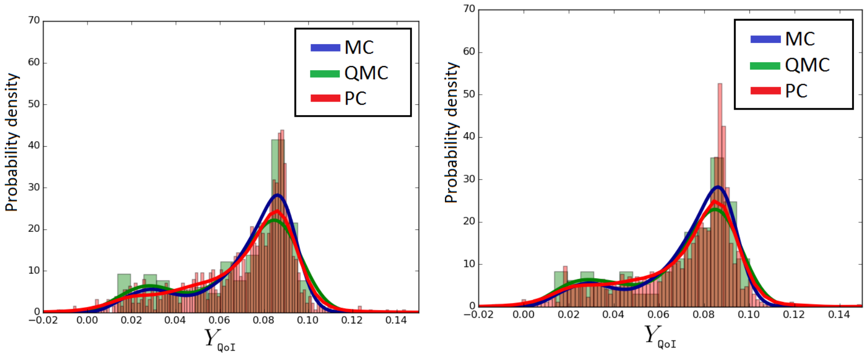
(A) $p = 1$ and $n_f = 4$

(B) $p = 2$ and $n_f = 4$

(C) $p = 3$ and $n_f = 4$

(D) $p = 4$ and $n_f = 4$

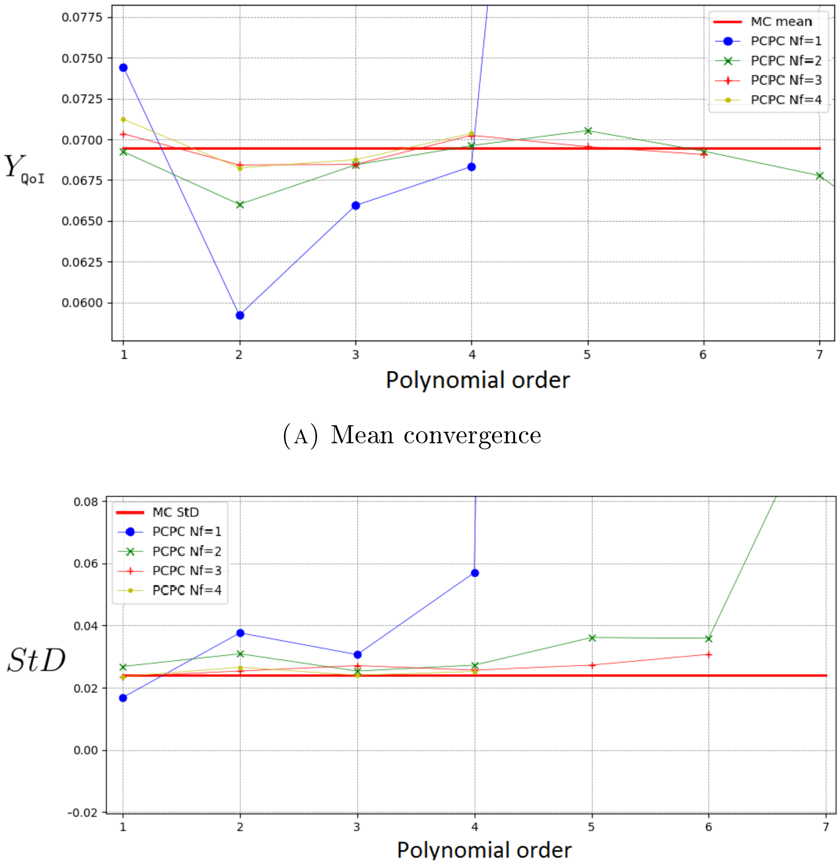
FIGURE 2.22 Probability density function evolution of Point Collocation Polynomial Chaos ($n_f = 4$, Hammersley sampling), QMC (Hammersley sampling) and MC (random sampling). Approaches launched with different polynomial order p and $\mathtt{n_{QMC}}$.



(A) $p = 5$ and $n_f = 2$

(B) $p = 6$ and $n_f = 2$

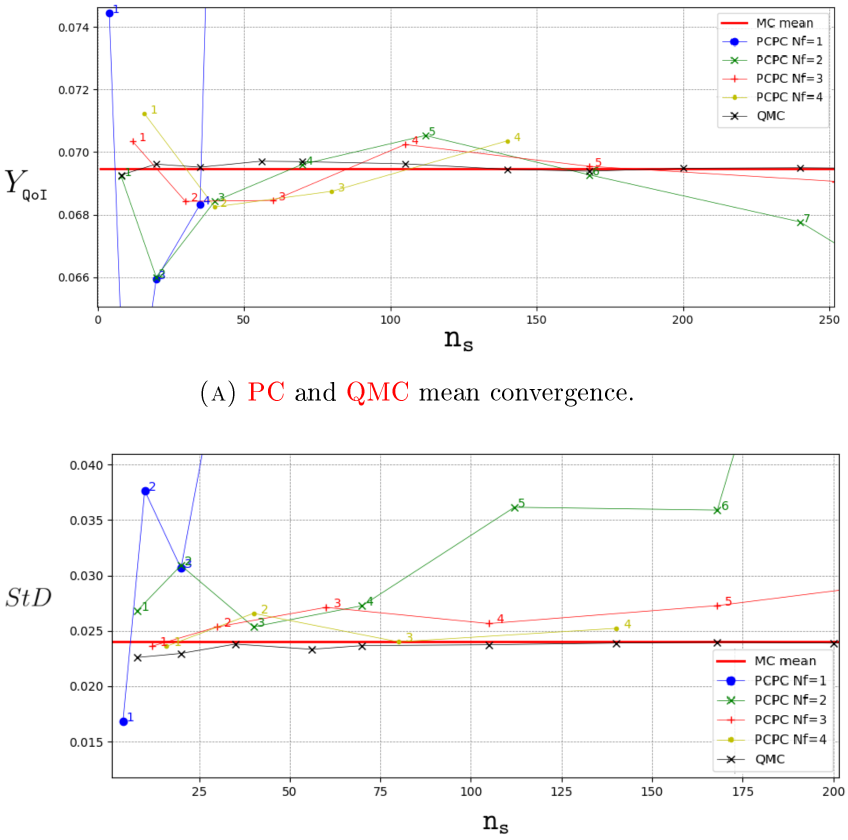
FIGURE 2.23 Probability density function evolution of Point Collocation Polynomial Chaos ($n_f = 2$, Hammersley sampling), QMC (Hammersley sampling) and MC (random sampling). Approaches launched with different polynomial order p and $\mathtt{n_{QMC}}$.



(A) Mean convergence

(B) Standard deviation convergence

FIGURE 2.24 Mean and standard deviation convergence plots with respect to the polynomial order p implemented with Point Collocation PC.



(A) PC and QMC mean convergence.

(B) PC and QMC standard deviation convergence.

FIGURE 2.25 Comparative analysis of the mean and standard deviation with MC, QMC and PC approaches for different configurations of the sampling size $\mathtt{n_s}$, the polynomial order p and the oversampling parameter n_f.

Chapter 3

Nonintrusive uncertainty quantification for nonlinear high dimensional problems

This chapter presents a nonintrusive methodology to deal with uncertainty quantification problems for high dimensional nonlinear outputs. The novelty lies in the combination of dimensionality reduction and surrogate modelling for crashworthiness problems, aiming to detect hidden structure modes and develop statistics of the reduced space. This chapter is structured as follows: section 3.1 presents a brief introduction of the most advanced publications combining dimensionality reduction and surrogate modelling for UQ analysis. The proposed UQ methodology is described in section 3.2. In section 3.3 the main ideas of PCA and kPCA techniques are recalled. Next, in section 3.4 the three different surrogates under consideration are detailed (SRS, OK and PRS). Finally, the different surrogates are readily used to quantify the uncertainty of the output. In section 3.5 it is presented the Monte Carlo sampling for the surrogate model and the comparative criterion. Section 3.6 illustrates the performance of the proposed methodology for the crash problem proposed in Section 2.1. Finally, section 3.7 includes some concluding remarks.

The content of this chapter has been published in *Finite Elements in Analysis and Design* Journal (Rocas et al., 2021).

3.1 Introduction and motivation

In crashworthiness, a single simulation takes CPU hours in a High Performance Computing facility. Thus, the very large number of queries associated with a standard UQ process are practically unaffordable in this context. For a more deeply UQ analysis, the state-of-the-art methods explained in Chapter 2 show weaknesses in terms of computational cost and data analysis to tackle high dimensional outputs. One viable alternative is using a surrogate model (or metamodel) build upon a reduced number of full-order simulations (denoted as *training set*), see Qiu et al., 2018; Wang et al., 2018; Moustapha et al., 2014 for different approaches and comparative analyses. Still, the viability of metamodels is limited by the number of input parameters: a large number of parameters results in a highly multidimensional input space and therefore the engineer is afflicted by the so-called *curse of dimensionality*. If the parametric model is already low-dimensional (3-4 design parameters), the actual threat is not the *curse of dimensionality* but dimensionality reduction is still necessary to computationally afford the simulation process for decision making. This is common in crashworthiness and in the example included here for illustration.

The idea is to determine a low number of relevant parameters (as combinations of the original ones) properly representing all the variability of the dataset. Principal Component Analysis (PCA) is the standard dimensionality reduction technique, to be used if the data structure is such that the low-dimensional subset where the data is contained (also referred as manifold) is linear. Other *manifold learning* techniques identify nonlinear low-dimensional structures. Among them, kernel Principal Component Analysis (kPCA) is considered here because it is one the simplest approaches, see García-González et al., 2020 for a synthetic presentation. The combination of dimensionality reduction with surrogate modeling is a common strategy to carry out UQ in different disciplines and contexts Lataniotis, Marelli, and Sudret, 2018; Li, Wang, and Jia, 2020; Nagel, Rieckermann, and Sudret, 2017.

This chapter analyzes the combination of different alternatives for dimensionality reduction and surrogate models for UQ in crashworthiness simulations with uncertain input parameters. Among the different techniques explores, the novel combination of kPCA and Separated Response Surface (SRS) demonstrates interesting properties. Other strategies are also used to define surrogate model fitting the training set like Polynomial Regression Surface (PRS) and Ordinary Kriging (OK). To do this, we used Monte Carlo sampling (as it is the simplest method for statistical analyses) in two steps of the proposed methodology: 1) to obtain the training set for dimensionality reduction and surrogate model reconstruction, and 2) once the low dimensional surrogate models were properly developed, standard Monte Carlo is performed (practically with no computational cost) to increase the probabilistic resolution in the description of the QoI.

3.2 Dimensionality reduction, surrogate model and UQ

The number of samples $\mathtt{n_s}$ affordable in a real problem is generally not sufficient to produce a proper Monte Carlo assessment of the statistical properties of the output of the system. Chapter 2 demonstrates that the standard Monte Carlo sampling is extremely demanding and, in practice, beyond the possibilities of standard industrial practitioners (Rocas et al., 2020).

As indicated in the previous Chapter 2, the standard Monte Carlo approach consists in generating random samples of the input, running the model and retrieving statistics of the output (or any QoI). This is what corresponds to the upper part (black arrows) in the scheme of Fig.3.1.

However, the part of the standard model (also denoted as full-order, here computed with VPS/Pamcrash) is too computationally expensive to be performed for the number of samples providing statistical relevance. Thus, the alternative is to replace this full-order model by a surrogate, that is a simple functional transformation from $\boldsymbol{h}$ to $\mathbf{x}$. The surrogate is created using a training set consisting in data generated by the full-order model.

An additional difficulty is encountered due to the high-dimension of the outcome of the model, $\mathbf{x}$. It is complicated to create a high-dimensional functional approximation having a target space of d (here 142) dimensions. Thus, previous to undertake the determination of the surrogate, it is convenient to apply some dimensionality reduction technique. In the context of crashworthiness simulation, the data generated by the models are often adopting nonlinear data structures (García-González et al., 2020; Van Der Maaten, Postma, and Herik, 2009). Thus, it is expected to require nonlinear dimensionality reduction techniques as kPCA. For this thesis the QoI is introduced as an indicator for decision making. The QoI summarizes the information contained in $\mathbf{x}$. Quantifying the uncertainty of the QoI is sufficient to take some decisions. Uncertainty Quantification of high-dimensional objects like $\mathbf{x}$ is cumbersome and the outcome is difficult to use as a tool supporting decision making. In that sense, the stochastic assessment focuses in a low-dimensional (even purely scalar) QoI, rather than in a high-dimensional object like $\mathbf{x}$. However, a deeper analysis of the phenomenon requires understanding the underlying mechanisms associated with the overall mechanical response of the system. In that sense, all the information contained in $\mathbf{x}$ is pertinent. The fact that the model order reduction strategy is able to recover back the full-order object in as accurately as possible is therefore extremely advantageous. In this aspect, kPCA behaves much better than PCA in many cases: the simple QoI is fairly approximated by the PCA reduction, but kPCA improves the mapping back to the original variable $\mathbf{x}$.

All these aspects are covered in the methodology described in remainder of the section, having the following steps:

- **Creation of training set.** Generate $\mathtt{n_s}$ realizations of the input parameters $\boldsymbol{h}^i$, for $i = 1, 2, \ldots, \mathtt{n_s}$, and compute the corresponding full-order solutions $\mathbf{x}^i$ (that constitute the training set).

- **Dimensionality reduction.** Analyze the training set and find the principal components allowing to reduce the dimensionality of the family of solutions. In practice, this boils down to apply dimensionality reduction techniques as PCA or kPCA and determine a mapping between the solutions $\mathbf{x} \in \mathbb{R}^d$ and some new variable $\mathbf{z}^\star \in \mathbb{R}^k$ in a much lower-dimensional space ($k \ll d$). The mapping between $\mathbf{x}$ and $\mathbf{z}^\star$ is to be characterized forward and backward. The kPCA backward mapping is found to be more accurate than with PCA. That is, kPCA recovers with much more accuracy a full-order $\mathbf{x}$ associated with a reduced-order coordinate $\mathbf{z}^\star$. Although this advantage is often not perceptible when assessing a low-dimensional (or scalar) QoI, a proper $\mathbf{x}$ recovery is crucial to deepen in the mechanical interpretation of the results. For instance, to identify the mechanisms associated with the different modes of the probability distribution.

- **Surrogate model.** The functional dependence $\mathbf{z}^\star = F(\boldsymbol{h})$ is determined from the data provided by the training set, and the dimensionality reduction.

- **Complete Monte Carlo UQ (using surrogate).** Once the surrogate $F(\cdot)$ is available, for each input value $\boldsymbol{h}$, the corresponding $\mathbf{z}^\star$ is straightforwardly computed as $F(\boldsymbol{h})$. Then the backward mapping produces the corresponding $\mathbf{x}$, and $l^0(\mathbf{x})$ is the associated QoI. The concatenation of the three operations is computationally affordable. Therefore standard Monte Carlo can be performed with a sufficient number of realizations.

The different aspects of the devised methodology are described in detail in the following sections. It is important noting that, among the four conceptual steps mentioned above, the computational cost is concentrated in the creation of the training set. Obtaining this representative collection of solutions requires a computational time in the range of weeks or months, depending on the type of

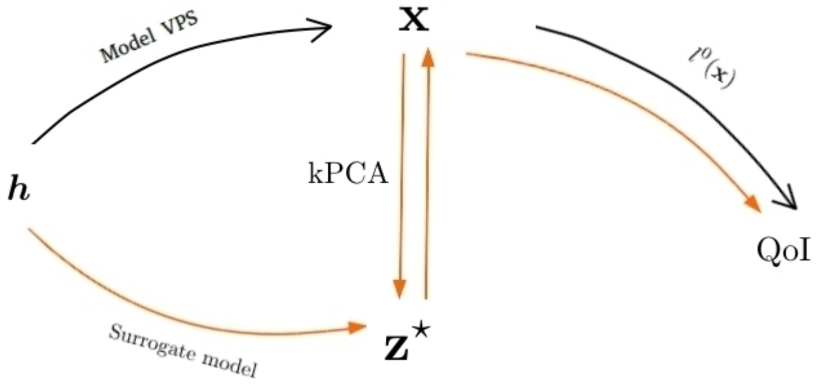


FIGURE 3.1 Schematic illustration of the methodology.

simulation in crashworthiness. The other steps: dimensionality reduction, surrogate modeling and Monte Carlo UQ represent in practice a negligible amount of computational efforts (in the order of seconds).

3.3 Dimensionality reduction

The training set matrix $\mathbf{X} = [\mathbf{x}^1\mathbf{x}^2 \cdots \mathbf{x}^{\mathrm{n_s}}] \in \mathbb{R}^{d\times \mathrm{n_s}}$ is seen as a set of $\mathrm{n_s}$ points in a d-dimensional space. The idea of Dimensionality Reduction (DR) is to find a subspace of lower dimension $k \ll d$ where the set of points is contained.

3.3.1 Principal Component Analysis

The Principal Component Analysis (PCA) strategy consists in diagonalizing the square $d \times d$ matrix $\mathbf{X}^\mathsf{T}\mathbf{X}$ (covariance matrix), that is finding $\mathbf{U} \in \mathbb{R}^{d\times d}$ such that

$$\mathbf{X}^\mathsf{T}\mathbf{X} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\mathsf{T} \tag{3.1}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$. The dimension is reduced from d to k if the last $d-k$ eigenvalues are negligible with respect to the k first. In this case, the new variable selected is

$$\mathbf{z}^\star = \mathbf{U}^{\star\mathsf{T}}\mathbf{x}, \tag{3.2}$$

being $\mathbf{U}^\star \in \mathbb{R}^{d\times k}$ the matrix with the first k columns of $\mathbf{U}$. Eq. (3.2) describes the forward mapping, that is how to map the high-dimensional vector $\mathbf{x}$ into the element $\mathbf{z}^\star$ reduced dimensional space, from dimension d to dimension k.

The backward mapping goes in the opposite direction and reads

$$\mathbf{x} = \mathbf{U}^{\star}\mathbf{z}^{\star}. \tag{3.3}$$

Thus, PCA is a straightforward methodology relying in the fact that the training set is lying in a linear subspace.

3.3.2 Kernel Principal Component Analysis

In many cases, the structure of the low-dimensional manifold where the solution ranges is nonlinear and more sophisticated dimensionality reduction techniques are required. The kernel Principal Component Analysis (kPCA) is an alternative based on the PCA, in fact, it performs PCA in a new feature space where the data is transformed from the original space (Schölkopf, Smola, and Müller, 1998).

The main characteristics of the kPCA are explained in detail in (García-González et al., 2020) and summarized here. It is assumed that some transformation $\mathbf{\Phi}$ from $\mathbb{R}^d$ to a higher-dimensional space is able to *flatten* the training set. That is the transformed training set $\{\mathbf{\Phi}(\mathbf{x}^1), \mathbf{\Phi}(\mathbf{x}^2), \dots, \mathbf{\Phi}(\mathbf{x}^{\mathtt{n_s}})\}$ is such that PCA is able to discover a linear subspace of dimension k. In other words, the transformation $\mathbf{\Phi}$ maps the nonlinear manifold (of dimension k) where the training set ranges into a linear subspace.

The transformation $\mathbf{\Phi}$ that produces this effect is a priori unknown. However, it is worthy trying with some different alternatives and see what is the reduced dimension they propose: the best choice for $\mathbf{\Phi}$ is the one producing the lower value of k. Moreover, in practice $\mathbf{\Phi}$ is indirectly characterized using the *kernel trick*. Thus, instead of describing directly $\mathbf{\Phi}$, an expression for the bivariate form $\kappa(\cdot,\cdot)$ is provided, assuming that the following relation between $\kappa(\cdot,\cdot)$ and $\mathbf{\Phi}(\cdot)$ holds

$$\kappa(\mathbf{x}^i, \mathbf{x}^j) = \mathbf{\Phi}(\mathbf{x}^i)^{\mathsf{T}}\mathbf{\Phi}(\mathbf{x}^j) \tag{3.4}$$

for $i, j = 1, 2, \dots, \mathtt{n_s}$.

A classical choice for the kernel $\kappa(\cdot,\cdot)$ is the so-called Gaussian kernel, defined as

$$\kappa(\mathbf{x}^i, \mathbf{x}^j) = \exp(-\beta \left\|\mathbf{x}^i - \mathbf{x}^j\right\|^2) \tag{3.5}$$

where β is a parameter that, in the applications of this research, is taken equal to 0.1.

Having the kernel at hand, one may compute a matrix equivalent to $\mathbf{X}\mathbf{X}^{\mathsf{T}}$ for the samples transformed by $\mathbf{\Phi}$. This matrix is denoted by $\mathbf{G} \in \mathbb{R}^{\mathtt{n_s}\times\mathtt{n_s}}$ and has generic coefficient

$$[\mathbf{G}]_{ij} = \kappa(\mathbf{x}^i, \mathbf{x}^j) \tag{3.6}$$

It is worth noting that the eigenvalues of $\mathbf{X}\mathbf{X}^\mathsf{T}$ are the same of those of $\mathbf{X}^\mathsf{T}\mathbf{X}$, which are the ones extracted in (3.1). Actually, $\mathbf{G}$ is also readily diagonalized and the following factorization is obtained

$$\mathbf{G} = \mathbf{V}\tilde{\mathbf{\Lambda}}\mathbf{V}^\mathsf{T} \tag{3.7}$$

where $\tilde{\mathbf{\Lambda}}$ contains the same non-zero eigenvalues that would be obtained from diagonalizing the corresponding covariance matrix, which is not available.

Thus, the eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{\mathtt{n_s}}$ are computed, and the reduced dimension k is selected such that the last $\mathtt{n_s} - k$ eigenvalues are negligible with respect to the k first.

Once k is obtained, the original variable $\mathbf{x} \in \mathbb{R}^d$ is mapped into a variable in the reduced space, $\mathbf{z}^\star \in \mathbb{R}^k$ using the following expression

$$\mathbf{z}^\star = {\mathbf{V}^\star}^\mathsf{T}\mathbf{g}(\mathbf{x}), \tag{3.8}$$

where $\mathbf{V}^\star \in \mathbb{R}^{\mathtt{n_s} \times k}$ is the matrix with the first k columns of $\mathbf{V}$ and $\mathbf{g}(\mathbf{x})\mathbb{R}^{\mathtt{n_s}}$ is a vector with generic component

$$[\mathbf{g}(\mathbf{x})]_i = \kappa(\mathbf{x}^i, \mathbf{x}) \tag{3.9}$$

for $i = 1, \ldots, \mathtt{n_s}$.

As described in detail in (García-González et al., 2020), if the samples transformed by $\mathbf{\Phi}$ are not centred, some corrections have to be done and both matrix $\mathbf{G}$ and vector $\mathbf{g}$ have to be modified accordingly. These corrections are straightforward and are omitted here for the sake of a simpler presentation.

Equations (3.8) and (3.9) characterize the forward kPCA mapping, from $\mathbf{x}$ to $\mathbf{z}^\star$. The backward mapping for kPCA is not as simple as for the PCA version described in equation (3.3). A point $\mathbf{z}^\star$ in the reduced space is mapped back to a point $\mathbf{x}$ which is recovered as a weighted average of the points of the training set, namely

$$\mathbf{x} = \sum_{i=1}^{\mathtt{n_s}} w_i(\mathbf{z}^\star)\mathbf{x}^i \quad \text{, with weights such that } \sum_{i=1}^{\mathtt{n_s}} w_i(\mathbf{z}^\star) = 1 \tag{3.10}$$

The weights $w_i(\mathbf{z}^\star)$ are computed such that the forward mapping of $\mathbf{x}$ is as close as possible to $\mathbf{z}^\star$. A popular strategy to compute these weights with a simple approach is to use a radial basis interpolation concept based on the distances of $\mathbf{z}^\star$ to the images of the sample points, $\mathbf{z}^{\star i}$ for $i = 1, \ldots, \mathtt{n_s}$. That is computing $d_i = \|\mathbf{z}^\star - \mathbf{z}^{\star i}\|$ and taking any value of $w_i(\mathbf{z}^\star)$ decreasing with d_i, for example $w_i(\mathbf{z}^\star) \propto \frac{1}{d_i^2}$.

3.4 Surrogate Modeling

As already announced, the dimensionality reduction techniques presented above are a previous step to build a Surrogate Model (SM). The training set is now used to approximate the functional dependence associated with the full-order model. The final goal is to compute $\mathbf{x}$ as an easy-to-evaluate function of $\boldsymbol{h}$, that is the surrogate. With the dimensionality reduction, this is split in two steps: a surrogate from $\boldsymbol{h}$ to $\mathbf{z}^\star$ plus the backward mapping from $\mathbf{z}^\star$ to $\mathbf{x}$, see Fig. 3.1.

Here, the surrogates are presented as generic methodologies to establish a functional dependency among some input $\boldsymbol{h}$ and some output function $\mathbf{y}(\boldsymbol{h})$ (we use $\mathbf{y}$ to account for any output, that could be either $\mathbf{x}$ or $\mathbf{z}^\star$). Obviously, doing the surrogate with $\mathbf{z}^\star$ has the advantage of dealing with a much lower dimension (number of components) of the model output. In practice, for the sake of a simpler presentation, a scalar output $Y(\boldsymbol{h})$ is considered in the following, that stand, for example, for any of the components of $\mathbf{y}(\boldsymbol{h})$. In the examples, Y coincides with the first component of the reduced space using kPCA, that is $Y = [\mathbf{z}^\star]_1$.

In the following, the parameters describing the stochastic input space where the function takes values are collected in the vector $\boldsymbol{h} = [h_1 \ldots h_{\mathtt{n_d}}]^\mathsf{T} \in \mathbb{R}^{\mathtt{n_d}}$. Where $\mathtt{n_d}$ is the number of stochastic dimensions of the problem ($\mathtt{n_d} = 3$ in the benchmark under consideration).

Thus, the goal is to approximate the functional dependence $Y(\boldsymbol{h})$ using the images of the points of the training set $y^k = Y(\boldsymbol{h}^k)$, $k = 1, \ldots, \mathtt{n_s}$, where, all the sample points are collected in the vector $\mathbf{y} = [y^1 y^2 ... y^{\mathtt{n_s}}]^T$.

3.4.1 Separated Response Surface

The idea of Separated Response Surface (SRS) is to find a separated approximation $F(\boldsymbol{h})$ to $Y(\boldsymbol{h})$. The separated character of $F(\boldsymbol{h})$ means that it is a sum of rank-one terms, being each rank-one term the product of sectional modes (the adjective sectional is used to indicate that the mode depends only on one of the parameters). The algorithm employed to compute the SRS is based on the ideas of the least-squares Proper Generalized Decomposition (PGD) approximations described in detail in (Díez et al., 2018; Díez et al., 2019; Lu, Blal, and Gravouil, 2018b; Lu, Blal, and Gravouil, 2018a).

Thus, $F(\boldsymbol{h})$ reads

$$F(\boldsymbol{h}) = \sum_{j=1}^{\mathtt{n_f}} \sigma_j \prod_{i=1}^{\mathtt{n_d}} f_i^j(h_i) \tag{3.11}$$

where each sectional mode $f_i^j(h_i)$ is represented in some discrete sectional space. The discrete sectional space is generated by a family of functions $\left\{\Psi_1^i(h_i)\,\Psi_2^i(h_i)\ldots\Psi_{\mathtt{n}_i}^i(h_i)\right\}$ being $\mathtt{n}_i$ the dimension of the sectional function space. Accordingly, sectional modes have the following expression

$$f_i^j(h_i) = \sum_{m=1}^{\mathtt{n}_i} a_i^m \Psi_m^i(h_i) \tag{3.12}$$

where the unknown coefficients a_i^m, for $i = 1, \ldots, \mathtt{n_d}$ and $m = 1, \ldots, \mathtt{n}_i$, have to be computed to determine the sectional mode $f_i^j(h_i)$, for $j = 1, 2, \ldots$

Different alternatives are available as the approximation space defined by $\Psi_m^i(h_i)$. Here we have considered a finite element discretization with $\mathtt{n}_i$ nodes (in 1D domains, $\mathtt{n}_i - 1$ elements).

A least-squares criterion based on a discrete Euclidean product is chosen to select $F(\boldsymbol{h})$. Thus, $F(\boldsymbol{h}) \approx Y(\boldsymbol{h})$ is taken such that it minimizes

$$\|F(\boldsymbol{h}) - Y(\boldsymbol{h})\|^2 = \langle F - Y, F - Y\rangle = \sum_{k=1}^{\mathtt{n_s}} w^k (F(\boldsymbol{h}^k) - y^k)^2 \tag{3.13}$$

where the weights w^k are introduced to assimilate the sum into an integral, that is, to assume that

$$\int_{\Omega_h} F(\boldsymbol{h})\, d\boldsymbol{h} \approx \sum_{k=1}^{\mathtt{n_s}} w^k F(\boldsymbol{h}^k)$$

note that the associated scalar product $\langle\cdot,\cdot\rangle$ of two arbitrary functions F and G reads

$$\langle F, G\rangle = \sum_{k=1}^{\mathtt{n_s}} w^k F(\boldsymbol{h}^k) G(\boldsymbol{h}^k). \tag{3.14}$$

Note that weights w^k, $k = 1, 2, \ldots, \mathtt{n_s}$ must be selected corresponding to a quadrature having as integration points $\boldsymbol{h}^k$, where Y is known. Typically, the distribution of points $\boldsymbol{h}^k$ is provided by a stochastic sampling and cannot be enforced a priori by the user to obtain his/her preferred quadrature (e.g. a Gauss-Legendre quadrature or a composite Simpson's rule). Thus, the weights of the quadrature are adapted to optimize the integration order in a (multidimensional) Newton-Cotes fashion.

Thus, least-squares solution in a linear functional space $\boldsymbol{V}$ is readily computed as a projection, that is finding $F \in \boldsymbol{V}$ such that

$$\langle F, F^{\star} \rangle = \langle Y, F^{\star} \rangle \text{ for all } F^{\star} \in \boldsymbol{V} \tag{3.15}$$

Note that *integral* equation (3.15) has to be fulfilled for any weighting function (or test function) $F^{\star}$ in $\boldsymbol{V}$, as in the standard weak form of a boundary value problem.

The key aspect of any PGD algorithm is how to solve the rank-one approximation. That, is how to find an approximation to $Y(\boldsymbol{h})$ with a function of the form

$$F(\boldsymbol{h}) = \sigma \prod_{i=1}^{\mathtt{n_d}} f_i(h_i) \tag{3.16}$$

which is a particular case of (3.11) with just one term.

The standard PGD strategy consists in an alternate direction approach, that is to compute the sectional mode f_γ, the rest of the sectional modes f_i for $i \neq \gamma$ are assumed to be known. Thus, in practice, $F(\boldsymbol{h})$ and $\delta F(\boldsymbol{h})$ are taken as

$$F(\boldsymbol{h}) = f_\gamma(x_\gamma) \left[\prod_{i \neq \gamma} f_i(h_i) \right] \tag{3.17}$$

and

$$\delta F(\boldsymbol{h}) = \delta f_\gamma(h_\gamma) \left[\prod_{i \neq \gamma} f_i(h_i) \right] . \tag{3.18}$$

This alternate directions strategy leads to a sectional problem, reduced to the γ coordinate. The family of sectional problem is to be solved sequentially for $\gamma = 1, 2, \dots, \mathtt{n_s}$, and then iterated until convergence is reached.

For the sake of simplifying the writing, the computable term depending on all the sectional modes but γ is denoted as T_γ and T_γ^k when evaluated in $\boldsymbol{h}^k$, namely

$$T_\gamma := \prod_{i \neq \gamma} f_i(h_i) \text{ and } T_\gamma^k := \prod_{i \neq \gamma} f_i(h_i^k). \tag{3.19}$$

Thus, the sectional counterpart of (3.15) reads

$$\langle f_\gamma T_\gamma, \delta f_\gamma T_\gamma \rangle = \langle Y, \delta f_\gamma T_\gamma \rangle \text{ for all } \delta f_\gamma \tag{3.20}$$

that is

$$\sum_{k=1}^{\mathtt{n}_\mathtt{s}} w^k (T_\gamma^k)^2 f_\gamma(x_\gamma^k)\delta f_\gamma(h_\gamma^k) = \sum_{k=1}^{\mathtt{n}_\mathtt{s}} w^k T_\gamma^k y^k \delta f_\gamma(x_\gamma^k). \tag{3.21}$$

Using a particular case of (3.12), that is

$$f_\gamma(h_\gamma) = \sum_{m=1}^{\mathtt{n}_\gamma} a_\gamma^m \varPsi_m^\gamma(h_i) \tag{3.22}$$

in (3.21) and taking $\delta f_\gamma(h_\gamma^k) = \varPsi_\ell^\gamma(h_\gamma^k)$ for all $\ell = 1, \ldots, \mathtt{n}_\gamma$ yields

$$\sum_{k=1}^{\mathtt{n}_\mathtt{s}} w^k (T_\gamma^k)^2 \sum_{m=1}^{\mathtt{n}_\gamma} a_\gamma^m \varPsi_m^\gamma(h_\gamma^k) \varPsi_\ell^\gamma(h_\gamma^k) = \sum_{k=1}^{\mathtt{n}_\mathtt{s}} w^k T_\gamma^k y^k \varPsi_\ell^\gamma(h_\gamma^k) \tag{3.23}$$

or

$$\sum_{m=1}^{\mathtt{n}_\gamma} \underbrace{\left[\sum_{k=1}^{\mathtt{n}_\mathtt{s}} w^k (T_\gamma^k)^2 \varPsi_m^\gamma(h_\gamma^k) \varPsi_\ell^\gamma(h_\gamma^k)\right]}_{M_{\ell m}} a_\gamma^m = \underbrace{\sum_{k=1}^{\mathtt{n}_\mathtt{s}} w^k T_\gamma^k y^k \varPsi_\ell^\gamma(h_\gamma^k)}_{f_\ell} \tag{3.24}$$

for all $\ell = 1, \ldots, \mathtt{n}_\gamma$. That is, a linear system of $\mathtt{n}_\gamma$ equation with $\mathtt{n}_\gamma$ unknowns

$$\boldsymbol{M}\boldsymbol{a}_\gamma = \boldsymbol{f}. \tag{3.25}$$

Once the sectional approximation is obtained solving (3.25), the loop in alternate directions iterations is continued until convergence and completion of the rank-one computation. As usual in PGD (Garikapati et al., 2020), once the rank-one solution is obtained, the greedy approach aims at computing the next term (next j in (3.11)).

As it is standard in this type of strategies, in order to compute an approximation having the separated form given in (3.11), there are three nested loops. First, the greedy approach (loop in j) aims at computing rank-one terms having the form given in (3.16). Then an alternated direction iterative scheme is applied consisting in two nested loops: the iterative loop to reach convergence (not described explicitly in this text with an iteration index) and an inner loop for $\gamma = 1, 2, \ldots \mathtt{n}_\mathtt{s}$, ranging all sectional dimensions. This is standard in the references describing any PGD scheme, see Díez et al., 2019 for an algorithmic description.

As mentioned above, functions $\varPsi_m^i$ in (3.12) are chosen as classical $\mathcal{C}^0$ finite elements shape functions. Contrary to other choices (e.g. high-order polynomials) these type of functions are more stable due to their local support but introduce a lack of smoothness (jumps in the first derivatives, singularities in the second derivatives). Consequently, when using a finite element approximation for the sectional modes, it is important having the possibility of enforcing the smoothness of the solution. This is equivalent to penalize in system (3.25), the non-smoothness of the sectional function described in (3.22). This requires, for instance, penalizing some postprocessed quantity of the sectional mode $f_\gamma(h_\gamma)$, represented by the vector of nodal values, $\boldsymbol{a}_\gamma$. The quantity to be penalized, the lack of smoothness, is represented by a matrix $\boldsymbol{G}$ mapping the nodal values of $f_\gamma(h_\gamma)$ into the postprocessed quantity in some representative points. In the following, $\boldsymbol{G}$ is taken as the standard *gradient* operator, computing the derivatives of $f_\gamma(h_\gamma)$ in the integration points of the elements of the mesh. Thus, $\boldsymbol{G}$ is a $\mathtt{n_G} \times \mathtt{n}_\gamma$ matrix, being $\mathtt{n_G}$ the number of integration points in the mesh (assuming that the dimension of the sectional space is 1). Thus, the measure of the lack of smoothness that has to be reduced is given by $\boldsymbol{a}_\gamma^\intercal \boldsymbol{G}^\intercal \boldsymbol{G} \boldsymbol{a}_\gamma$. Provided that system (3.25) is equivalent to minimize the following functional

$$\frac{1}{2}\boldsymbol{a}_\gamma^\intercal \boldsymbol{M} \boldsymbol{a}_\gamma - \boldsymbol{f}^\intercal \boldsymbol{a}_\gamma$$

Enforcing the smoothness requires minimizing the perturbed functional

$$\frac{1}{2}\boldsymbol{a}_\gamma^\intercal \boldsymbol{M} \boldsymbol{a}_\gamma - \boldsymbol{f}^\intercal \boldsymbol{a}_\gamma + \lambda \frac{1}{2}\boldsymbol{a}_\gamma^\intercal \boldsymbol{G}^\intercal \boldsymbol{G} \boldsymbol{a}_\gamma$$

for some value of the factor λ that states the importance of the smoothing. The larger is λ, the smoother is the recovered solution. This results in the following linear system, which is a modification of (3.25) accounting for the smoothing

$$\left[\boldsymbol{M} + \lambda \boldsymbol{G}^\intercal \boldsymbol{G}\right] \boldsymbol{a}_\gamma = \boldsymbol{f} \tag{3.26}$$

3.4.2 Ordinary Kriging

Ordinary Kriging (OK) is an interpolation technique commonly used in engineering and originated for geostatistical problems (Oliver and Webster, 2014). The OK method determines weights for a set of simulation points to calculate a prediction of a new sample. The weights are calculated with a variogram model that has the main feature to estimate variances for any distance. The kriging metamodel $F(\boldsymbol{h})$ of any point $\boldsymbol{h}$ is defined by:

$$F(\boldsymbol{h}) = \sum_{i=1}^{\mathtt{n_s}} w_i y^i, \tag{3.27}$$

where the unknowns w_i are the weights and y^i are the scalar values of the function to be interpolated. To determine the optimal values for the Kriging weights, the variogram function plays an important role. There exist different variograms: Gaussian, exponential, linear among others (Oliver and Webster, 2014). The OK matrix system to obtain the weights reads,

$$\begin{pmatrix} \gamma_{11} & \gamma_{12} & \dots & \gamma_{1\mathtt{n_s}} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \gamma_{\mathtt{n_s}1} & \gamma_{\mathtt{n_s}2} & \ddots & \gamma_{\mathtt{n_s}\mathtt{n_s}} & 1 \\ 1 & 1 & \dots & 1 & 0 \end{pmatrix} \begin{pmatrix} w_1 \\ \vdots \\ w_{\mathtt{n_s}} \\ \mu \end{pmatrix} = \begin{pmatrix} \gamma_{10} \\ \vdots \\ \gamma_{\mathtt{n_s}0} \\ 1 \end{pmatrix}.$$

A specific condition for OK with respect to other Kriging methods is enforcing the sum of weights equal to 1, $\sum_{i=1}^{\mathtt{n_s}} w_i = 1$. This condition is achieved by introducing the new unknown μ as Lagrange multiplier (Malvić and Balić, 2009). The entries of the matrix in the equation above depend on the variogram function γ evaluated for each distance δ between a pair of samples, that is $\gamma_{ij} = \gamma(\delta)$, being $\delta = \|\boldsymbol{h}^i - \boldsymbol{h}^j\|$. The entries $\gamma_{10} \dots \gamma_{\mathtt{n_s}0}$ are evaluations of the variogram γ between all the sample points with respect to the new (current) point. Here, we used the spherical variogram defined as:

$$\gamma(\delta) = \begin{cases} C_0 + C_1 \left[\frac{3}{2}\left(\frac{\delta}{a}\right) - \frac{1}{2}\left(\frac{\delta}{a}^3\right)\right], & 0 < \delta \leq a \\ C_0 + C_1, & \delta > a \end{cases} \tag{3.28}$$

C_0 is the nugget constant representing the noise of the data, a is the range of the transition zone where the variogram levels off and the sill $(C_0 + C_1)$ is defined as the total variance of the model. For the benchmark problem $C_0 = 0$, in consequence C_1 is the total variance of the model. In Fig.3.2 it is illustrated a spherical variogram function. A specific condition for OK with respect others Kriging methods are the sum of weights equal to 1, $\sum_{i=1}^{\mathtt{n_s}} w_i = 1$. This condition is achieved by using Lagrange multipliers (Malvić and Balić, 2009).

3.4.3 Polynomial Response Surface

Polynomial Response Surface (PRS) has been applied in numerous studies to build metamodel for different engineering problems (Gano, Kim, and Brown, 2006; Fang et al., 2005; Giunta and Watson, 1998). It consists in a simple multidimensional polynomial fitting. A second order polynomial model $F(\boldsymbol{h})$

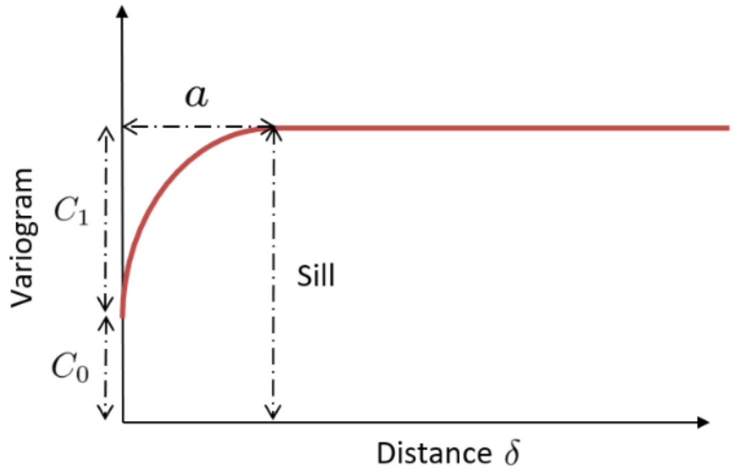


FIGURE 3.2 Variogram with the three main parameters. The nugget C_0, the range a and the sill $C_0 + C_1$.

takes the form,

$$F(\boldsymbol{h}) = c_0 + \sum_{i=1}^{\mathtt{n_d}} c_i h_i + \sum_{i=1}^{\mathtt{n_d}} c_{ii} h_i^2 + \sum_{i=1}^{\mathtt{n_d}-1} \sum_{j=i+1}^{\mathtt{n_d}} c_{ij} h_i h_j, \tag{3.29}$$

where h_i is the i-th stochastic input, the different coefficients c are the unknowns to be computed, collected in a vector $\mathbf{c}$. If the approximation was able to interpolate the data, the following linear system should be solved:

$$\mathbf{Ac} = \mathbf{y}, \tag{3.30}$$

where $\mathbf{A}$ is the matrix containing the values of the different interpolation functions in (3.29), that is, for $\mathtt{n_d} = 3$,

$$\{1, h_1, h_2, h_3, (h_1)^2, (h_2)^2, (h_3)^2, h_1h_2, h_1h_3, h_2h_3\}$$

in the sample points $\boldsymbol{h}^k$, for $k = 1, \ldots, \mathtt{n_s}$. This results in

$$\mathbf{A} = \begin{bmatrix} 1 & h_1^1 & h_2^1 & h_3^1 & (h_1^1)^2 & (h_2^1)^2 & (h_3^1)^2 & h_1^1h_2^1 & h_1^1h_3^1 & h_2^1h_3^1 \\ 1 & h_1^2 & h_2^2 & h_3^2 & (h_1^2)^2 & (h_2^2)^2 & (h_3^2)^2 & h_1^2h_2^2 & h_1^2h_3^2 & h_2^2h_3^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & h_1^{\mathtt{n_s}} & h_2^{\mathtt{n_s}} & h_3^{\mathtt{n_s}} & (h_1^{\mathtt{n_s}})^2 & (h_2^{\mathtt{n_s}})^2 & (h_3^{\mathtt{n_s}})^2 & h_1^{\mathtt{n_s}}h_2^{\mathtt{n_s}} & h_1^{\mathtt{n_s}}h_3^{\mathtt{n_s}} & h_2^{\mathtt{n_s}}h_3^{\mathtt{n_s}} \end{bmatrix}, \tag{3.31}$$

The model is often non-interpolative, with more equations (points in the sample) than unknowns (number of coefficients in $\mathbf{c}$). Therefore system (3.30) cannot be solved exactly but using a least squares minimization criterion, that is minimizing the Euclidean norm of the residual, namely $\|\mathbf{y} - A\mathbf{c}\|$. This results in taking the vector of unknown coefficients solution of

$$(\mathbf{A}^T\mathbf{A})\mathbf{c} = \mathbf{A}^T\mathbf{y}. \tag{3.32}$$

This method presents drawbacks for high-dimensional data and data with oscillations. Increasing the order of the Polynomials may improve accuracy. However, for high-order approximations Runge's phenomenon creates instabilities and wrong predictions (Boyd and Xu, 2009).

3.5 Uncertainty Quantification

3.5.1 Monte Carlo sampling with surrogate modeling

Once the surrogate model is available, the Monte Carlo UQ assessment with a large number of samples $\mathtt{n}_{\mathtt{MC}}$ is produced at an affordable computational cost.

Thus, for each of the three metamodels introduced above (SRS, OK and PRS) $\mathtt{n}_{\mathtt{MC}}$ realizations $\boldsymbol{h}^1, \boldsymbol{h}^2, \ldots, \boldsymbol{h}^{\mathtt{n}_{\mathtt{MC}}}$ are produced and the corresponding value of the mean, variance (and standard deviation) and probability density function (pdf, to be approximated as a histogram) is readily estimated:

$$\text{Mean} = \mathbb{E}[F(\boldsymbol{h})] = \frac{1}{\mathtt{n}_{\mathtt{MC}}} \sum_{k=1}^{\mathtt{n}_{\mathtt{MC}}} F(\boldsymbol{h}^k) \tag{3.33}$$

$$\text{Variance} = \sigma^2 = \frac{1}{\mathtt{n}_{\mathtt{MC}} - 1} \sum_{k=1}^{\mathtt{n}_{\mathtt{MC}}} \left(F(\boldsymbol{h}^k) - \mathbb{E}[F(\boldsymbol{h})]\right)^2, \tag{3.34}$$

The PDF corresponding to $Y = F(\boldsymbol{h})$ is denoted by $f_Y(y)$ and it is approximated by histogram $p_Y(y)$ computed on the basis of the $\mathtt{n}_{\mathtt{MC}}$ Monte Carlo samples $y^k = F(\boldsymbol{h}^k)$, for $k = 1, 2, \ldots, \mathtt{n}_{\mathtt{MC}}$. Note that histogram $p_Y(y)$ is a piecewise constant function defined over a partition in uniform intervals of the Y domain, $\Omega_Y = \bigcup_{\ell=1}^{n_Y} I_\ell$. Piecewise constant function p_Y is such that for $y \in I_\ell$, $p_Y(y)$ is equal to the number of samples y^k lying in I_ℓ divided by $\mathtt{n}_{\mathtt{s}}$.

Each response surface is used to generate the images of the $\mathtt{n}_{\mathtt{MC}} = 50000$ samples of the input space $\boldsymbol{h} \in \mathbb{R}^{\mathtt{n}_{\mathtt{d}}}$, that is $\boldsymbol{h} \to Y$). The backward mapping technique ($Y \to \mathbf{X} \to QoI$) described in Section 3.3 is used to obtain the statistics of the QoI.

Comparing the obtained values of mean and variance is straightforward because they are scalar values. However, comparing PDFs is not as trivial. Here, the Kullback Leibler divergence technique is proposed as a criterion to compare the PDF functions.

3.5.2 Comparative criterion for PDFs

Kullback-Leibler (KL) divergence is used as a comparative criterion for the different resulting Monte Carlo PDFs of each metamodel. This quantity measures differences between two PDF functions Galas et al., 2017. Two random variables F and G have PDFs f and g. The KL divergence is introduced as a distance that quantifies if the two random variables are similar enough. The random variable F and its PDF f are taken as the reference and g is considered to be an approximation to f. Thus, KL divergence between the two continuous PDFs f and g reads:

$$D_{KL}(F \| G) = \int_{-\infty}^{\infty} f(y) \log \left(\frac{f(y)}{g(y)} \right) dy. \tag{3.35}$$

Note that equation (3.35) is associated with the notion of *entropy* and it is interpreted as the relative entropy or the *information gain* from G to F.

In the case the PDFs are replaced by their discrete counterparts, that is histograms, instead of f and g, one has histograms p_Y and q_Y with the format described in the previous section. Thus, p_Y and q_Y are expressed as the values of the probability of being in each of the n_Y bins, that is $p_Y(y^\ell)$ and $q_Y(y^\ell)$, for $\ell = 1, 2, \ldots, n_Y$ and $y^\ell \in I_\ell$. The discrete counterpart of equation (3.35) reads

$$D_{KL}(p_Y \| q_Y) = \sum_{\ell=1}^{n_Y} p_Y(y^\ell) \log \left(\frac{p_Y(y^\ell)}{q_Y(y^\ell)} \right). \tag{3.36}$$

The values obtained using the discrete KL divergence introduced in equation (3.36) depend on the number of bins, n_Y. In order to normalize these values, a normalizing constant is introduced providing a reference to understand whether the resulting discrete KL divergence is actually small enough. Note that the KL divergence is seen as a distance but it is not conceived as the norm of a difference. Thus, it is not possible to normalize dividing directly by the norm of p_Y (or f in (3.35)). In order to obtain a reference value, we propose taking the *distance* of p_Y to the less informative distribution, that is the uniform histogram q_U such that $q_U(y^\ell) = \frac{1}{n_Y}$, for $\ell = 1, 2, \ldots, n_Y$. The rationale behind this choice is taking q_U as the *zero* or absolute reference distribution. This value is denoted as D^0_{KL} and reads

$$D^0_{KL} = D_{KL}(p_Y \| q_U) = \sum_{\ell=1}^{n_Y} p_Y(y^\ell) \log\left(n_Y\, p_Y(y^\ell)\right). \tag{3.37}$$

Note that the quantity D^0_{KL} defined in equation (3.37) is actually the entropy of p_Y. Dividing the figures obtained with the KL divergence of (3.36) by D^0_{KL} provides a relative value that allows elaborating a more informed criterion to decide if p_Y and q_Y are sufficiently close to each other.

3.6 Benchmark B-Pillar results with DR and SM

In this section, the combination methodology of kPCA + SRS explained in detail in the previous is applied for the B-Pillar crash problem to show the performance of the methodology. Also, OK and PRS are applied as a classic surrogate techniques to compare the SRS results.

3.6.1 DR with kPCA

Initially, a certain amount of VPS/Pamcrash simulations is required to reconstruct the input matrix $\mathbf{X}$ for kPCA manifold analysis and dimensionality reduction (the training set computed in an off-line phase). These simulations are the initial samples for the data analysis. A key issue is quantifying the number of samples required to obtain enough and credible information to describe the low-dimensional manifold containing the solution. As described above, we devise the combined use of kPCA to reduce the dimensionality, different techniques to build a response surface, and the KL divergence as a measure to compare the different probability distributions and stop enriching the sampling. This process is illustrated in the flowchart scheme shown in Fig.3.3, and it is detailed next:

The eigenvalues of the kernel matrix $\mathbf{G} = \mathbf{V}\tilde{\mathbf{\Lambda}}\mathbf{V}^\intercal$ in kPCA (equation 3.7), show the quantity of information collected by each associated eigenvector, as explained in section 3.3 and more detailed in García-González et al., 2020. Summarizing, the largest eigenvalue measures the largest amount of information collected by the corresponding eigenvector. For instance, for the first eigenvalue of matrix $\tilde{\mathbf{\Lambda}}$, the associated eigenvector is the first column of matrix $\mathbf{V}$. Adopting the first component as reduced model (keeping only one principal component, the first one), the $d = 142$ dimensions of the training set samples (each column

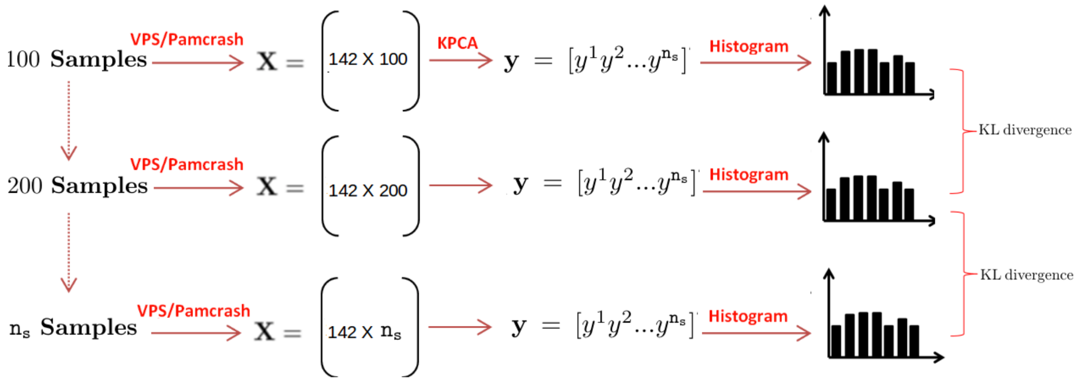


FIGURE 3.3 Flowchart scheme to select the number of simulation $\mathtt{n_s}$ for the kPCA input matrix $\mathbf{X}$.

of matrix $\mathbf{X}$) are reduced to one scalar number and the $\mathtt{n_s}$ samples are stored in the vector $\mathbf{y} = [y^1 y^2 ... y^{\mathtt{n_s}}]^T$, see sections 3.3 and 3.4 for more details.

In the case of our benchmark crash problem, reducing to one dimension collects more than 80% of information of the manifold where data belong. This 80% figure allows considering as admissible in this context, and in agreement with the resulting approximations, the very advantageous reduction to a single dimension. This figure is calculated along the sampling refinement process (for different values of $\mathtt{n_s}$) to check the behaviour of the quantity of information retained in the one-dimensional reduction. This is shown in Fig.3.4, where it can be noticed that even using only 100 samples, the first eigenvalue collects already 80% of information.

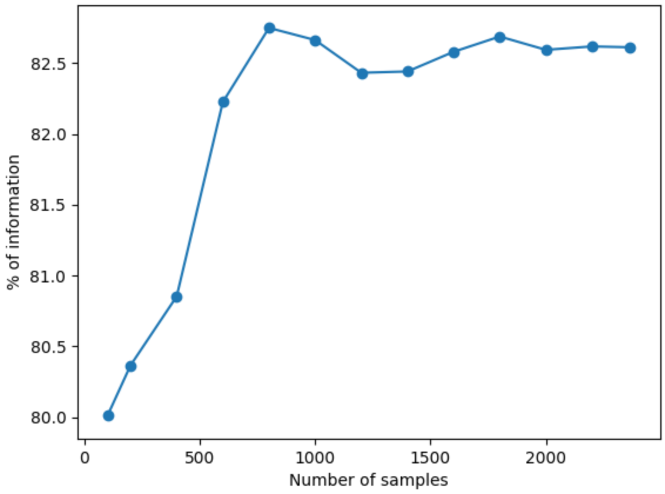


FIGURE 3.4 Quantity of information [%] stored by the first eigenvector by increasing the number of samples $\mathtt{n_s}$.

At this point, the number of samples $\mathtt{n_s}$ is required to guarantee some statistical accuracy. The KL divergence is used to compare the subsequent distributions of probability of the QoI, obtained with the training sets corresponding to the different values of $\mathtt{n_s}$, see Fig. 3.3. That means, to validate the final value of $\mathtt{n_s} = 2366$, to be used in the input matrix $\mathbf{X}$ for analysis (instead of the starting value of $\mathtt{n_s} = 100$). That is, the histogram obtained by the values of $\mathbf{y}$ for a low number of samples $\mathtt{n_s}$ is compared by the KL divergence criterion with the histogram for a higher number of samples. This process is repeated by increasing the number of samples until the value of the KL divergence becomes smaller than 10^{-2} (considered low enough for our required accuracy). The results obtained in this process are detailed in Fig.3.5, where it is clear how the histograms become more stable increasing the number of samples, and consequently the KL-Div value decreases. For the final number of samples $\mathtt{n_s} = 2366$, the KL-Div value is below the prescribed tolerance. Recalling expression (3.37), the calculated value of D^0_{KL} for the final histogram in Fig.3.5 is $D^0_{KL} = 0.2361$. Thus, the relative value of the difference of the last two distributions is of 2.9%.

Additionally, Fig.3.6 shows how the KL-Div value becomes stable when the number of samples is rich enough. At this point, increasing the number of samples does not add extra information to the model.

Fig.3.7a shows the final histogram obtained for the final sampling, $\mathtt{n_s} = 2366$. Additionally, the low dispersion of the results by using one reduced dimension is confirmed (the first eigenvalue contains 82.61% of information, stored in $\mathbf{y} = [y^1 y^2 ... y^{\mathtt{n_s}}]^T$). Moreover, the consistency of the *backward mapping* from vector $\mathbf{y}$ to $\mathbf{x}$ and then calculating the corresponding QoI is also confirmed by the results in Fig.3.7b.

3.6.2 Link between input space and reduced space

The first principal component $\mathbf{y}$ is linked to its corresponding values of $\boldsymbol{h} = [h_1, h_2, h_3]^T$. Fig. 3.8 shows the scatter plot between the reduced space and the inputs h_2, h_3. Moreover, two clusters with different density are observed. The input h_1 is discarted by the criterium of Spearman Correlation coefficient (SpC) (Hauke and Kossowski, 2011). The dependences between the first princial component with respect to the inputs are:

- $SpC(\mathbf{y}, h_1) = 0.035$
- $SpC(\mathbf{y}, h_2) = 0.163$
- $SpC(\mathbf{y}, h_3) = -0.968$

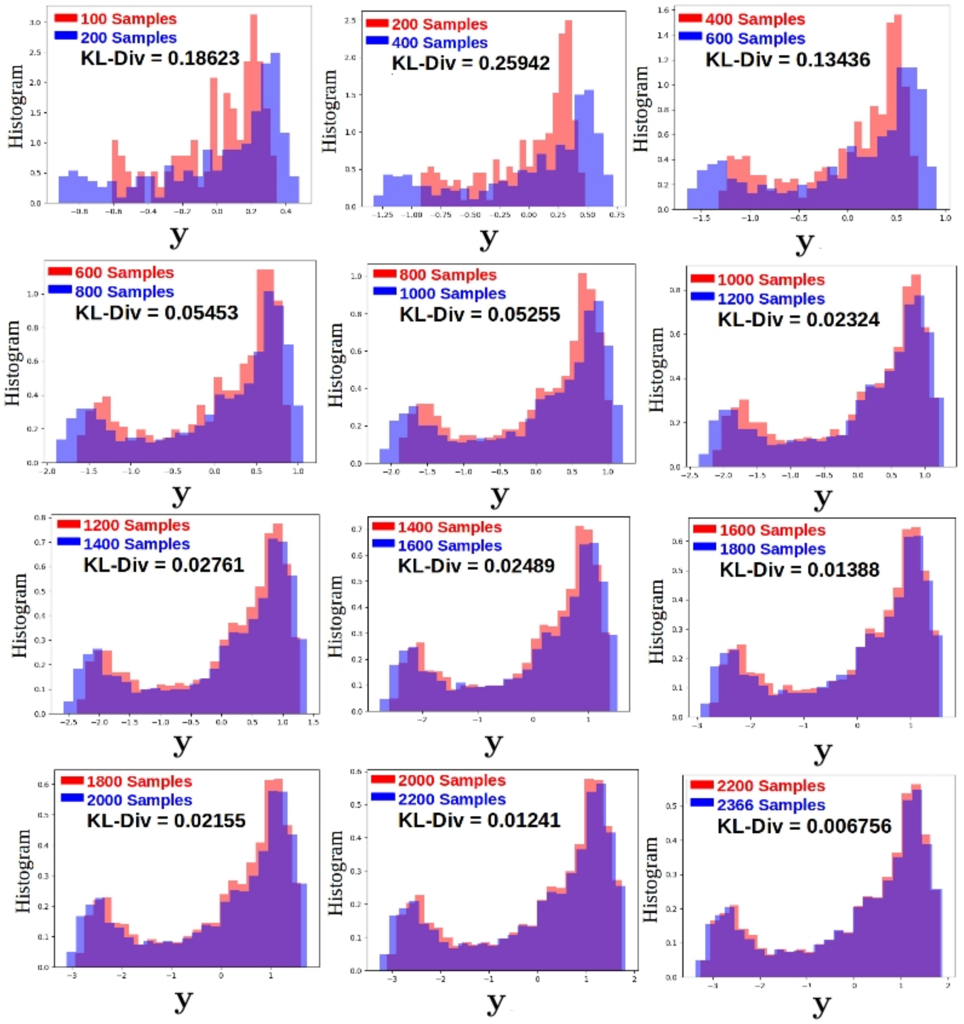


FIGURE 3.5 KL divergence evolution between histograms with different sampling size.

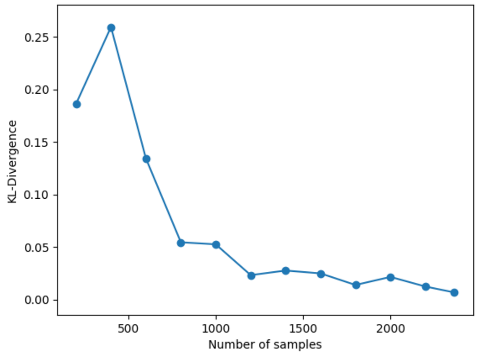
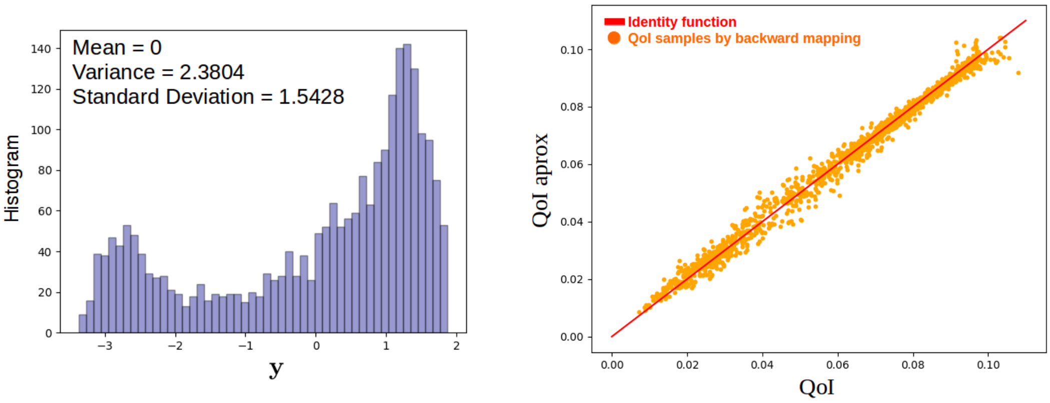


FIGURE 3.6 Evolution of KL divergence with respect the number of simulations.

Clearly, h_1 shows a very small correlation, and therefore is discarded for surrogate modeling.



(A) (B)

FIGURE 3.7 (a) Reference values of histogram, mean, variance and standard deviation of the first principal component $\mathbf{y}$ of kPCA. This reference values are achieved with 2366 simulations in VPS/Pamcrash. (b) Scatter plot around the identity function (red) of the QoI with respect to the approximated for the first principal component $\mathbf{y}$.

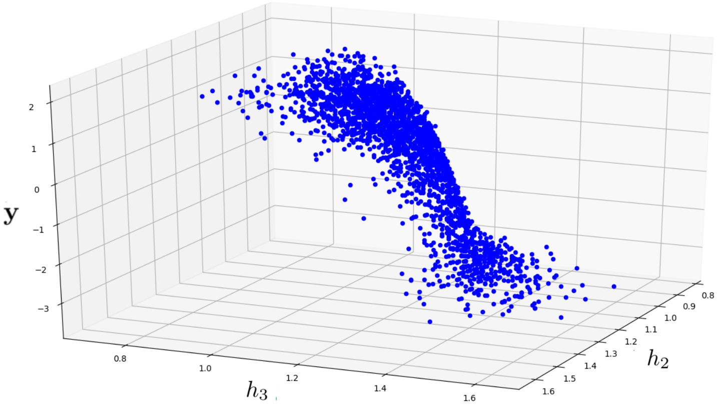


FIGURE 3.8 Scatter plot between the reduced space $\mathbf{y}$ and the inputs h_2, h_3.

Scattering plots for sensitivity analysis

In Fig. 3.9 it is shown the scattered plot between the reduced space $\mathbf{y}$ and the inputs h_2, h_3. Referring to Fig. 2.16, two modes of probability are observed. The input samples that are falling in the small mode are plotted in red and in the big mode in blue. It is illustrated that in plot (A) for any sample point h_2 we can obtain any response $\mathbf{y}$. Instead, in plot (B) it is observed a sigmoid shape, Any value of the input parameter h_3 is defined practically all the behaviour of the model modes. However, in the range $h_3 = [1.2 - 1.3]$ it is shown that the response $\mathbf{y}$ can fall in the red or blue area.

To tackle this uncertainty range, the parameter h_3, in Fig. 3.10 it is blocked to $h_3 = 1.3$. The green samples shows the behaviour of h_2 and $\mathbf{y}$ with an h_3 equal to 1.3 mm.

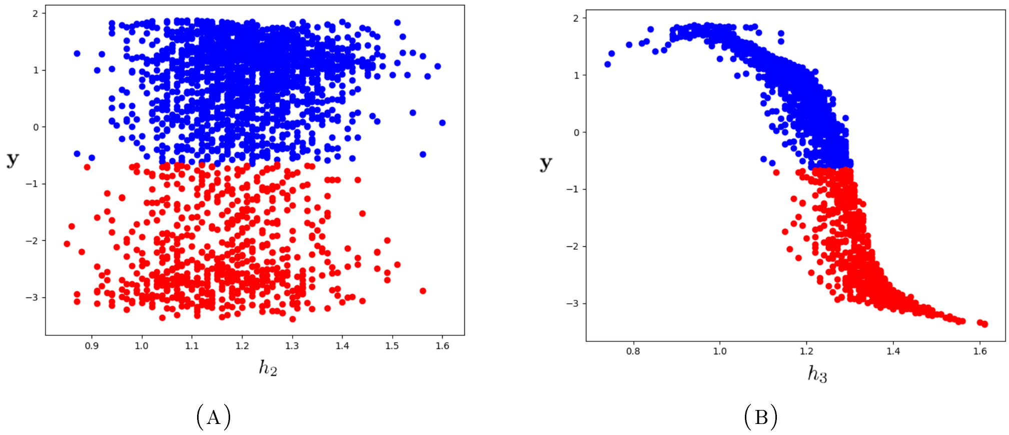


FIGURE 3.9 (A) Scattered plot $h_2, \mathbf{y}$ (B) Scatter plot $h_3, \mathbf{y}$. In red samples falling in the small mode and in blue samples falling in the big mode with respect to Fig. 2.16.

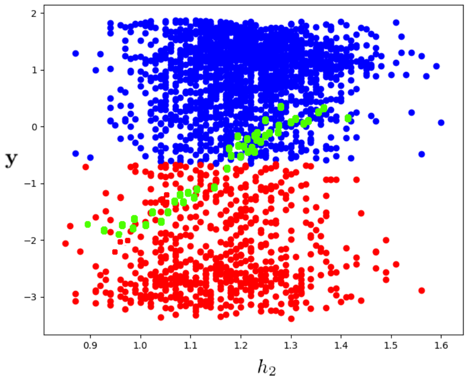
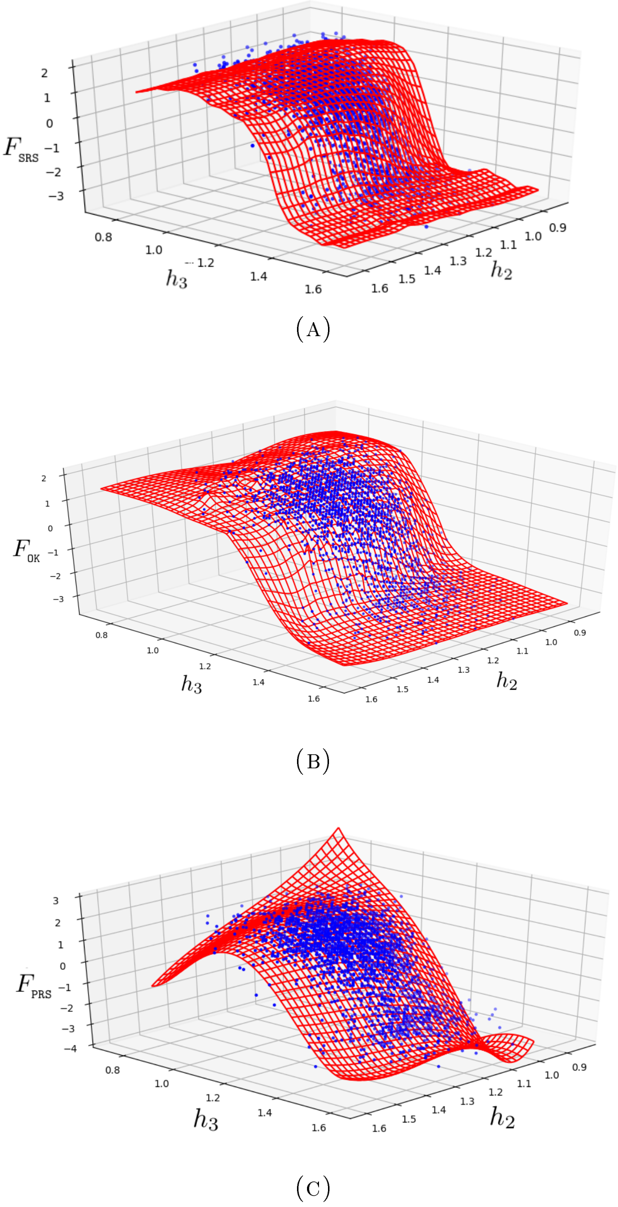


FIGURE 3.10 Scattered plot $h_2, \mathbf{y}$. The green points are the samples with the condition $h_3 = 1.3$ mm.

Fig. 3.10 shows a clear sigmoid behaviour between $h_2 - \mathbf{y}$ when the input h_3 is in the order of 1.3 mm (green samples). In consequence, the input parameter h_2 will define the the whole behaviour of a sample point to fall in the red or blue clusters when $h_3 = 1.3$.

3.6.3 Surrogate modeling

In Fig. 3.11 it is shown the response surface of SRS, OK and PRS metamodels between the first principal component of kPCA (storing 82.61% of information) and the inputs h_2, h_3. The metamodels $F_{\mathtt{OK}}$ and $F_{\mathtt{SRS}}$ show adaptive behaviour with the sample points (in blue). However, the F_{PRS} metamodel exhibits unstable tails in areas where there are few samples from the training set. This problem is called Runge's phenomenon and is common due to lack of sampling in the tails of the distributions. To evaluate the behaviour and the robustness of the response surfaces, each surface is evaluated increasing new random samples until 50000 points, aiming to compare histograms, means and standard deviations with respect to the reference values plotted in Fig. 3.7.



(A)

(B)

(C)

FIGURE 3.11 In red, is shown the response surfaces of (A) SRS, (B) OK and (C) PRS. In blue, the scattering samples.

In Fig.3.12 the histogram, mean, variance and standard deviation results of 50000 new random samples are illustrated for the evaluation of each metamodel. The three metamodels give approximations to the mean with an error around

5% of the Standard Deviation. Since the current reference mean (Figure 3.7a) is zero, both positive and negative values can be expected. The histograms of the $F_{\texttt{SRS}}$ and $F_{\texttt{OK}}$ metamodels show similar bimodal distribution with respect to the reference histogram illustrated in Fig. 3.7a. However, the two modes of distribution are not captured with the $F_{\texttt{PRS}}$ metamodel. The Runge's phenomenon of the response surface and a worse adaptation to the sampling points overlooks the two distribution modes.

In Fig. 3.13 the convergence of the three surrogate models are compared with the reference values while new random points for each metamodel are increased up. The results of this comparative study shows similar results in terms of mean and standard deviation for the three techniques. Meaning that these statistical values are not sensitive for the criterion to select the best surface. However, the results for the KL divergence clearly shows a worse behaviour for the PRS method caused for the tails of the response surface. In contrast, SRS and OK have similar results for the KL divergence where a very good performance is observed.

3.6.4 Uncertainty quantification for the surrogate model

Once the surrogates $F(\cdot)$ are available, for each input value $\boldsymbol{h}$, the corresponding $\mathbf{z}^{\star}$ is straightforwardly computed as $F(\boldsymbol{h})$. Then, the backward mapping produces the corresponding input vector $\mathbf{x}$ of plastic deformation values in the area of interest, being $l^{0}(\mathbf{x})$ its the associated QoI. The concatenation of the three operations is computationally negligible with respect to the cost of the training set of full order simulations. At this point, standard Monte Carlo is performed with 50000 new random samples for h_1, h_2 and h_3 to evaluate each metamodel. In Fig. 3.14 it is presented the corresponding PDFs of the QoI for the metamodels (SRS, OK and PRS). A bimodal function with approximately 19% of probability for the small mode and 81% for the big mode can be appreciated for SRS and OK. Otherwise, PRS fails to capture such behaviour. The statistical variables of the QoI for each metamodel are presented in Table 4.1. Here the three variables present similar results, which means that any metamodel captures similar information in terms of mean, variance and standard deviation.

Recalling that kPCA improves the mapping back to the original variable $\mathbf{x}$, a physical interpretation of the bimodal PDF can be performed. The corresponding behaviour of the structure for each mode of the PDF is illustrated in Fig. 3.15. Clearly, two physical modes are observed. The first snapshot of Fig. 3.15 shows the higher values of plastic strain and a significant back-bend of the plate profile. Otherwise, the second snapshot shows lower values of plastic deformation with a more rigid behaviour of the plate. The first physical case present 81% of probability and the second case 19% of probability occurrence.

Table 3.1 Statistical variables for each metamodel.

	Mean	**Variance**	**StD**
Reference values	0.0695	0.1546	0.0239
SRS	0.07245	0.1452	0.0211
OK	0.0659	0.15	0.0225
PRS	0.0707	0.1479	0.0219

3.7 Conclusions

A nonintrusive methodology to perform uncertainty quantification for crashworthiness problems is presented. The basic idea is to combine some dimensionality reduction technique (here kPCA) with a surrogate model based on a training set of full-order solutions (ideally not too many, because of their computational cost). The dimensionality reduction eases the task of the surrogate model and enables the analyst to detect clusters and categorize the data. The surrogate model (or metamodel) substitutes at a negligible computational cost the original full-order model. It therefore permits producing multiple queries to the model, corresponding the different parametric input values demanded by the Monte Carlo strategies.

In the benchmark problem considered, kPCA allows describing the full phenomenon with only one principal component, accounting for more than 82% of the total variance (that is, of the information). This problem is relevant in automotive engineering (and often used as benchmark by SEAT engineers) and, despite the fact that only three input parameters are assumed to have stochastic nature (and 3 dimensions are not awakening the *curse of dimensionality*), the dimensionality reduction is still pertinent to simplify the output of interest to be analyzed. Actually, using kPCA, only one principal component is accounting for more than 82% of the total information. It also detects two clusters corresponding to two deformation modes and two different levels of the QoI. The UQ methodology is also providing the probabilities of occurrence of these two modes, which are 19% and 81%. This is reflected in a bimodal PDF, one mode having a probability four times larger than the other. Moreover, using kPCA as dimensionality reduction strategy the backward mapping from the reduced space is more accurate and allows interpreting the mechanisms associated with these two modes.

In the presented methodology for a crash problem, the use of linear PCA is also a suitable option. On the one side, if a single scalar QoI is required for decision making, PCA is simpler than kPCA. On the other side, if it is necessary to find both an accurate QoI, as well as a more detailed approximation of the full original variable $\mathbf{x}$ map, corresponding to a complete deformation field, kPCA improves the mapping back to the original input space, since it accounts on the intrinsic nonlinearities involved in the manifold of training set. Thus, both PCA or kPCA can be used for this methodology, depending on the main objective sought. In this manuscript, for the reasons mentioned above, kPCA is used and described in more detail. This allows dealing with problems representing more complex phenomena, where data lies in highly nonlinear manifold.

Three formats of the surrogate models are taken into consideration, Ordinary Kriging (OK), Polynomial Response Surface (PRS) and a Separated Response Surface (SRS) approach, introduced here as novelty and based in the PGD methodology. The assessment of the mean and variance of the outcome (the QoI) is properly computed using the three alternative surrogates. However, when it comes to analyze the PDF (approximated by histograms), the SRS and OK surrogates perform much better than the PRS. The PRS surrogate fails to capture the bimodal character of the PDF.

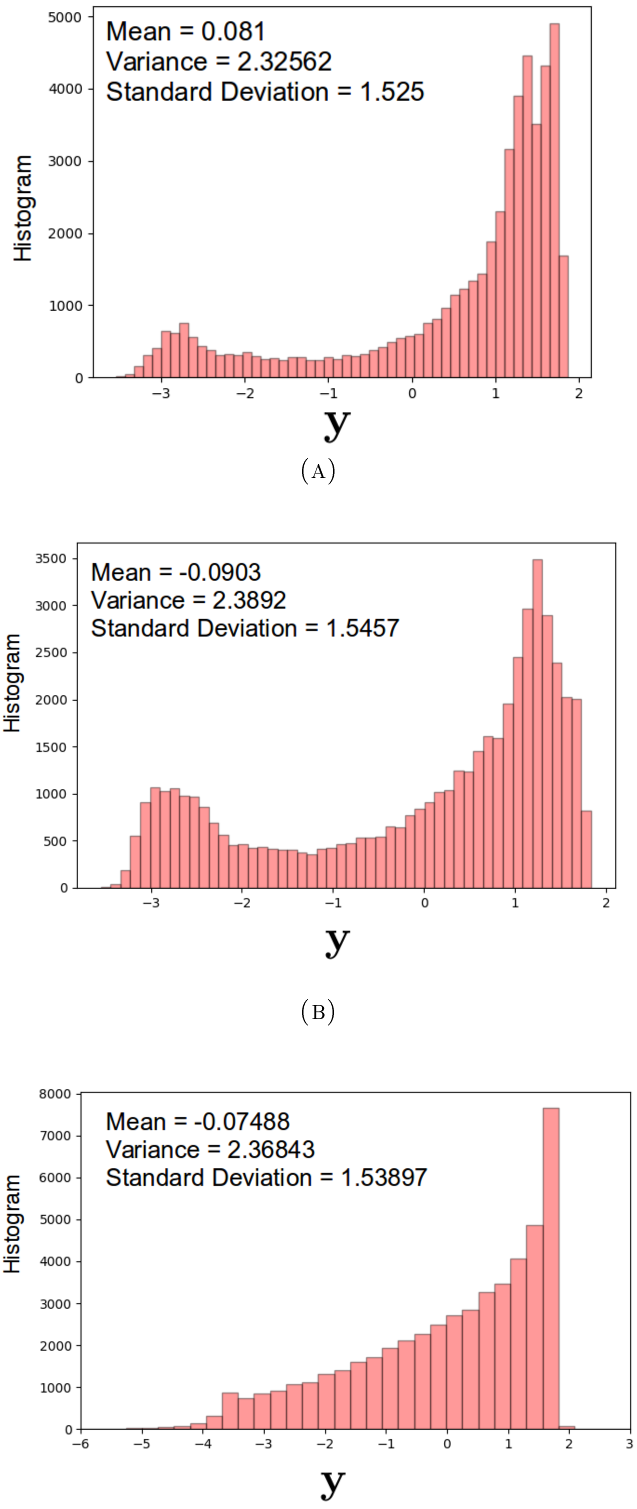
Being OK an interpolative methodology (the response surface passes through the data of the training test), it is pretty sensitive to the noise contained in the data. In the current examples, this is not an important issue, because the data is not particularly noisy. However, it may be relevant in other cases. SRS being a least-squares fitting it is not suffering of this drawback. Moreover, SRS is proposing an explicit parametric solution, therefore it can be used to compute derivatives or to integrate it analytically. This allows also to compute the statistical moments, probability density function and cumulative density function with analytical methods, circumventing the Monte Carlo sampling. Another interesting feature of the SRS is its fair scalability with the number of input parameters (stochastic dimension).

The combination of the kPCA manifold learning technique with the different surrogates offers an attractive framework to perform UQ in complex problems. Here, the application to parametric crashworthiness simulations open new perspectives. The available alternatives for the surrogates (in particular OK and

SRS) and the dimensionality reduction techniques at hand, are a powerful toolbox allowing to attack challenging problems in science and engineering.

The combination of dimensionality reduction and surrogate models produces accurate solutions at an affordable computational cost, accounting also for the uncertainty, that is assessing the credibility of the simulation. Particularly in the context of crashworthiness UQ, the computational cost is a key issue and a driving force for the research developments in the field. Obviously, increasing accuracy requires a higher computational effort. Finding a trade-off between these two factors is a daily concern for research engineers. This chapter intends to provide tools to achieve accurate and credible crashworthiness industrial simulations at an acceptable computational effort.

The proposed approach demonstrates promising results for UQ analysis in the crashworthiness framework. However, from a data-driven point of view, it is interesting to refocus the methodology proposed in this chapter towards a more autonomous algorithm for multi-purpose analyses. The next chapter takes into account the development of an adaptive autonomous application for engineer decision making for multi-purpose engineering analyses accounting for uncertainty.

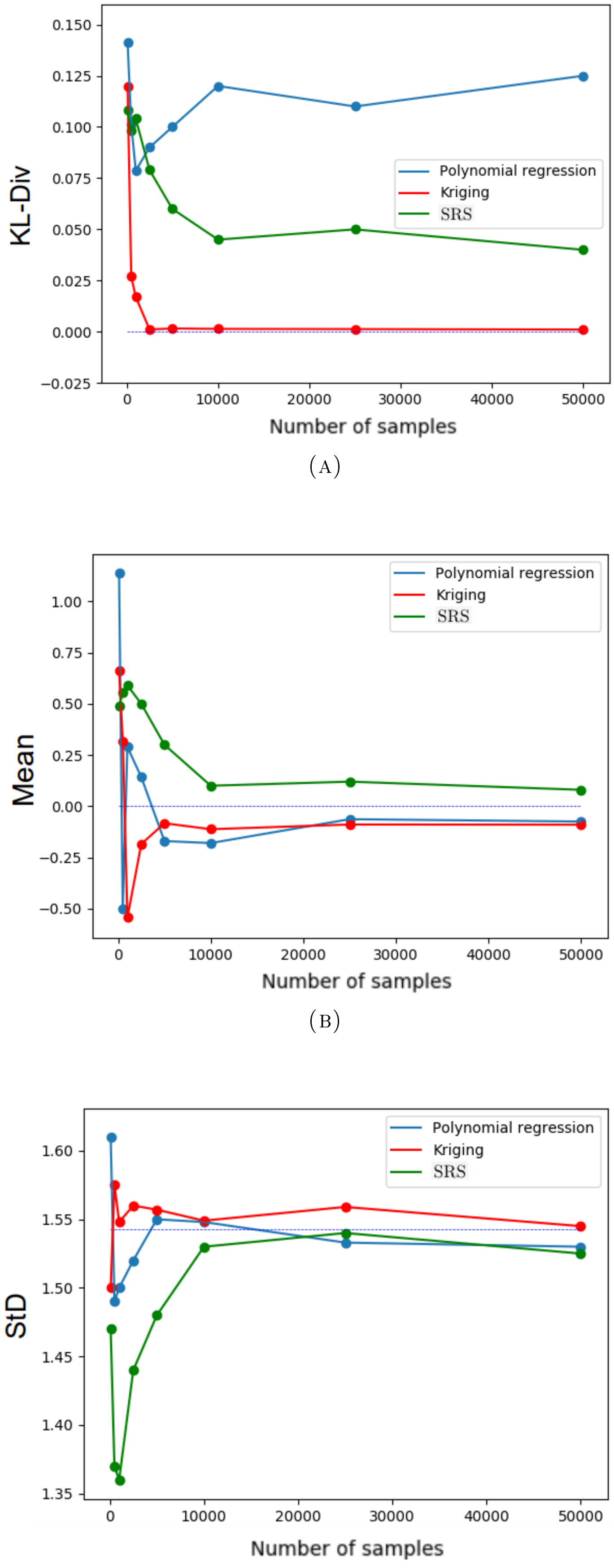


(A)

(B)

(C)

FIGURE 3.12 Histogram, mean and standard deviation results of the different surrogate models. (A) SRS, (B) OK and (C) PRS. The results are obtained by evaluating each surrogate model with 50000 new random samples.

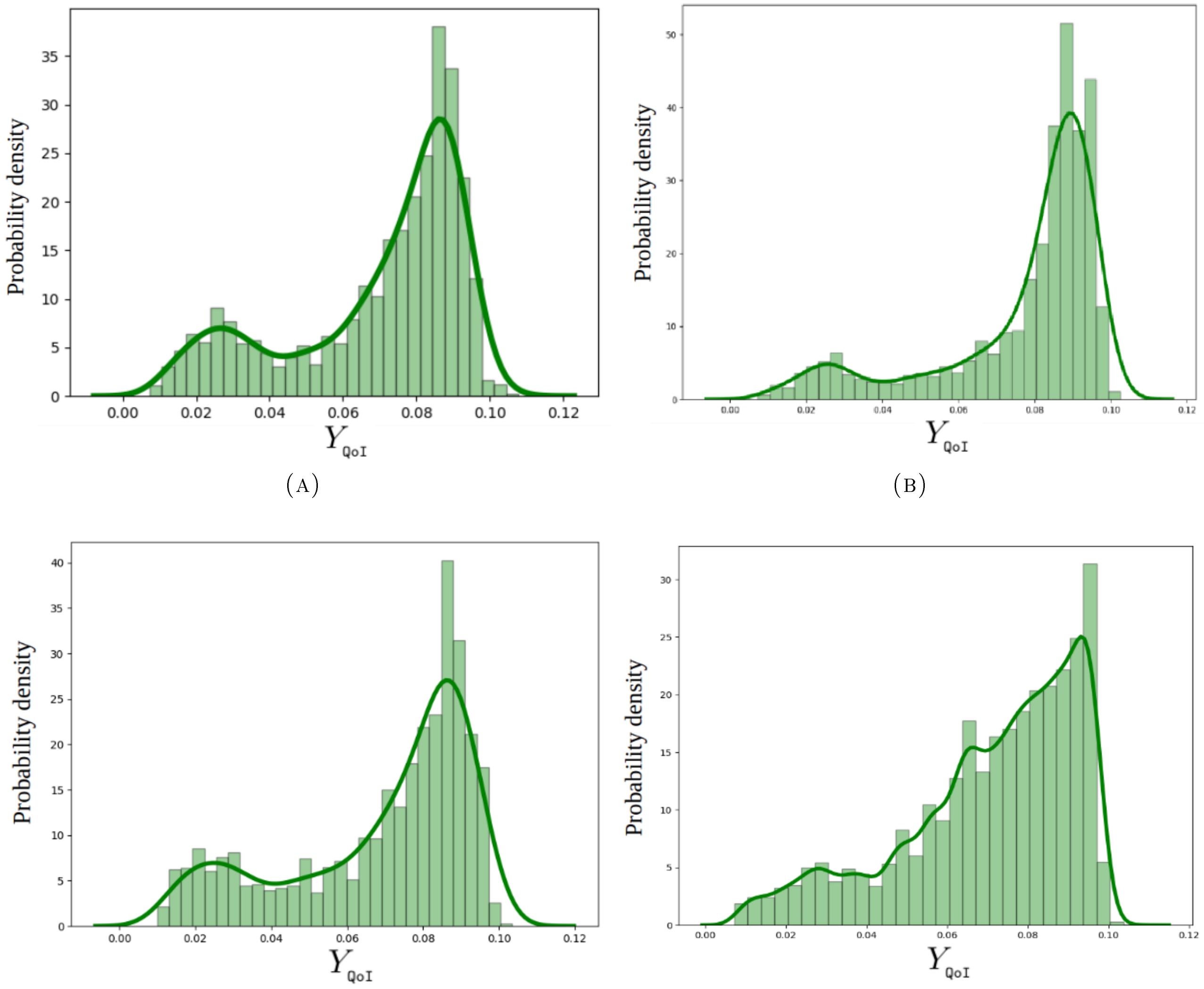


(A)

(B)

(C)

Figure 3.13 Convergence plots of SRS, OK and PRS evaluating KL divergence, mean and standard deviation with respect to the reference values(KL=0, mean=0, standard deviation=1.5428 plotted with the dashed line).



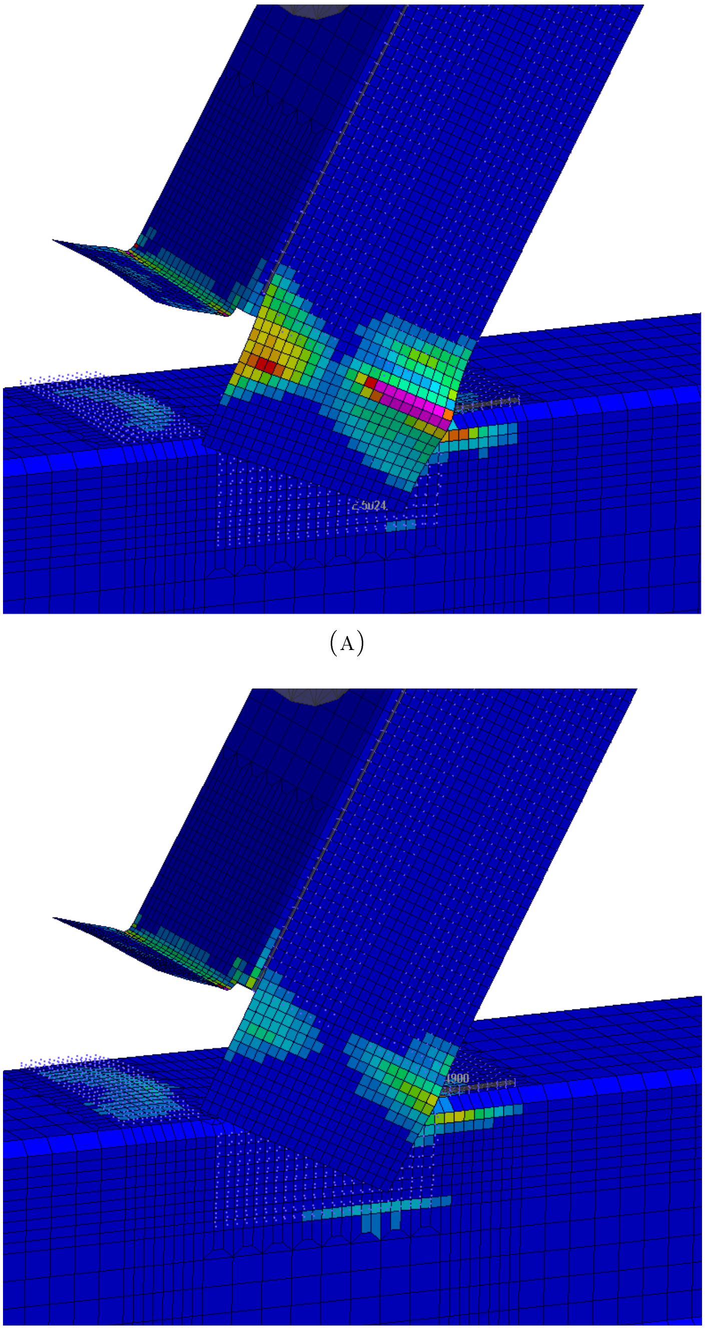
(A)

(B)

(C)

(D)

FIGURE 3.14 a) shows the histogram of the 2366 reference samples. b), c) and d) shows the histograms of the QoI by evaluating 50000 random samples for SRS, OK and PRS metamodels.



(A)

(B)

FIGURE 3.15 a) Snapshot simulation of the plastic strain in the biggest mode in the QoI histogram. b) Snapshot simulation of the plastic strain in the smallest mode in the QoI histogram.

Chapter 4

Adaptive UQ methodology for multi-purpose engineering analysis

Chapter 3 presents a novel UQ methodology for high dimensional outputs with non linear behaviours in the field of crashworthiness, combining dimensionality reduction and surrogate modeling. A dimensionality reduction technique is proposed to reduce the output data to a reduced space with lower dimension. The reduced space is compared with different training sets until achieve a KL convergence criterion. Then, different surrogate models are proposed to establish a relationship between the input space and the reduced space of outputs. Once the metamodel is achieved, the standard Monte Carlo is implemented to perform an UQ analysis within a negligible computational cost.

In this chapter is presented an methodology to evaluate only the necessary samples for the training set without losing precision for multi-purpose engineering analyses. The novelty approach is based in an adaptive strategy combining dimensionality reduction and surrogate modeling explained in Chapter 3 with a data-driven strategy.

The chapter is structured as follows: section 4.1 a brief introduction and motivation. In Section 4.2 a benchmark crash problem is presented. In Section 4.3 is provided the description of the adaptive methodology divided in main steps. In Section 4.4 it is presented the results of the benchmark problem for a vademecum of 3000 simulations and the results of the proposed methodology. Finally, Sections 4.5 closes the chapter with conclusions.

4.1 Introduction and motivation

Empathizing that for the crash industry each single model consume hours of CPU it is essential to reduce the number of simulations to a small set of training runs. Recently, Min Li proposed a sensitivity analysis methodology combining PCA and Kriging for models with high-dimensional outputs. The study was applied for the San Francisco coastal protection (Li, Wang, and Jia, 2020). Also J.B Nagel, J. Rieckermann and B. Sudret proposed a sensitivity analysis methodology applying PCA for the reduction of the outputs and Polynomial Chaos for surrogate modeling for an urban drainage model (Nagel, Rieckermann, and Sudret, 2017). All of these studies present strategies for dealing with dimensionality reduction and surrogate modeling for UQ. However, large number of simulations for the training set are needed, and in consequence high computational cost for the field of crashworthiness.

In this chapter, an adaptive (or levelled) methodology that combines dimensionality reduction and surrogate modeling for nonlinear complex models is proposed. Specifically, kPCA is used to reduce the high-dimensional outcomes to low number of components and OK to metamodel between the stochastic input space and the reduced space from kPCA. Quite apart from the terms of standard UQ, the methodology allows obtaining complementary multi-purpose information of the model. Different features of the model are obtained in this context with a negligible additional computational effort: structural modes associated with output data, sensitivity analysis (influence from perturbation of input parameters in the results), statistical assessment of various quantities of interest. The method provides an efficient and robust tool for decision making with the minimum evaluations of the full order model but guaranteeing precision. This adaptive strategy allows to evaluate only the necessary samples for the training set to optimize the computational cost. In order to demonstrate the performance of the proposed approach, a benchmark crash problem is studied. The literature contains different works in crashworthiness UQ field (Rocas et al., 2021; Rocas et al., 2020; Wang et al., 2018), where different UQ approaches are shown implementing techniques as Monte Carlo, Polynomial Chaos, Quasi Monte Carlo, dimensionality reduction and surrogate modelling.

4.2 Industrial application: The tapered model

In this section, a industrial benchmark problem is presented to validate the feasibility of the proposed UQ methodology.

4.2.1 Model description

In the field of crashworthiness, the B-pillar is a part of the structure of a car that plays an important role in passenger safety. The manufacture process is one of the keys to achieve a successful design. Initially, the B-pillars were designed by assembling monolithic parts with different strengths capabilities. The idea was to produce a substructure with variable strength depending on the external loads. Recently, this structure design has been improved by reducing the number of pieces and weight by introducing a tailored templed B-pillar with a variable hardness profile, that is variable mechanical properties. The *tailored tempering* manufacturing process results in a progressive hardness profile, as illustrated in Fig. 4.1. An austenitized sheet piece with a thickness of 1.5 mm is introduced to a tailored press. This press is divided in two halves. One half with a temperature of 40°C and the other half with 530°C. After a holding time of 20 seconds the piece is extracted and cooled down to room temperature. As a result, the piece has a progressive hardness profile. In Fig. 4.3 is illustrated the hardness curve. The problem of the tailored tempering process is to ensure certain reproducibility within a series production process. Random perturbations of the mechanical properties are inevitable in serial production. Thus, the resulting mechanical properties are affected by important uncertainties, to be modelled with material parameters of aleatoric nature. Controlling this stochastic process with robustness is a challenge for the industry, and represents a cumbersome task. Therefore, computational modelling can be a hard task due to its random behaviour.

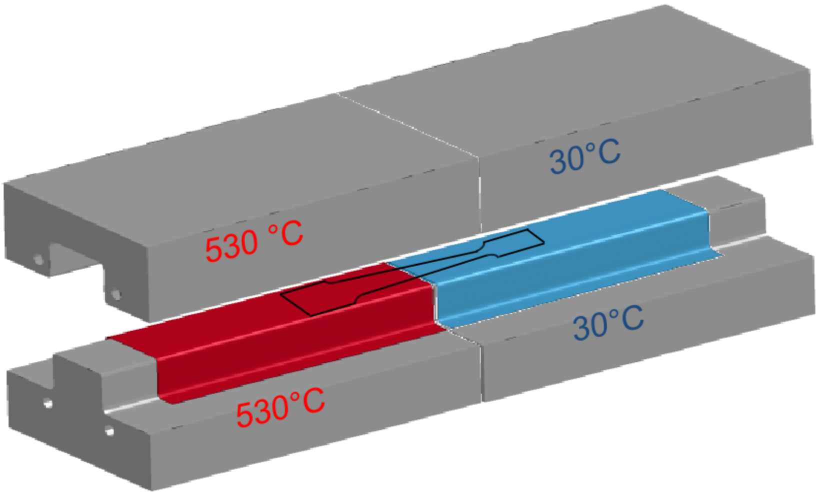


FIGURE 4.1 Tailored tempering process. Tailored press with two temperatures. The Right press with 40°C and the left press with 530°C.

For this research, the model is developed with explicit formulation in VPS/Pamcrash. Solving the equation 2.1 of transient dynamics explained in Chapter 2.

A simplified model is adopted here to demonstrate the strengths and capabilities of the proposed algorithms and data driven strategies. It models a tapered tensile specimen. With respect to a realistic B-pillar profile, this benchmark has much lower computational requirements while containing the essential features of the problem, allowing to account for the same conceptual difficulties and reproduce all the pertinent mechanisms. The geometry of the model is illustrated in Fig. 4.2. The structure is fixed in the right side. In the left side a uniform displacement of 7 mm in 40 ms (uniaxial load) is prescribed. The benchmark is modelled using the Belytschko-Tsay shell element with one integration point in the plane. The model has a total of 329 quadratic shell elements of 1.5 mm (thickness) and 384 nodes. A fracture model with a no element elimination configuration is implemented to guaranty the same number of elements for each simulation. For the time discretization it is used a time step of 0.2 ms.

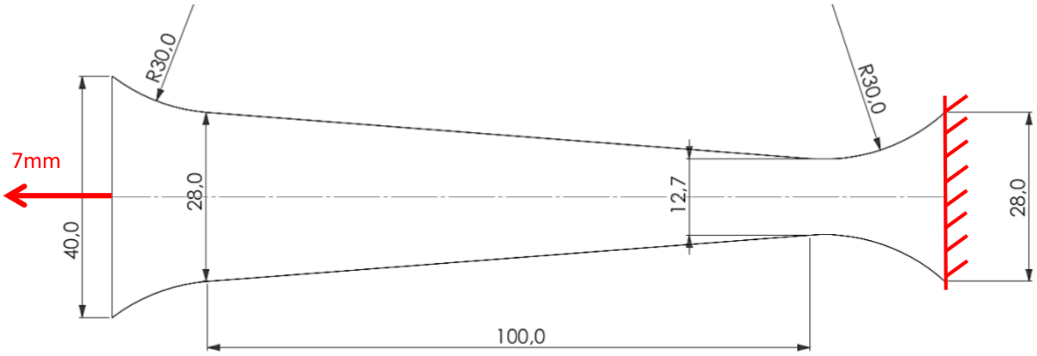


FIGURE 4.2 Geometry of the benchmark model.

The model is characterized with a Young Modulus of 200 GPa and a Poisson ratio of 0.3. Due to the manufacturing process explained above, the variability of the problem comes from the hardness curve of the material. To characterize the random behaviour of the curve, the position of the three points in Fig. 4.3 are taken as the uncertain input of the problem. Namely, Point 1=(h_1, h_4), point 2=(h_2, h_5) and point 3=(h_3, h_6). The six stochastic variables are collected in a vector of inputs $\boldsymbol{h} = [h_1, h_2, h_3, h_4, h_5, h_6]^T$. The random inputs are assumed to be uncorrelated with Gaussian distributions $h_i \sim \mathcal{N}(\mu_i, \sigma_i), i = 1, 2, ..., 6$. All the other parameters in the model are considered deterministic. In Table 4.1 the mean and standard deviation for each variable is described.

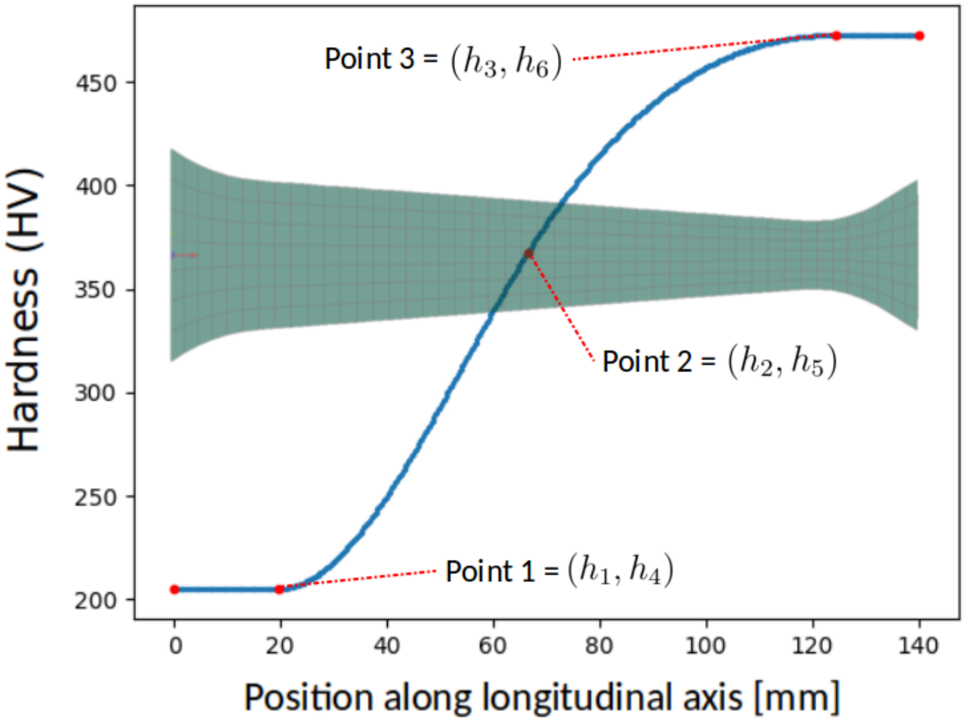


FIGURE 4.3 Hardness curve for the sheet piece through the manufacture process of heated and cooled press halves.

TABLE 4.1 Uncertainty variables.

Inputs h_i	**Mean** μ_i	**StD** σ_i
h_1	20	5.5%
h_2	70	5.5%
h_3	120	3%
h_4	212	5.5%
h_5	360	2.5%
h_6	460	3%

The output of the solution $\mathbf{U}$ of eq. 2.1 is characterized as a QoI vector $\mathbf{x}$ of dimension $d = 329$, corresponding to the values of the plastic strain in 329 shell elements. In practise, sampling the parametric input values and computing the full order model results in collecting different vectors of $\mathbf{x} \in \mathbb{R}^d$.

4.3 Adaptive UQ methodology

A novel UQ methodology for high dimensional outputs in the field of crashworthiness is proposed in the Chapter 3 (Rocas et al., 2021). The methodology combines Dimensionality Reduction (DR) and Surrogate Modeling (SM). The approach requires a specific number of evaluation (obtained with hierarchical KL-Divergence criterion) of the high order model for the UQ methodology. Then, a DR technique is proposed to reduce the output data to a reduced space with lower dimension and SM to establish a relationship between the input space and the reduced space. Once the metamodel is achieved, a standard Monte Carlo analysis is carried out to perform an UQ study with a negligible computational cost.

The main disadvantage of this approach is how to define the number of samples for the training set (to guarantee enough information for an UQ analysis). In industry, the size of the data set is defined based on computational resources, objectives, and the model. In terms of efficiency, this is an inappropriate approach to deal with this kind of problems, since the computational cost of the full order model is high, where each evaluation consume high resources and time. In this section it is presented an UQ methodology to evaluate only the necessary samples for the training set without losing precision.

To guarantee a robust design with a small number of evaluation of expensive models becomes a real challenge. The proposed methodology is developed to deal with the complex issue to quantify the uncertainty for crash problems, with the aim of minimizing computational cost, while preserving precision with an adaptive approach. In this section, the main steps of the proposed strategy are described. Fig. 4.4 presents the flowchart, and in the following subsections each step is explained in detail. A general overview of the main steps in the flowchart follows:

- The first step, called *A-Training set*, relies on the identification and characterization of the stochastic inputs to evaluate the expensive model in a set of training points. Then, the QoI of each simulation is stored in the output matrix $\mathbf{X}$.

- The second step, *B-Dimensionality reduction* is based on the kPCA dimensionality reduction technique applied to nonlinear data set. However, other techniques can be implemented. This step is intended to reduce the dimension of the output matrix $\mathbf{X}$. The reduced space of principal components allows to detect hidden structural modes and also avoids jeopardizing the metamodel approach.

- The third step, *C-Surrogate modeling* , corresponds to the development of a response surface between the reduced space, from kPCA, and the stochastic space of inputs. The metamodel allows to substitute the full order model to evaluate any new point in the reduced space and to map it backwards in the original space (García-González et al., 2020).

- The fourth step, *D-Parametric convergence quantification*, it is evaluated the metamodel with new Monte Carlo samples as a substitute of the expensive model. Sensitivity analysis of the input parameters (Sobol' indices) and clustering are performed in the enriched reduced space of kPCA. These indices are used in a stopping criterion to check the convergence and stability of the method.

- In the fifth step, *E-Uncertainty quantification*, the enriched space of kPCA is mapped backwards to perform uncertainty quantification of the input space. Statistical measures of the input space (QoI histograms, means, variances, standard deviations and free new simulations) are analyzed with negligible computational cost.

The first four steps (A, B, C, D) are implemented in an adaptive scheme with different levels of sampling sizes. The variance of the percentage of the clusters and the sensitivity indices of the input parameters are compared for each level aiming to analyze the stability of the problem outcome. If the stopping criterion is fulfilled, the method stops. If not, it goes to the next level with a new sampling size $\mathtt{n_s}$, for enriching the training set. This levelled approach is implemented until stability is achieved. The details of the adaptive stopping criteria are explained in Section 4.3.5.

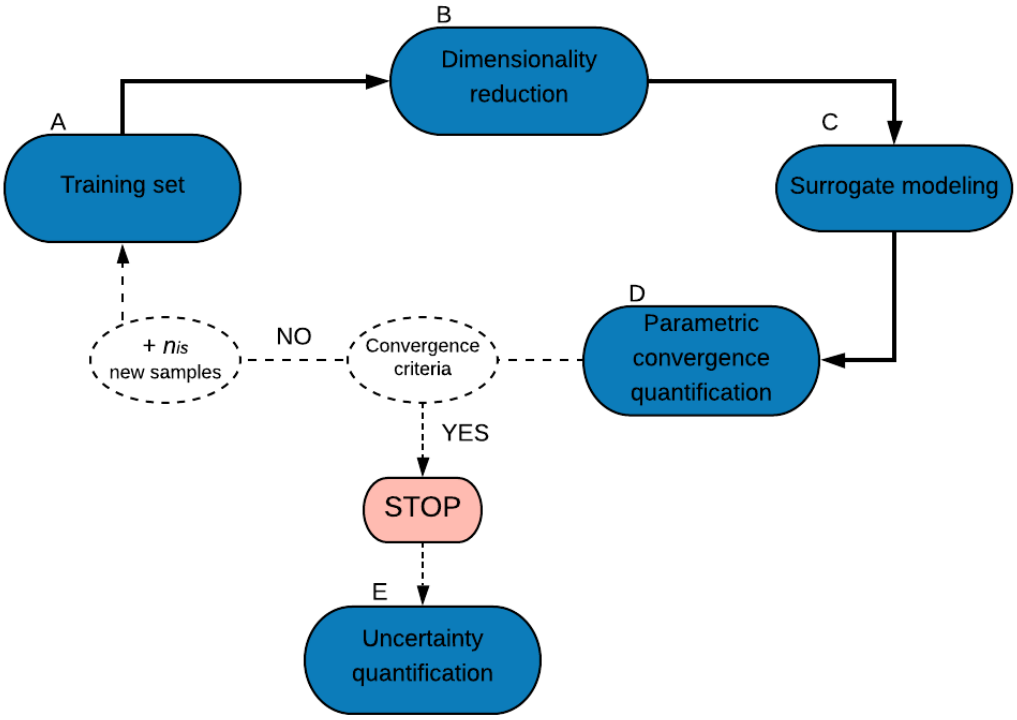


FIGURE 4.4 Flowchart of the adaptive UQ methodology

4.3.1 Training set

Let us consider a set of random variables describing the input parameters of the model by $\boldsymbol{h} = [h_1 h_2 \cdots h_{\mathtt{n_d}}]^T$. A Halton sampling technique (Wong, Luk, and Heng, 1997) is used to select $\mathtt{n_s}$ points (to build a discrete training set in the input space). The advantage of Halton sequence with respect other techniques as Monte Carlo, Hammersley sequences, among others, is the nested samples property for each resampling level size. The sampling points of the input space are stored in the matrix $\mathbf{H} = [\boldsymbol{h}^1 \boldsymbol{h}^2 \cdots \boldsymbol{h}^{\mathtt{n_s}}] \in \mathbb{R}^{\mathtt{n_d} \times \mathtt{n_s}}$. Each input vector $\boldsymbol{h}^i, i = 1, 2, ..., \mathtt{n_s}$ requires a single run in VPS/pamcrash. The idea is to evaluate $\mathtt{n_s}$ simulations of the expensive model to store the output responses in $\mathbf{X} = [\mathbf{x}^1 \mathbf{x}^2 \cdots \mathbf{x}^{\mathtt{n_s}}] \in \mathbb{R}^{d \times \mathtt{n_s}}$ as a training set. In the benchmark problem, each $\mathbf{x}^i$ collects the maximum plastic strain of all the elements of the model in the last time step. Fig. 4.5 illustrates the flowchart to sample the parametric space and obtain the training set.

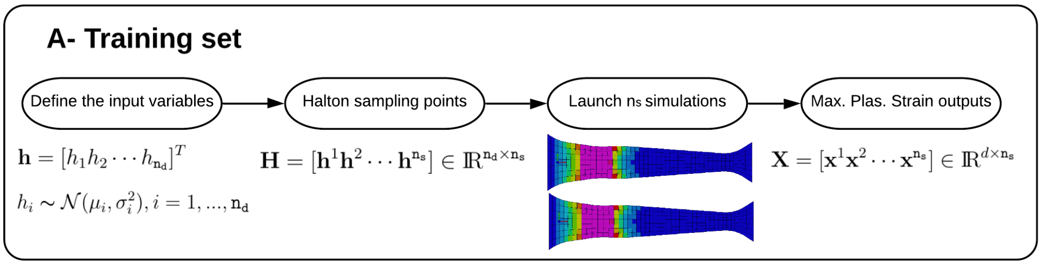

A- Training set
Define the input variables
Halton sampling points
Launch ns simulations
Max. Plas. Strain outputs
$\mathbf{h} = [h_1 h_2 \cdots h_{\mathtt{n_d}}]^T$
$h_i \sim \mathcal{N}(\mu_i, \sigma_i^2), i = 1, ..., \mathtt{n_d}$
$\mathbf{H} = [\mathbf{h}^1 \mathbf{h}^2 \cdots \mathbf{h}^{\mathtt{n_s}}] \in \mathbb{R}^{\mathtt{n_d} \times \mathtt{n_s}}$
$\mathbf{X} = [\mathbf{x}^1 \mathbf{x}^2 \cdots \mathbf{x}^{\mathtt{n_s}}] \in \mathbb{R}^{d \times \mathtt{n_s}}$

FIGURE 4.5 Flowchart of the training model.

4.3.2 kPCA dimensionality reduction

Analyze the training set and find the principal components allows to reduce de complexity of the problem. The dimensionality reduction technique kPCA is proposed for the reduction of the output matrix $\mathbf{X} = [\mathbf{x}^1 \mathbf{x}^2 \cdots \mathbf{x}^{\mathtt{n_s}}] \in \mathbb{R}^{d \times \mathtt{n_s}}$. It is of utmost importance consider that in the field of crashworthiness the data is nonlinear. The standard Principal Component Analysis (PCA) captures linear behaviours, however for this research kPCA is implemented for its non-linear ability and the extremely advantageous backward mapping to recover back the full-order object in as accurately as possible. In this aspect, kPCA behaves much better than PCA in many cases.

kPCA provides a useful tool to transform the original data from high dimensional space to a low dimensional space where the main features of the input data are kept. Considering the training set matrix $\mathbf{X} = [\mathbf{x}^1 \mathbf{x}^2 \cdots \mathbf{x}^{\mathtt{n_s}}] \in \mathbb{R}^{d \times \mathtt{n_s}}$ as the input matrix, the main objective is to find a low dimensional space, where the first k principal components retain most of the information to capture the data behaviour. For this, a nonlinear mapping function $\mathbf{\Phi}(\mathbf{x})$ is needed, where in general it is unknown. However, the most used kernel functions are:

- Gaussian kernel: $\kappa(\mathbf{x}^i, \mathbf{x}^j) = e^{-\beta \|\mathbf{x}^i - \mathbf{x}^j\|^2}$
- Linear kernel: $\kappa(\mathbf{x}^i, \mathbf{x}^j) =< \mathbf{x}^i, \mathbf{x}^j >$
- Polynomial kernel: $\kappa(\mathbf{x}^i, \mathbf{x}^j) = (< \mathbf{x}^i, \mathbf{x}^j > +b)^p$

Collecting a reduced number of terms with enough pieces of information allows to reduce the number of metamodels for the feature space $\mathbf{z}^\star$, and in consequence, the computational cost. This combination is problem dependent and refit kPCA multiple times to compare different kernels and parameters with a optimization function is needed. Also, if the data has different behaviours and the mapping function is appropriate, then the reduced space $\mathbf{z}^\star$ is a sensitive measure for cluster detection (structure modes).

In practice, this boils down to apply kPCA and determine a mapping function $\mathcal{G}(\mathbf{x})$ between the solutions $\mathbf{x} \in \mathbb{R}^d$ and some new variable $\mathbf{z}^\star \in \mathbb{R}^k$ in a much lower-dimensional space ($k \ll d$). The set of eigenvalues provides the criterion to choose the number of terms k to be retained for the reduction (for the benchmark problem, 90% of information must be retained), leading to,

$$\mathbf{z}^\star = \mathcal{G}(\mathbf{x}). \tag{4.1}$$

The mapping between $\mathbf{x}$ and $\mathbf{z}^\star$ is to be characterized forward and backward as $\mathbf{x}^\star = \mathcal{G}^{-1}(\mathbf{z}^\star) \approx \mathbf{x}$. There exist different techniques available in the literature (Zheng, Lai, and Yuen, 2010; Wang, 2012). For this research, it is implemented a technique based on a minimization of the discrepancy functional (residual) (García-González et al., 2020).

The proposed weighting distance technique (García-González et al., 2020) associates weights to each value in the calculation based on the distance between the samples and the input sample $\mathbf{x}$. Let d_i for $i = 1, ..., \mathtt{n_s}$ be the squared distances $d_i = \|\mathbf{z}^\star - \mathbf{z}^{\star i}\|$. Where w_i are the weights defined by,

$$w_i = \frac{1/d_i^2}{\sum_{j=1}^{\mathtt{n_s}} 1/d_j^2}. \tag{4.2}$$

Here, the inverse of the squared distances ($1/d^2$) is used to define the weights, following (García-González et al., 2020). Any other decreasing function of the distance is admissible, to account for the influence of the distance in the weights. Any version of the radial-based interpolation is commonly used to construct surrogate models based on samples from a training set.

For a $\mathbf{z}^\star$ point in the reduced space it corresponds a $\mathbf{x}$ point in the original space defined by:

$$\mathbf{x} \approx \sum_{i=1}^{\mathtt{n_s}} w_i \mathbf{x}^i. \tag{4.3}$$

This technique allows to backward any point from the reduced space $\mathbf{z}^\star$ to the original space. In Fig. 4.6 it is illustrated the flowchart for the dimensionality reduction step.

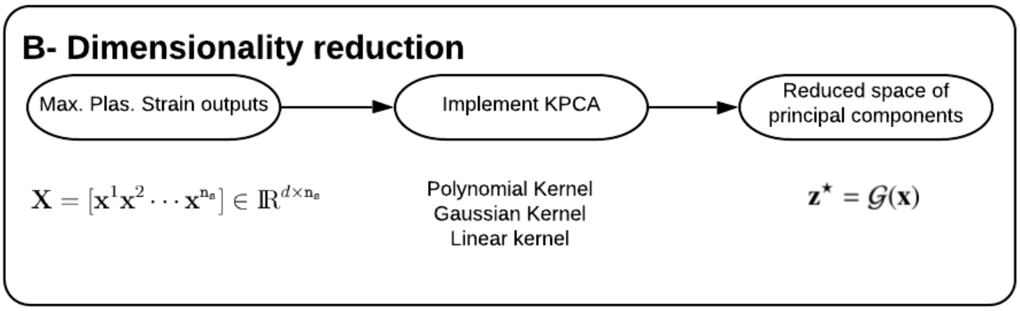

B- Dimensionality reduction
Max. Plas. Strain outputs
Implement KPCA
Reduced space of principal components
$\mathbf{X} = [\mathbf{x}^1\mathbf{x}^2 \cdots \mathbf{x}^{\mathrm{n_s}}] \in \mathbb{R}^{d\times \mathrm{n_s}}$
Polynomial Kernel
Gaussian Kernel
Linear kernel
$\mathbf{z}^\star = \mathcal{G}(\mathbf{x})$

FIGURE 4.6 Flowchart of the dimensionality reduction step.

4.3.3 Surrogate modeling

In crashworthiness, it is common to have high dimensional output matrices. In consequence, it is unaffordable to construct a surrogate model in terms of computational cost to deal for an efficient approach.

The main idea of this surrogate modeling step is constructing a response surface $\mathbf{z}^\star = F(\boldsymbol{h})$, from input $\boldsymbol{h}$ to the reduced space $\mathbf{z}^\star$. Here, the surrogate technique is presented to establish a functional dependency among some input $\boldsymbol{h}$ and some output function $\mathbf{y}(\boldsymbol{h})$ (typically, a post-process or reduced model of $\mathbf{x}^\star$).

A scalar output Y is considered for any of the components of $\mathbf{y}(\boldsymbol{h})$. For the benchmark problem, Y corresponds to the first principal component of kPCA, that is $Y = [\mathbf{z}^\star]_1 = [y^1 y^2 \cdots y^{\mathrm{n_s}}]^T$, where $y^i, i = 1, 2, ..., \mathrm{n_s}$ are the points of the reduced space .

The functional dependence $Y = F(\boldsymbol{h})$ is determined from the data provided by the training set, and the dimensionality reduction space. The metamodel function $F(\boldsymbol{h})$, approximates for any input $\boldsymbol{h}$ the corresponding image y^i in the reduced space. Then the backward mapping explained in Section 4.3.2 returns to the input sample by $\mathbf{x}^\star \approx \mathcal{G}^{-1}(F(\boldsymbol{h}))$.

For this research, Ordinary Kriging (OK) is used for metamodeling. In the literature different papers regarding kriging metamodeling (Oliver and Webster, 2014; Rocas et al., 2021). Other surrogate modelling technique can be implemented as Polynomial Chaos or Separated Response Surface (Rocas et al., 2020). However, kriging shows a better performance for crash modelling (Rocas et al., 2021). The purpose is to evaluate the metamodel with new $\mathtt{n_{MC}}$ realizations to estimate new $\mathbf{z}^\star$ values to enrich the reduced space for a posteriori UQ analysis. Here it is presented a brief review of OK for the methodology.

OK is an interpolation surrogate method that determines weights for a set of sample points to obtain a prediction of a new input. The weights are based on a variogram model that has the main advantage of estimating different variances for any distance between a pair of samples. The kriging metamodel $F(\boldsymbol{h})$ of any point $\boldsymbol{h}$ is defined by:

$$F^j(\boldsymbol{h}) = \sum_{i=1}^{\mathtt{n_s}} w_j^i(\boldsymbol{h})[\mathbf{y}^i]_j, j = 1, 2, ..., k. \tag{4.4}$$

The unknowns w are the weights and $[y^i]_j$ are the scalars of the principal component j of kPCA. This means that for each dimension of $\mathbf{z}^\star$ is needed a particular surrogate model. Therefore, the first k terms of the feature space determines the number of metamodels needed for the approach. The main condition with OK with respect other Kriging approaches is that the sum of weights is equal to 1. For more theoretical details of OK see (Oliver and Webster, 2014). In Fig. 4.7 it is illustrated the scheme for the surrogate modeling block.

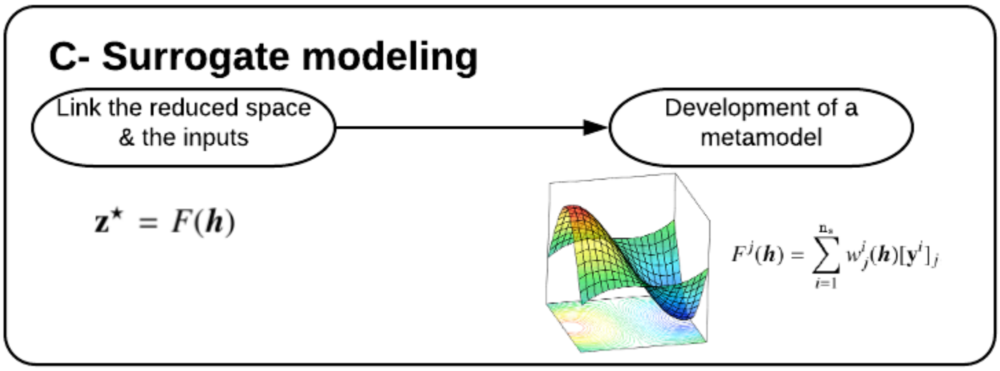

C- Surrogate modeling
Link the reduced space & the inputs
Development of a metamodel
$\mathbf{z}^\star = F(\boldsymbol{h})$
$F^j(\boldsymbol{h}) = \sum_{i=1}^{\mathtt{n_s}} w_j^i(\boldsymbol{h})[\mathbf{y}^i]_j$

FIGURE 4.7 Flowchart of the surrogate modeling.

4.3.4 Parametric convergence quantification

Optimization and redesign is a common task in crashworthiness, where detecting the main structure modes and the principal parameters could provide a useful tool for the engineers. For that, once the surrogate model $F(\boldsymbol{h})$ is available, for each input value $\boldsymbol{h}$, the corresponding $\mathbf{z}^\star$ is computed as $F(\boldsymbol{h})$. At this point, standard Monte Carlo is performed with $\mathtt{n}_{\mathtt{MC}} = 10^5$ new random samples of $\boldsymbol{h}$ to evaluate the surrogate model. The operations are computationally negligible with respect to the cost of the training set of the full order simulations. Therefore, sensitivity and statistical measures are easily performed.

In order to quantify the parametric uncertainty of the problem, sensitivity analysis of the input parameters (Sobol' indices) and clustering (structure modes) are performed in the enriched reduced space $\mathbf{z}^\star$ of kPCA with new $\mathtt{n_{MC}} = 10^5$ samples. Here, the clustering technique K-means (Likas, Vlassis, and Verbeek, 2003) is implemented for the reduced space $\mathbf{z}^\star \in \mathbb{R}^{k\times \mathtt{n_s}}$ in order to detect clusters in the data. However, other cluster techniques can be implemented (Saxena et al., 2017) . On the other hand, Sobol' indices are implemented to characterize the influence of the inputs to the outputs.

The essence of Sobol' indices is based on a variance decomposition of the feature space Y. We can define the total variance of Y as Var_Y. Therefore, Var_Y can be decomposed into partial variance associated with the inputs $\boldsymbol{h}$ as $Var_Y = \sum_{i=1}^{\mathtt{n_d}} Var_i + \sum_{i=1}^{\mathtt{n_d}-1}\sum_{j=i+1}^{\mathtt{n_d}} Var_{ij} + ... + Var_{1,...,\mathtt{n_d}}$, where Var_i denotes the variance contribution of the parameter h_i. While the other terms make reference with high order of interaction between inputs $\boldsymbol{h}$. Two sensitivity measures provide the Sobol' indices:

- Parameter influence ranking
- Identification of negligible parameters

For this study three types of Sobol' indices are calculated:

1. *First order Sobol' Index:*

   The first order sensitivity index S_i measures the single effect of the input h_i on the output variance of the model (Sobol, 1993; Saltelli et al., 2010). S_i is defined as

   $$S_i = \frac{Var_i}{Var_Y} = \frac{Var_i[\mathbb{E}_{\sim i}(Y|h_i)]}{Var_Y}, \tag{4.5}$$

   where the conditional expectation $\mathbb{E}_{\sim i}(Y|h_i)$ denotes the expected value of the output Y when the input h_i is fixed.

2. *Second order Sobol' Index:*

   The second order sensitivity index S_{ij} measures the interaction between h_i and h_j (Sobol, 1993; Saltelli et al., 2010). S_{ij} can be computed as

   $$S_{ij} = \frac{Var_{ij}}{Var_Y} = \frac{Var_{ij}[\mathbb{E}_{\sim ij}(Y|h_i,h_j)]}{Var_Y} - S_i - S_j, \tag{4.6}$$

   where the conditional expectation $\mathbb{E}_{\sim i}(Y|h_i,h_j)$ is the expected value of the output Y when the two input h_i and h_j are fixed.

3. *Total order Sobol' Index*:

   The Total sensitivity index S_{Ti} is called the "Total effect" of a input parameter h_i (Nagel, Rieckermann, and Sudret, 2017). This index includes the effect of the first order indices and the effects between the input h_i and all the possible combinations with the other inputs (Homma and Saltelli, 1996). It is defined as

$$S_{Ti} = 1 - \frac{Var_{\sim i}[\mathbb{E}_i(Y|h_{\sim i})]}{Var_Y}, \tag{4.7}$$

where $h_{\sim i}$ refers to all the inputs except h_i. For instance, $h_{\sim 1} = h_2, h_3, h_4, h_5, h_6$.

For the calculation of the conditional expectation of the Sobol' indices there exist different estimators in order to optimize the number of model evaluations (Saltelli et al., 2010). For this research, Saltelli algorithm (Kucherenko and Song, 2017) is implemented to calculate the conditional expectation. In addition, it is important to remark that Sobol' indices are computed for all principal components of $\mathbf{z}^\star \in \mathbb{R}^k$. In Fig. 4.8 it is illustrated the main ideas for the parametric convergence quantification step.

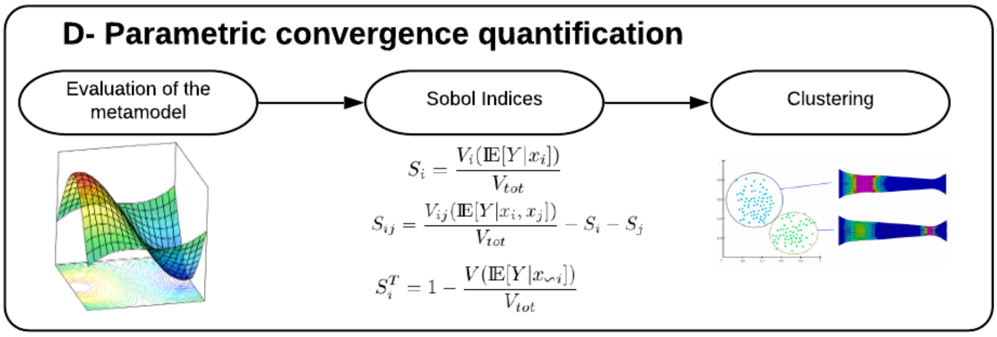

D- Parametric convergence quantification
Evaluation of the metamodel
Sobol Indices
Clustering
$S_i = \frac{V_i(\mathbb{E}[Y|x_i])}{V_{tot}}$
$S_{ij} = \frac{V_{ij}(\mathbb{E}[Y|x_i, x_j])}{V_{tot}} - S_i - S_j$
$S_i^T = 1 - \frac{V(\mathbb{E}[Y|x_{\sim i}])}{V_{tot}}$

FIGURE 4.8 Flowchart of the parametric convergence quantification step.

4.3.5 Autonomous stopping criteria

For the proposed adaptive UQ approach is required a stopping criteria. The variables used to analyze the convergence are both, the Sobol' indices and the percentage of the clusters. These two variables are proposed for the importance in the field of crashworthiness to detect structure modes and main parameters. However, any other sensitive measures can be implemented as a stopping criteria for the reduced space $\mathbf{z}^\star$ (e.g. mean, variance, standard deviation, histograms) depending on the problem.

The strategy is based in a comparative approach between levels of different sampling size. The size of the training sets for each level are defined by $\mathtt{n}_\mathtt{s}^\ell =$

$\mathtt{n_{con}}\ell$ with $\ell = 1, 2, ..., L$. Where $\mathtt{n_{con}}$ is a constant variable of the number of simulations. This parameter depends on the problem and is defined by the user. For the benchmark problem $\mathtt{n_{con}} = 10$, which leads to obtain the levels:

- $Level_1 \rightarrow \mathtt{n}_\mathtt{s}^1 = 10$
- $Level_2 \rightarrow \mathtt{n}_\mathtt{s}^2 = 20$
- $Level_3 \rightarrow \mathtt{n}_\mathtt{s}^3 = 30$

  $\vdots$
- $Level_L \rightarrow \mathtt{n}_\mathtt{s}^L = 10L$

Referring to the flowchart in Fig. 4.9, the methodology starts with $Level_1$ and the computation of the steps A, B, C, D. Then, $Level_2$ is launched with the computation of the steps with a new sampling size $\mathtt{n}_\mathtt{s}^2$. For the new sampling size of each level it is reused the simulations of the previous level, aiming to evaluate the minimum number of simulations of the full order model. The first levels are launched until s levels (for the benchmark problem $s = 5$). In the last level ($Level_s$) the variance of the percentage of the modes and the Sobol' indices of the previous s levels are analyzed. If the variances of the stopping variables (Sobol' indices and cluster percentage) are sufficiently small with respect to a stopping variance condition, the method stops. If not, the approach goes to a next level with the new increment of sampling size. The stopping variables are always analyzed for the last s levels, meaning that the parameter s defines how stationary is the solution. The process is repeated for each new level until the stability is achieved. In addition, it is necessary to achieve these stopping conditions for each dimension of the reduced space $\mathbf{z}^\star \in \mathbb{R}^k$.

4.3.6 Uncertainty quantification

Once the methodology is converged, the backward mapping returns $\mathbf{z}^\star$ to the corresponding input space $\mathbf{x}$ through $\mathbf{x}^\star = \mathcal{G}^{-1}(\mathbf{z}^\star)$, aiming to develop uncertainty quantification of the input space.

Uncertainty quantification of high-dimensional objects like $\mathbf{x}$ is cumbersome and the outcome is difficult to use as a tool supporting decision making. In that sense, the stochastic assessment focuses in a low-dimensional (even purely scalar) QoI, rather than in a high-dimensional object like $\mathbf{x}$. A specific QoI is introduced as an essential indicator for decision making.

For this study, the specific QoI is the average plastic strain of the input vector $\mathbf{x}$. However, any QoI function can be analyzed depending of the purpose

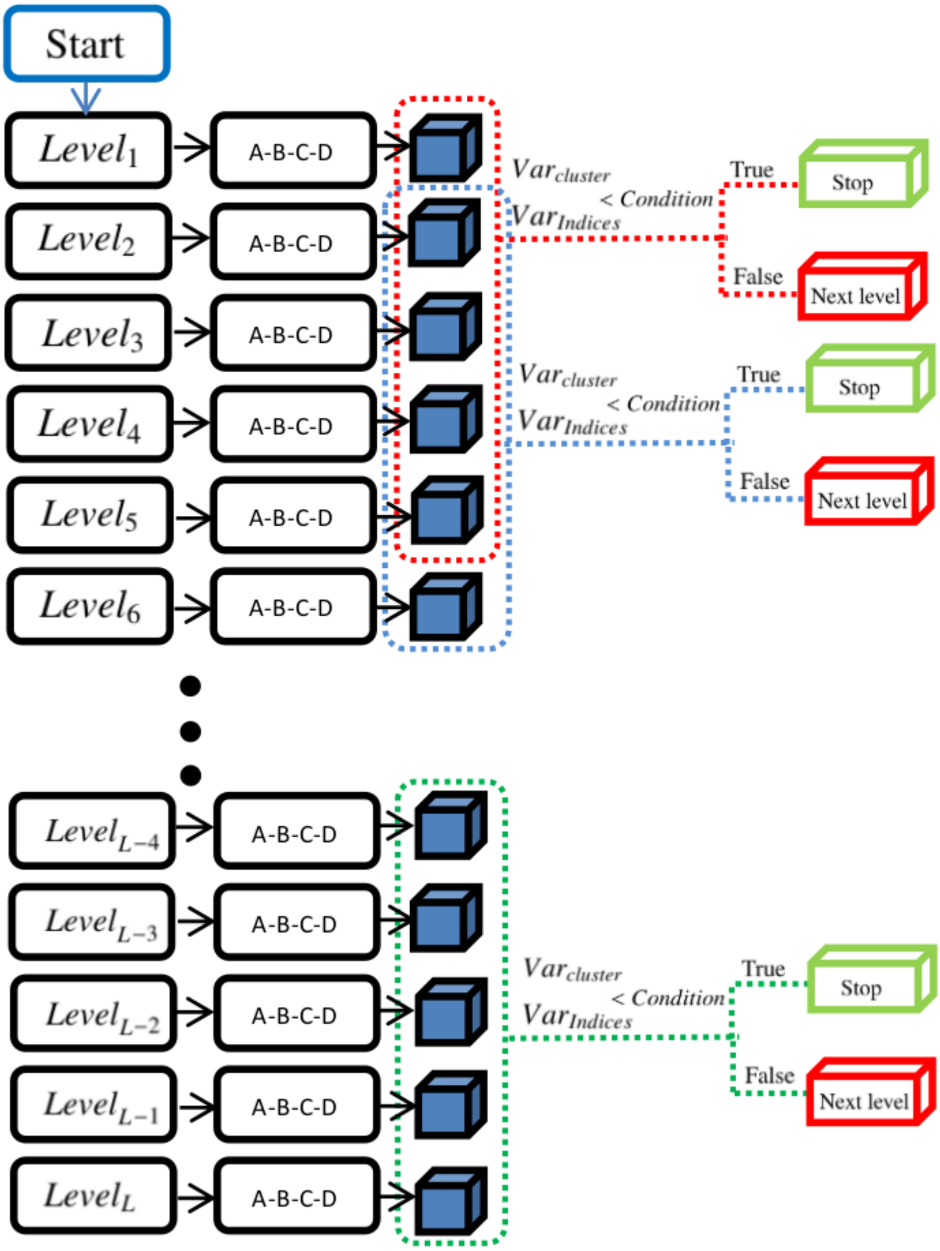


FIGURE 4.9 Flowchart scheme of the adaptive stopping criteria.

of the problem. Here, it is represented by a function form $l^0(\cdot)$, and for each $\boldsymbol{h}^i$ and $\mathbf{x}^i$ reads

$$l^0(\mathbf{x}^i) = \frac{1}{d}\sum_{j=1}^{d} x_j^i. \tag{4.8}$$

Statistical measures of $l^0(\mathbf{x})$ can be performed (e.g. mean, variance, standard deviation).

On the other hand, the fact that the model order reduction strategy is able to recover back the full-order object $\mathbf{x}^\star$, is extremely advantageous to represent new simulations with negligible computational cost. Therefore, for any combination of the input parameters $\boldsymbol{h}$ it is able reproduce the solution of the vector $\mathbf{x}$ for the physical model. In Fig. 4.10 it is illustrated the main ideas for the uncertainty quantification step.

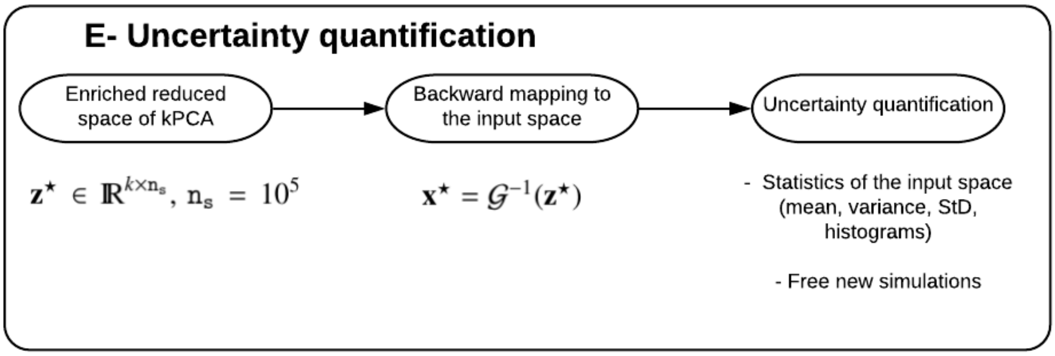

E- Uncertainty quantification
Enriched reduced space of kPCA
Backward mapping to the input space
Uncertainty quantification
$\mathbf{z}^\star \in \mathbb{R}^{k \times n_s}$, $n_s = 10^5$
$\mathbf{x}^\star = \mathcal{G}^{-1}(\mathbf{z}^\star)$
- Statistics of the input space (mean, variance, StD, histograms)
- Free new simulations

FIGURE 4.10 Flowchart of the uncertainty quantification step.

On the other hand, for a better understanding of the previous sections, a more detailed overview of the adaptive method is presented in Fig. 4.11. There are illustrated the 5 most important steps (A, B, C, D, E in navy blue) and the most important information derived from them.

4.4 Industrial Benchmark results

In this section the proposed methodology is implemented for the benchmark problem. The numerical results are divided in two sub-sections: i) The *Reference results*, where the benchmark problem has been evaluated with a vademecum of 3000 simulations, and ii) *UQ adaptive results*, where it is evaluated the methodology and compared with the reference results.

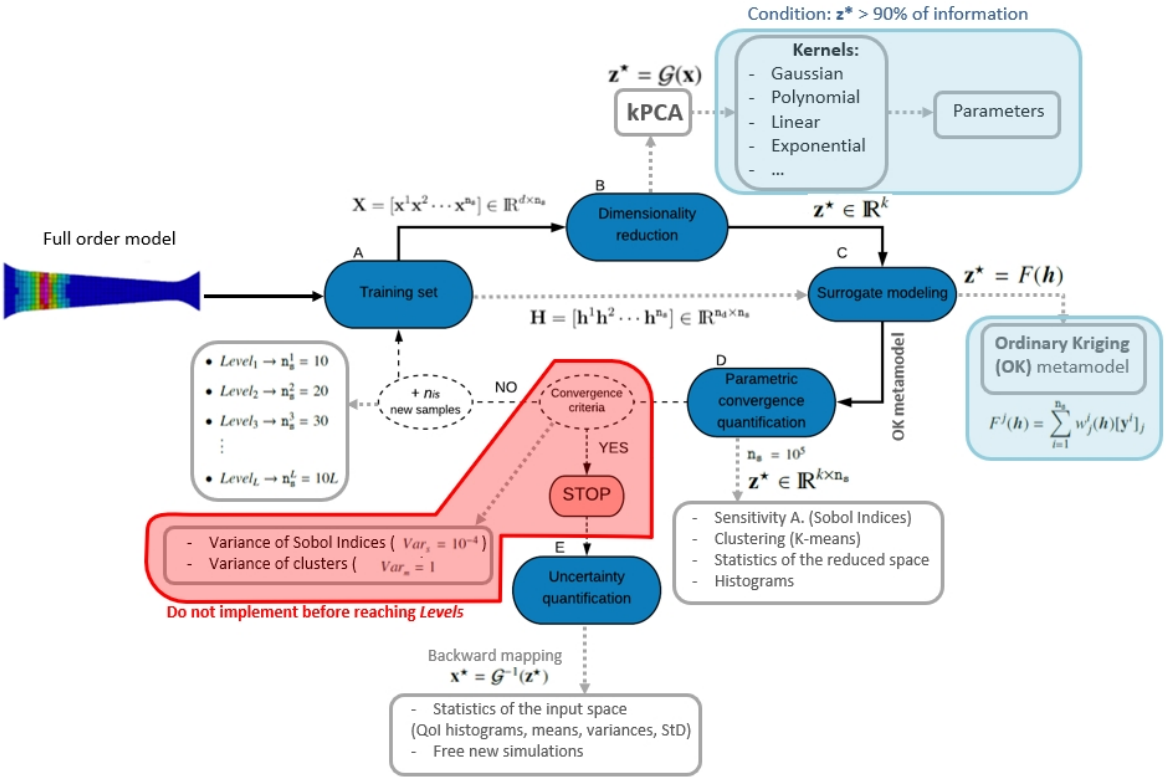

Condition: z* > 90% of information
Kernels:
- Gaussian
- Polynomial
- Linear
- Exponential
- ...
Parameters
$\mathbf{z}^\star = \mathcal{G}(\mathbf{x})$
kPCA
B
Dimensionality reduction
$\mathbf{X} = [\mathbf{x}^1\mathbf{x}^2\cdots\mathbf{x}^{\mathrm{n_s}}] \in \mathbb{R}^{d\times \mathrm{n_s}}$
$\mathbf{z}^\star \in \mathbb{R}^k$
Full order model
A
Training set
C
Surrogate modeling
$\mathbf{z}^\star = F(\boldsymbol{h})$
$\mathbf{H} = [\mathbf{h}^1\mathbf{h}^2\cdots\mathbf{h}^{\mathrm{n_s}}] \in \mathbb{R}^{\mathrm{n_d}\times \mathrm{n_s}}$
Ordinary Kriging (OK) metamodel
$F^j(\boldsymbol{h}) = \sum_{i=1}^{\mathrm{n_s}} w^i_j(\boldsymbol{h})[\mathbf{y}^i]_j$
OK metamodel
$Level_1 \rightarrow \mathrm{n}_s^1 = 10$
$Level_2 \rightarrow \mathrm{n}_s^2 = 20$
$Level_3 \rightarrow \mathrm{n}_s^3 = 30$
$\vdots$
$Level_L \rightarrow \mathrm{n}_s^L = 10L$
+ n_{is} new samples
NO
Convergence criteria
D
Parametric convergence quantification
YES
STOP
$\mathrm{n_s} = 10^5$
$\mathbf{z}^\star \in \mathbb{R}^{k\times \mathrm{n_s}}$
- Sensitivity A. (Sobol Indices)
- Clustering (K-means)
- Statistics of the reduced space
- Histograms
- Variance of Sobol Indices ($Var_s = 10^{-4}$)
- Variance of clusters ($Var_m = 1$
Do not implement before reaching Levels
E
Uncertainty quantification
Backward mapping
$\mathbf{x}^\star = \mathcal{G}^{-1}(\mathbf{z}^\star)$
- Statistics of the input space
(QoI histograms, means, variances, StD)
- Free new simulations

FIGURE 4.11 Flowchart overview of the proposed adaptive methodology.

4.4.1 Vademecum results

The idea of this section is to calculate the reference results, aiming to be compared with the proposed adaptive methodology. For this, the proposed methodology is implemented with a fixed number of samples $\mathtt{n_s}$ for the training set. It is considered six stochastic inputs $\boldsymbol{h} = [h_1, h_2, h_3, h_4, h_5, h_6]^T$ for the benchmark problem, where each input follows a normal distribution with its corresponding mean and standard deviation described in Table 4.1. For the QoI output it is considered the maximum plastic strain of all the shell elements ($d = 329$) from the tapered geometry. For the reference data set, a vademecun of $\mathtt{n_s} = 3000$ Monte Carlo samples are computed to obtain the output matrix, $\mathbf{X} = [\mathbf{x}^1\mathbf{x}^2\cdots\mathbf{x}^{\mathtt{n_s}}] \in \mathbb{R}^{d\times\mathtt{n_s}}$. Each vector $\mathbf{x}^i, i = 1, 2, ..., \mathtt{n_s}$ store the maximum plastic strain of all the elements for each simulation in the last step of time. In Fig. 4.12 it is shown the 3000 stochastic hardness curves for each simulation.

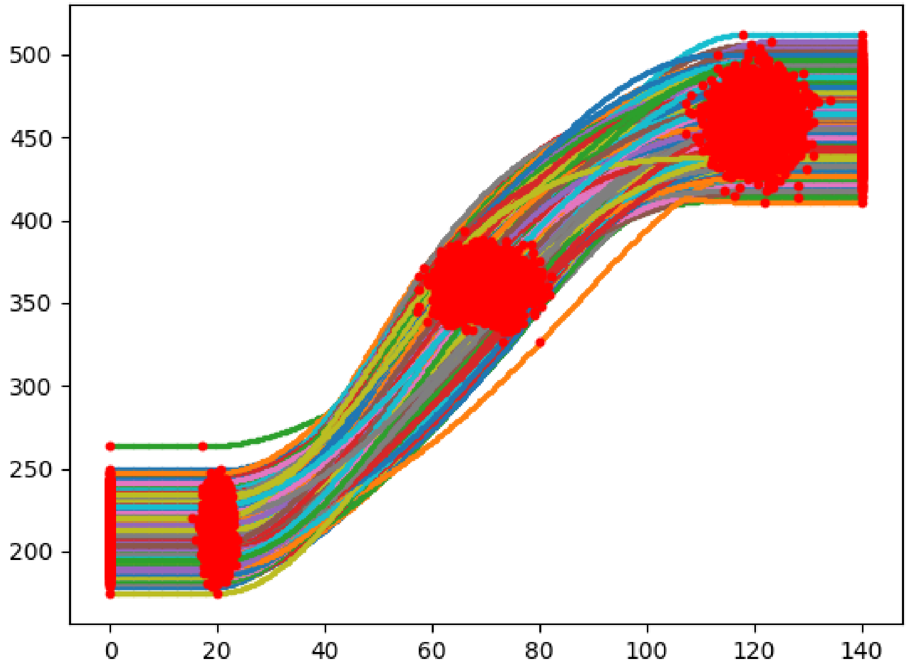


FIGURE 4.12 Input hardness curves of 3000 samples.

The dimensionality reduction problem is implemented with a Polynomial Kernel, $k(\mathbf{x}^i, \mathbf{x}^j) = (< x^i, x^j > +b)^p$ with a coefficient $b = 0.1$ and a polynomial degree $p = 3$. With this kernel configuration it is reduced the dimension of the problem to the first principal component ($k = 1$) with capturing $98,8\%$ of the variance information. Leading to obtain a reduced space $Y = \mathbf{z}^\star = [\mathbf{z}^\star]_1 \in \mathbb{R}^{k\times \mathrm{n_s}}$. Using linear PCA, 17 principal components ($k = 17$) are required to capture the same percentage information, and in consequence 17 metamodels for each dimension. Clearly an improvement is shown with respect to PCA (kPCA →k=1 and PCA →k=17). For the metamodel it is implemented OK between the input parameters $\boldsymbol{h}$ and the feature space $[\mathbf{z}^\star]_1$ with a spherical variogram (Oliver and Webster, 2014). For the UQ analysis (statistic measures and Sobol' indices) it is evaluated the metamodel with 10^5 random samples.

In Fig. 4.13 it is shown the reduced space of the first principal component $Y = [\mathbf{z}^\star]_1$ of kPCA and the corresponding PDF. Clearly, two clusters (modes of the structure) of samples are differentiated. The red and blue samples are plotted by K-means algorithm (Kodinariya and Makwana, 2013). The percentage of probability for each clusters are:

- Red cluster: 84.43%.

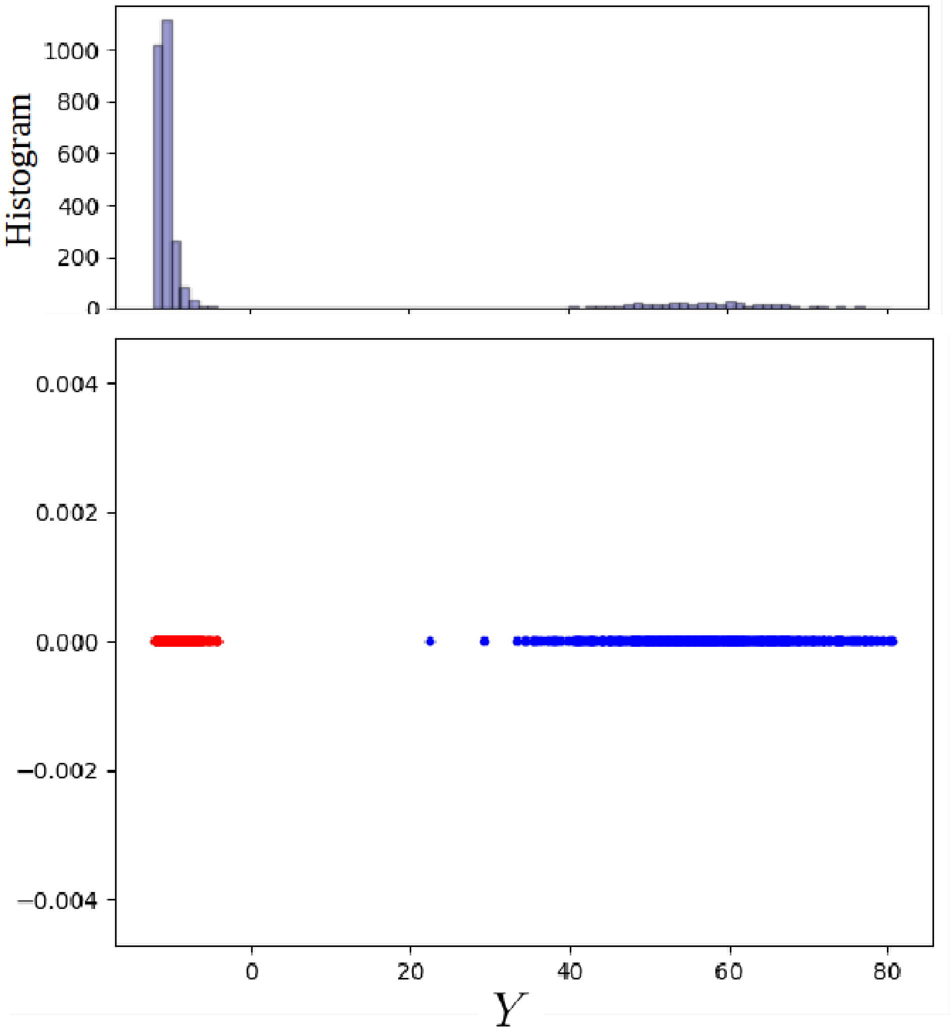


FIGURE 4.13 Histogram and reduced space Y.

- Blue cluster: 15.57%.

In Fig. 4.14 a solution of the original model for each cluster (red and blue) is shown. It is observed two different behaviours. The structure breaks either on the right (Fig. 4.14a) or to the left (Fig. 4.14b).

For the Sobol' sensitivity analysis it is used Saltelli method to obtain the conditional variance for each index (Saltelli et al., 2010). In Fig.4.15 it is plotted the values of the first order of Sobol' indices for each input. In Fig. 4.16 are shown the second order Sobol' indices. This index explains the interaction effect between all the possible pairs of parameters with respect to the output variance output. In Fig. 4.17 it is illustrated the 6 Total Sobol' indices. This index explains the total effect of an input parameter h_i to the total variance Var_Y.This measures the effect of the output variance of h_i, with respect to any variable and any order of interaction between parameters.

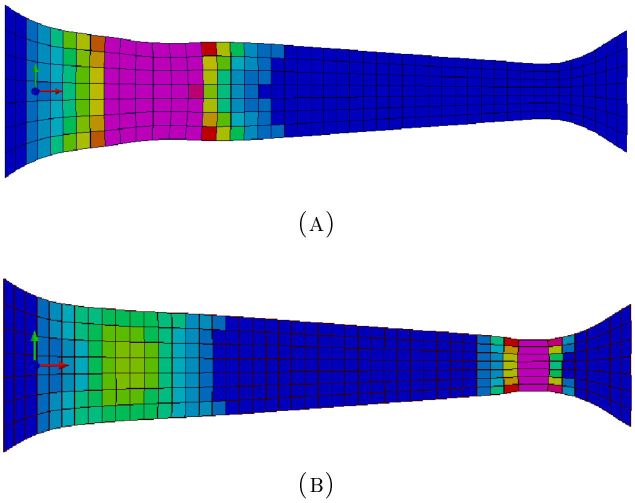


FIGURE 4.14 (a) Corresponds to a sample from the red mode of the reduced space Y transformed by backward mapping to the original space of the model. Figure (b) corresponds to a sample from the red mode of the reduced space Y transformed by backward mapping to the original space of the model.

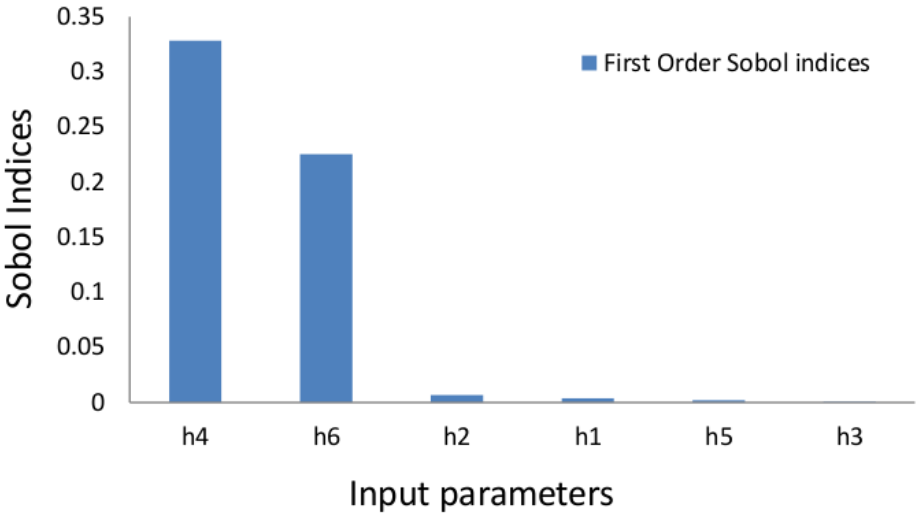


FIGURE 4.15 First order Sobol' indices.

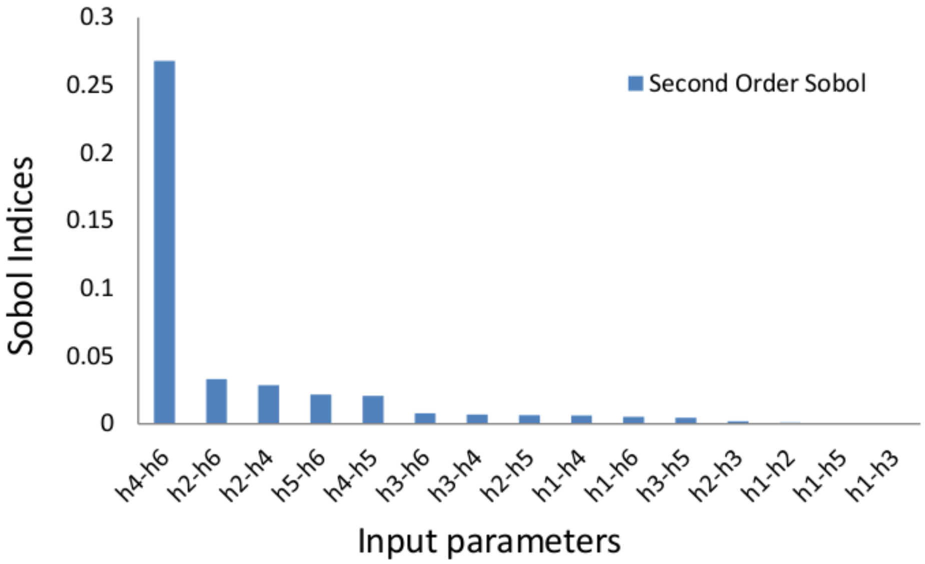


FIGURE 4.16 Second order Sobol' indices.

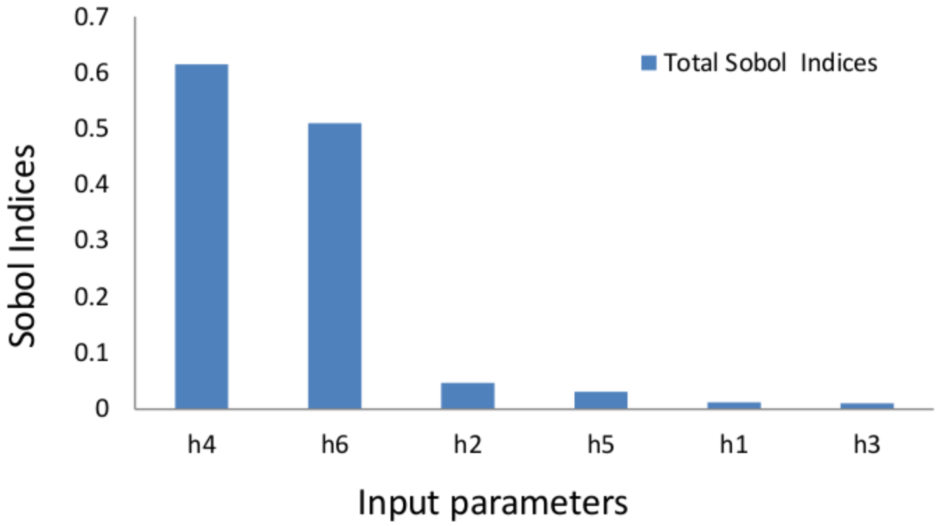


FIGURE 4.17 Total Sobol' indices.

From the above presented results, in Table 4.2 it is illustrated the most important variables to take into account as a reference values for the next section 4.4.2 where the adaptive methodology is tested.

TABLE 4.2 Reference values from a vademecum of 3000 training samples.

Clustering modes
Left mode = 84.43% Right mode = 15.57%
Sobol' indices
First order
$S_4 = 0.35$ $S_6 = 0.23$
Second order
$S_{46} = 0.26$
Total indices
$S_{T4} = 0.62$ $S_{T6} = 0.51$

4.4.2 Adaptive UQ methodology results

In this section it is presented the numerical results to validate the performance of the adaptive methodology applied to the benchmark problem described in Section 4.2. The approach is implemented with a polynomial kernel (degree $p = 3$) for the dimensionality reduction problem and OK for the surrogate modeling from $\boldsymbol{h}$ to Y. For the benchmark, Y coincides with the first component of the reduced space using kPCA. The criterion to stop the adaptive methodology is based in a variance criterion. The approach stops once the variance of the previous 5 levels ($s = 5$) achieves the order of $Var_S = 10^{-4}$ (for the Sobol' indices) and $Var_m = 1$ (for the mode percentage).

In Fig. 4.18 it is plotted the evolution of the Sobol' indices and the cluster percentage for each level of sampling size. We can analyze that the last 5 sample points of each graph have small variability, corroborating that the stopping criteria is accomplished. The method stops in $Level_{24}$ ($L = 24$) with a training set of $\mathtt{n}_\mathtt{s}^{24} = 240$.

In Table 4.3 it is compared the results obtained with the training set of 3000 samples (*Vade.*) with respect to the adaptive approach with 240 training samples (*Adapt.*).

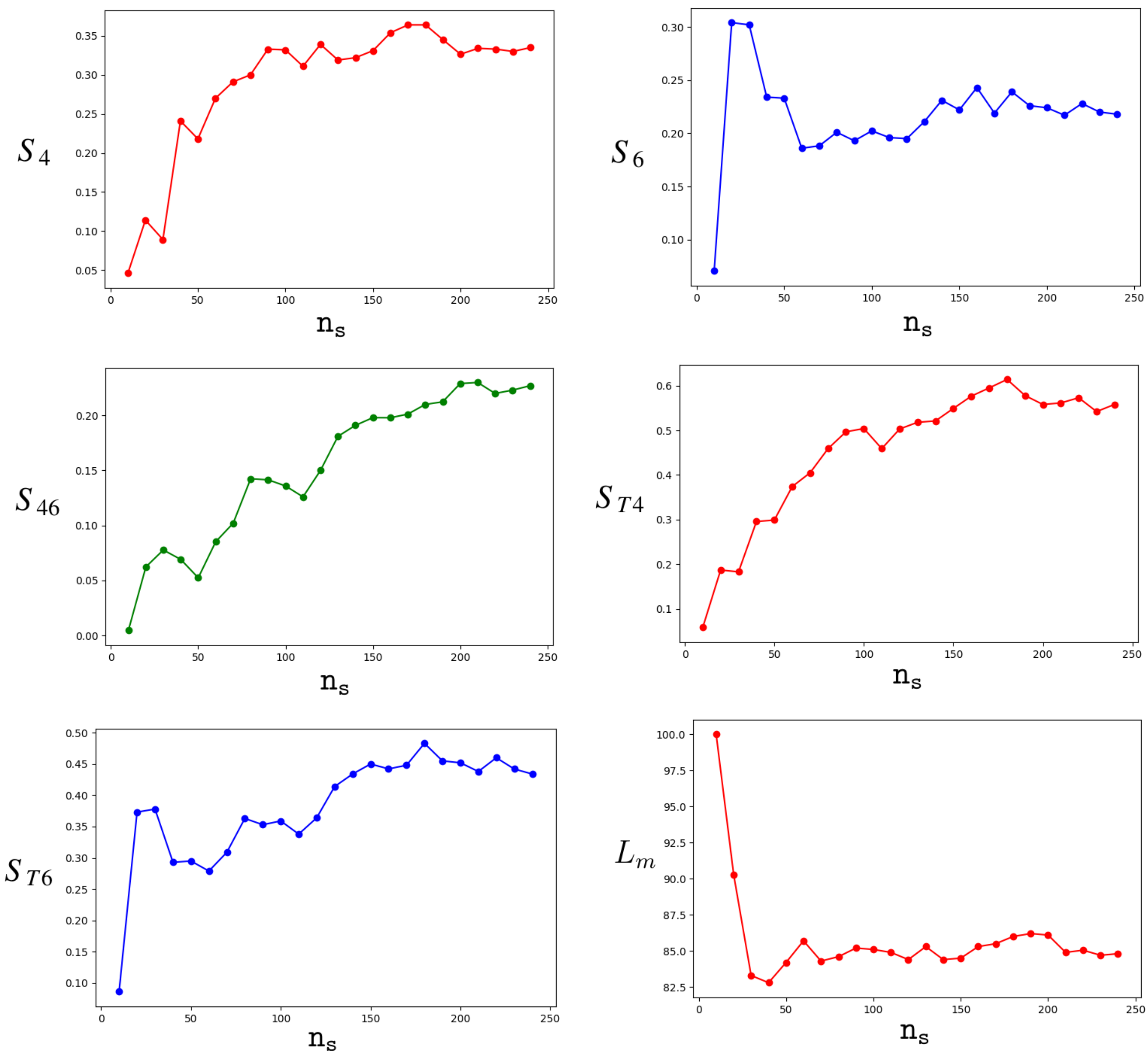


FIGURE 4.18 Evolution plots of the First Sobol' indices (S_4, S_6), Second order Sobol' Indice (S_{46}), Total Sobol' indices (S_{T4}, S_{T6}) and the percentage of the left mode (L_m).

Table 4.3 Comparison results between reference vademecum (3000 training samples) with respect to the adaptive methodology (240 training samples).

	L_m	R_m	S_4	S_6	S_{46}	S_{T4}	S_{T6}
Vade.	84.43%	15.57%	0.35	0.23	0.26	0.62	0.51
Adapt.	84.81%	15.19%	0.34	0.22	0.23	0.56	0.43

The surrogate model constructed with 240 training samples brings a powerful tool. Statistical measures (mean, variance and standard deviation) and scattered plots offer and interesting analysis to understand an analyze the cause of each structure mode. Fig. 4.19 shows the scatter plot between the two main parameters h_4 and h_6. The sample points are coloured in red or in blue, depending on the structure mode (left=red, right=blue).

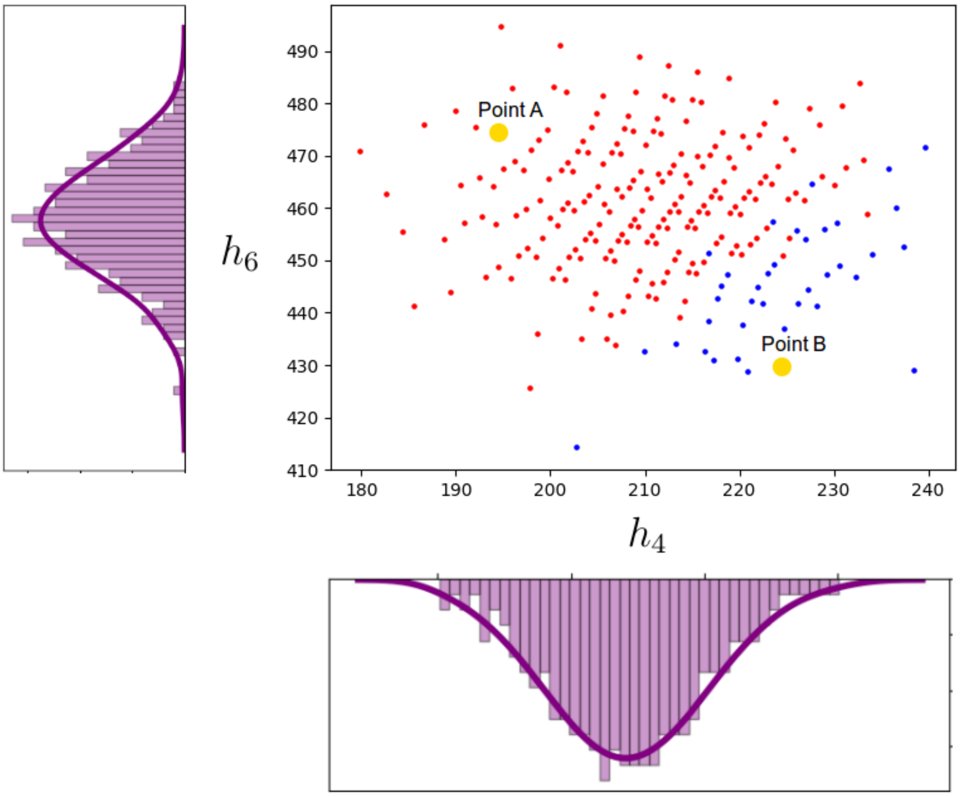
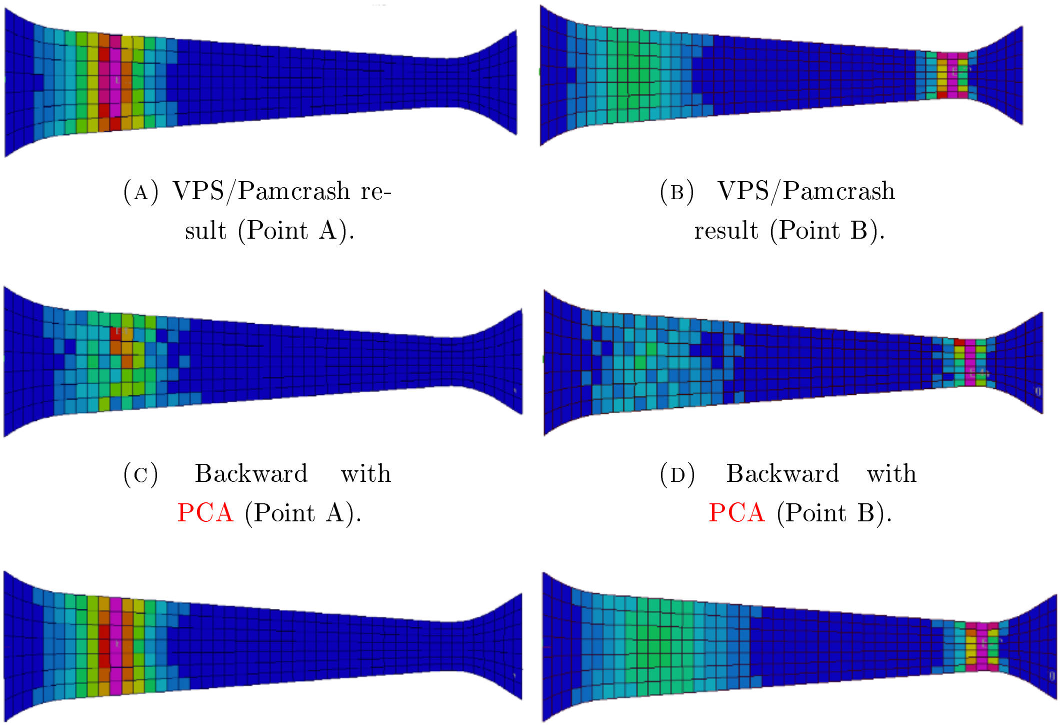


Figure 4.19 Scatter plot between the inputs h_4 and h_6. Red samples corresponds to the left mode and blue samples to the right mode. Point A ($h_1 = 22$, $h_2 = 60$, $h_3 = 128$, $h_4 = 195$, $h_5 = 333$, $h_6 = 472$). Point B ($h_1 = 18.5$, $h_2 = 65$, $h_3 = 122$, $h_4 = 224.5$, $h_5 = 365$, $h_6 = 430$).

Here it is clearly visible two differentiated areas of color points. This means that the relation between the parameters h_4 and h_6 defines practically the whole behaviour of each mode.

Each training sample needs considerable CPU resources and time. This method allows to obtain new simulations with a negligible consumption of time. In the scatter plot (Fig.4.19) a new point $\boldsymbol{h}$ for each color area is selected in a empty zone of the plot (yellow points) to backward to the original space $\mathbf{x}$. In Fig. 4.20 it is compared the full order computational samples with respect to the corresponding backward samples from Y to $\mathbf{x}$ with PCA and kPCA techniques. The performance of PCA and kPCA is shown to illustrate the improvement of kPCA for this crashworhtiness model.

kPCA clearly shows better performance in the backward mapping. Taking into account that with kPCA the first principal component $[\mathbf{z}^{\star}]_1$ contains 98.8% of information and PCA only 77.7%.



(A) VPS/Pamcrash result (Point A).

(B) VPS/Pamcrash result (Point B).

(C) Backward with PCA (Point A).

(D) Backward with PCA (Point B).

(E) Backward with kPCA (Point A).

(F) Backward with kPCA (Point B).

FIGURE 4.20 (a) and (b) illustrates the full order simulation with VPS/Pamcrash for the points A and B. Also (c), (d), (e) and (f) show the backwards from Y to the original space $\mathbf{x}$ with PCA and kPCA.

On the other hand, in Fig. 4.21 it is illustrated the histogram of the specific QoI function $l^0(\mathbf{x})$. Depending on the problem, the QoI is sufficient i some cases, for decision making (Rocas et al., 2021), since the QoI summarizes the information contained in $\mathbf{x}$. Here, the QoI corresponds to the average of vector $\mathbf{x}$. The statistical measures of the QoI are: mean=0.0407, variance=$1.12e-06$ and standard deviation=0.0011. The histogram presents a normal distribution centred approximately at 0.041 and a long tail on the left. The samples falling in the normal distribution corresponds to the left structure mode. Otherwise, the samples that fall into the distribution tail makes reference to the samples with the structure mode on the right.

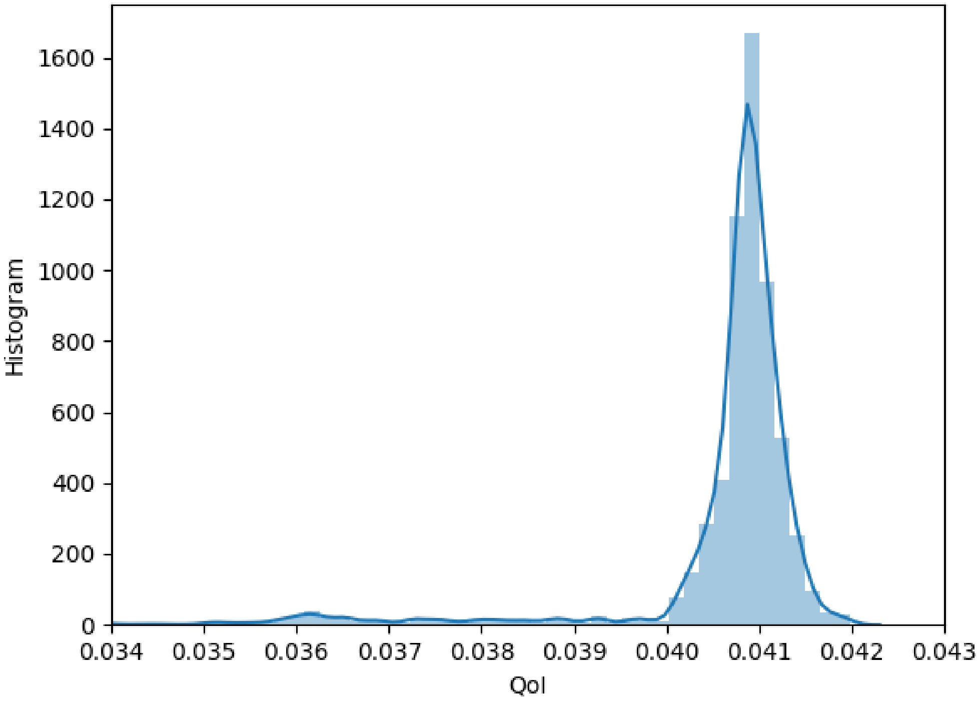


FIGURE 4.21 Histogram of the QoI with $\mathtt{n_s} = 10^5$.

4.5 Conclusions

Uncertainty quantification in crash simulation is a highly demanding research field for the automotive industry. On the one hand, its non linear behaviours combined with hidden structure modes leads to a challenging task for UQ analysis. On the other hand, high dimensional outputs for the quantity of interest can be a challenging problem for surrogate modelling by suffering the curse of dimensionality. Additionally, each evaluation of the high order model needs hours. Therefore, classic approaches as Monte Carlo are not viable.

This chapter presents an adaptive methodology for crashworthiness combining dimensionality reduction and surrogate modelling for an UQ and sensitivity analysis approach . The adaptive method evaluates limited set of samples of the high order model guaranteeing a good accuracy. The problem of dimensionality reduction for the outputs is tackle using kPCA in such a way OK for meta-modeling the reduced space of kPCA and the input samples $\mathbf{h}^i, i = 1, 2, \cdots, \mathtt{n_s}$. Moreover, cluster detection, percentages of success or failure, sensitivity analysis, statistics and free new simulations are a set of robust and reliability information for decision making. Having access to this information of the model is a rich tool for CAE departments for multi-purpose analysis.

The proposed methodology is implemented in a realistic industrial benchmark problem. The uncertainty of the problem is characterized with 6 random parameters defining the hardness curve of the material model. The maximum plastic strain for all the elements in the last time step is considered as QoI of the model. The methodology is implemented with a polynomial kernel. The convergence is achieved with 240 samples for the training set with a stopping criteria of a variance condition of $Var_S = 10^{-4}$ (for the Sobol' indices) and $Var_m = 1$ (for the mode percentage). The method detected two structure modes (clusters). The big mode approximately with 84% and a small mode with 16% of probability. In the biggest structure mode, the pyshical model concentrates high values of plastic strain in the left part. In contrast, in the small mode they are on the right area. Moreover, the main influence parameters for the output are h_4, h_6 ($h_4 \rightarrow 34\%$ and $h_6 \rightarrow 22\%$) for the first order, and $h_4 - h_6$ ($h_4 - h_6 \rightarrow 23\%$) for the second order of sensitivity. This means that the relation between these two parameters is defining practically all the behaviour of the model. In addition, the total Sobol' indices S_{T4} and S_{T6} are showing similar relation with respect to the first and second order of Sobol' indices. This emphasizes that the other parameters h_1, h_2, h_3, h_5 have low influence to the output. Also, the specific QoI function $l^0(\cdot)$ shows a normal distribution with a long tail in the left allowing to facilitate the understanding of the high dimension vector $\mathbf{x}$.

The results from the industrial benchmark verified the performance and accuracy of the proposed methodology with respect to a vademecum approach of 3000 samples for the training set. The method can be extended and applied for other disciplines (e.g. aerodynamics, occupant safety, aeroacoustic, among others) with uncertainty inputs, nonlinear responses and high dimensional outputs. The methodology is presented with kPCA and OK for dimensionality reduction and surrogate modeling, respectively. Nevertheless, other dimensionality reduction techniques as Isometric Mapping or Locally Embeddings can be implemented as well. In addition, other metamodel techniques (regression, interpolation) can be used as well depending on the data.

The combination of dimensionality reduction and surrogate models with an adaptive approach for multi-purpose information produces accurate solutions with an affordable computational cost, accounting also for the uncertainty, that is assessing the credibility of the simulation. Particularly in the context of crashworthiness UQ, the computational cost is a key issue and a driving force for the research developments in the field. Since, increasing accuracy requires a higher computational effort, finding a trade-off between these two factors is a critical concern for last decision making. This chapter intends to provide tools to achieve accurate and credible crashworthiness industrial simulations at an acceptable computational effort.

Chapter 5

Conclusions

The present Doctoral thesis aims to bridge the gap between uncertainty quantification, machine learning and crashworthiness simulations in order to enable robust models. Robustness is essential to qualify numerical simulations as credible alternatives to experimental test upon prototypes. Freeing the design loop from the dependence on experiments is suppressing a bottleneck both in terms of financial cost and timely response.

The field of crashworthiness presents three main characteristics: (i) uncertain inputs, coming from different sources of variability, e.g. manufactured processes, supplier tolerance, human errors, simplifications and assembly processes, (ii) nonlinear behaviours resulting from the complex nature of a complex system which simulation demands high computational costs, and, (iii) high dimensional responses that jeopardize the efficiency of any UQ approach for post-processing. In this thesis, all these concepts have been accounted for and analyzed in order to propose different solutions.

As a first step, a benchmark crash problem from the SEAT portfolio is presented as a starting point for the thesis in Chapter 2. This chapter provides a state-of-the-art review of uncertainty quantification techniques for crashworthiness. The content is divided in intrusive and non intrusive approaches. Quasi Monte Carlo and Point Collocation Polynomial Chaos are implemented as non intrusive approaches and compared with the classic Monte Carlo (as a reference approach). Both methods show interesting results for capturing the general behaviour of the benchmark problem. However, different limitations were detected. Nonlinear behaviours present difficulties for Polynomial Chaos to capture the principal modes. Also, high dimensional outputs jeopardize the postprocess and the data analysis. In consequence, sensitivity analysis and the detection of new hidden modes becomes a cumbersome task.

In Chapter 3 it is presented a new methodology to tackle the disadvantages detected in Chapter 2 by combining dimensionality reduction and surrogate modeling. The novelty of this methodology/algorithm lies in how the dimensionality reduction and surrogate modeling are coupled together for crashworthiness UQ analysis. The coupling is performed between the reduced space obtained with a dimensionality reduction technique and the stochastic input space of parameters. For this purpose, kPCA is used for dimensionality reduction and for the surrogate modeling, Ordinary Kriging and Polynomial Regression are implemented to show their performance with respect to a new proposed technique developed in this thesis, called, Separated Response Surface and based in the PGD method. The methodology is tested for the benchmark B-Pillar problem, where the best performance is achieved with the combination of kPCA and OK or SRS. Depending on how is the dataset of the reduced space, it is properly to use OK for non noisy samples and SRS for noisy samples. The proposed methodology has shown good results for uncertainty quantification, allowing to describe the propagation of the input randomness to the model outputs with a smart approach.The advantage of this approach lies in the reduced space of kPCA where the output samples are containing the most important information of the training set. In this way, the reduced space can be analyzed in a more intelligent and efficient way.

The proposed algorithm is useful for robust analysis when the model can present nonlinear behaviours that leads to hidden structure modes. The method allows to quantify statistics (e.g. mean, variance, standard deviation, probability density function), structure modes (e.g. detection of bifurcation modes, percentage of modes), robustness criterion, sensitivity analysis (e.g. scattering plots of influence parameters) and mapping new simulations by a backward mapping to the physical space in almost real time. The approach is extremely useful for a rich dataset. However, obtaining a large dataset of simulations is sometimes unaffordable in crashworthiness. Therefore, it is of critical importance an adaptive criterion to decide how many number of simulations are needed for the training set to guaranty enough information for the analysis and avoid oversampling.

Chapter 4 presents a realistic industrial problem for Volkswagen and SEAT portfolio. The objective is to quantify the uncertainty of a model that presents a randomness hardness curve. It is presented an adaptive approach improving the weaknesses of the previous algorithm (dimensionality reduction and surrogate modeling) presented in Chapter 3. A novel methodology with an adaptive approach is developed to evaluate only the necessary samples for the training set. Taking into account that the input space is sampled with Halton sequences, and the stopping criterion for the method is achieved by evaluating the convergence of the sensitivity index (Sobol' indices) and the cluster percentages (these stopping criterion variables are used to evaluate the convergence, since these are sensitive measures for the industrial specialist). This adaptive methodology is tested using a industrial benchmark problem of interest for SEAT and it is compared with a vademecum of 3000 samples for the training set. The method illustrates a good performance equilibrium between the number of evaluation for the high order model and the uncertainty results of the model.

Overall, this thesis provides a new methodology tool for a wide class of problems, but specifically for crashworthiness. This has strong practical implications for numerous relevant problems for SEAT and the industry in general (e.g. structural design, aeroacoustic, aerodynamics, occupant protection, data analysis,). In the industrial framework of SEAT, the proposed methodology has allowed to deal with uncertainty problems with an efficient approach for the EK department. This thesis has been the incentive and the starting point of a new line of research and development for SEAT, where the combination of uncertainty quantification, machine learning and data science is growing exponentially. The thesis provide a useful knowledge and tool for engineering decision making. Allowing to evaluate the robustness of any project to decide new redesigns, decisions or paths.

However, the methodology developed in this doctoral thesis presents some limitations. In the dimensionality reduction step, explained in section 4.3.2 and specifically detailed in section 3.3.2 (containing the theoretical details of kPCA), an important concern is the choice of the kernel and the setting of the hyperparameters, as they define the reduced space for a future metamodeling step. The criterion to select the these variables is determined with the percentage of information of the principal components. This is a good criterion to store the maximum content of each dimension and to mapped backward to the physical space $\mathbf{x}$. However, it is also of interest to evaluate the ability of the dimensionality reduction technique for clustering detection. In this way, we could improve the metamodeling step between the inputs and the reduced space. This merge between the percentage of information and the clustering capacity would improve the methodology allowing to optimize the reduced space taking into account these two criteria.

Moreover, the methodology presented here assumes that the stochastic behaviour of the uncertain input parameters is properly characterized. This stochastic behaviour is propagated to the output. The determination of this aleatory description deserves however more intensive research, and the final results are extremely sensitive to it. Also output data resulting from the simulations has to be properly managed.

Referring to the nature of this Industrial Doctorate, the achievements of this doctoral thesis are presented below from two points of view: i) one focused on a scientific direction, and ii) on a more industrial branch.

The main achievements and advantages of this UQ methodology for a scientific approach are:

- Discover the nonlinear structure hidden modes.
- Percentage of success or failure for each mode to evaluate the robustness of the modes.
- Know the cause of any structure mode by the combination of the input parameters.
- Ranking of input influence (independent influence, second order influence, total influence).
- Statistical measures of a crash model(mean, variance, standard deviation and probability density functions).
- Evaluation of new simulations with negligible computational cost with any combination of the input parameters.
- Reduced computational cost with respect to the classic UQ methods.

The main benefits and advantages that this thesis brings to the EK department from an industrial point of view:

- Introduce a new development and research line into SEAT for uncertainty quantification, machine learning and data science for EK calculus department. Enabling to create a team for stochastic/data science.
- Engineers' expertise is used to monitor and make decisions for new redesigns and ideas. It is reduced the work time of changing models, launching simulations and many hours in post processing the responses.
- A compact machine learning/methodology tool to evaluate the robustness of a crash model.

Bibliography

Al-Momani, Emad and Ibrahim Rawabdeh (2008). “An application of finite element method and design of experiments in the optimization of sheet metal blanking process”. In: *JJMIE* 2.1, pp. 53–63.

Arregui-Mena, Jose David, Lee Margetts, and Paul M Mummery (2016). “Practical application of the stochastic finite element method”. In: *Archives of computational methods in engineering* 23.1, pp. 171–190.

Askey, Richard and James Arthur Wilson (1985). “Some basic hypergeometric orthogonal polynomials that generalize Jacobi polynomials”. In: 319.

Aslett, Louis JM, Tigran Nagapetyan, and Sebastian J Vollmer (2017). “Multilevel Monte Carlo for reliability theory”. In: *Reliability Engineering & System Safety* 165, pp. 188–196.

Atkinson, Anthony, Alexander Donev, and Randall Tobias (2007). “Optimum experimental designs, with SAS”. In: 34.

Barth, Andrea, Christoph Schwab, and Nathaniel Zollinger (2011). “Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients”. In: *Numerische Mathematik* 119.1, pp. 123–161.

Bergman, LA et al. (1997). “A state-of-the-art report on computational stochastic mechanics”. In: *Probabilistic Engineering Mechanics* 12.4, pp. 197–321.

Berveiller, M, B Sudret, and M Lemaire (2004). “Presentation of two methods for computing the response coefficients in stochastic finite element analysis”. In: *Proc. 9th ASCE Specialty Conference on Probabilistic Mechanics and Structural Reliability, Albuquerque, USA.*

Böttcher, Curd-Sigmund, Steffen Frik, and Bernd Gosolits (2005). *20 years of crash simulation at Opel-experiences for future challenges.*

Boyd, John P and Fei Xu (2009). “Divergence (Runge phenomenon) for least-squares polynomial approximation on an equispaced grid and Mock–Chebyshev subset interpolation”. In: *Applied Mathematics and Computation* 210.1, pp. 158–168.

Datta, Saurav, Asish Bandyopadhyay, and Pradip Kumar Pal (2008). “Grey-based Taguchi method for optimization of bead geometry in submerged arc bead-on-plate welding”. In: *The International Journal of Advanced Manufacturing Technology* 39.11-12, pp. 1136–1143.

Díez, Pedro et al. (2018). “Algebraic PGD for tensor separation and compression: an algorithmic approach”. In: *C. R. Mec.* 346.7, pp. 501–514. DOI: 10.1016/j.crme.2018.04.01.

Díez, Pedro et al. (2019). “Encapsulated PGD algebraic toolbox operating with high-dimensional data”. In: *Archives of Computational Methods in Engineering*, pp. 1–16.

Doostan, Alireza, Roger G Ghanem, and John Red-Horse (2007). “Stochastic model reduction for chaos representations”. In: *Computer Methods in Applied Mechanics and Engineering* 196.37-40, pp. 3951–3966.

Du Bois, Paul et al. (2004). “Vehicle crashworthiness and occupant protection”. In:

Eiermann, Michael, Oliver G Ernst, and Elisabeth Ullmann (2007). “Computational aspects of the stochastic finite element method”. In: *Computing and visualization in science* 10.1, pp. 3–15.

Eldred, Michael (2009). “Recent advances in non-intrusive polynomial chaos and stochastic collocation methods for uncertainty analysis and design”. In: *50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference 17th AIAA/ASME/AHS Adaptive Structures Conference 11th AIAA No*, p. 2274.

Eldred, Michael, Clayton Webster, and Paul Constantine (2008). “Evaluation of non-intrusive approaches for Wiener-Askey generalized polynomial chaos”. In: *49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 16th AIAA/ASME/AHS Adaptive Structures Conference, 10th AIAA Non-Deterministic Approaches Conference, 9th AIAA Gossamer Spacecraft Forum, 4th AIAA Multidisciplinary Design Optimization Specialists Conference*, p. 1892.

Fang, Hongbing et al. (2005). “A comparative study of metamodeling methods for multiobjective crashworthiness optimization”. In: *Computers & structures* 83.25-26, pp. 2121–2136.

Fei, Ng Chin, Nik Mizamzul Mehat, and Shahrul Kamaruddin (2013). “Practical applications of Taguchi method for optimization of processing parameters for plastic injection moulding: a retrospective review”. In: *ISRN Industrial engineering* 2013.

Feinberg, Jonathan and Hans Petter Langtangen (2015). “Chaospy: An open source tool for designing methods of uncertainty quantification”. In: *Journal of Computational Science* 11, pp. 46–57.

Fratila, Domnita and Cristian Caizar (2011). “Application of Taguchi method to selection of optimal lubrication and cutting conditions in face milling of AlMg3”. In: *Journal of Cleaner Production* 19.6-7, pp. 640–645.

Galas, David J et al. (2017). “Expansion of the Kullback-Leibler divergence, and a new class of information metrics”. In: *Axioms* 6.2, p. 8.

Gano, Shawn, Harold Kim, and Don Brown (2006). “Comparison of three surrogate modeling techniques: Datascape, kriging, and second order regression”. In: *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, p. 7048.

García-González, Alberto et al. (2020). “A kernel Principal Component Analysis (kPCA) digest with a new backward mapping (pre-image reconstruction) strategy”. In: *arXiv preprint arXiv:2001.01958*.

Garikapati, Hasini et al. (2020). “A Proper Generalized Decomposition (PGD) approach to crack propagation in brittle materials: with application to random field material properties”. In: *Computational Mechanics* 65.2, pp. 451–473.

Ghanem, Roger G and Robert M Kruger (1996). “Numerical solution of spectral stochastic finite element systems”. In: *Computer methods in applied mechanics and engineering* 129.3, pp. 289–303.

Ghanem, Roger G and Pol D Spanos (2003). *Stochastic finite elements: a spectral approach*. Courier Corporation.

Giles, Michael B (2008). “Multilevel monte carlo path simulation”. In: *Operations Research* 56.3, pp. 607–617.

Gilli, L et al. (2013). “Uncertainty quantification for criticality problems using non-intrusive and adaptive polynomial chaos techniques”. In: *Annals of Nuclear Energy* 56, pp. 71–80.

Giunta, Anthony and Layne Watson (1998). “A comparison of approximation modeling techniques-Polynomial versus interpolating models”. In: *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, p. 4758.

Gopalsamy, Bala Murugan, Biswanath Mondal, and Sukamal Ghosh (2009). “Taguchi method and ANOVA: An approach for process parameters optimization ofhard machining while machining hardened steel”. In:

Graham, Ivan G, Matthew J Parkinson, and Robert Scheichl (2018). “Modern Monte Carlo variants for uncertainty quantification in neutron transport”. In: *Contemporary Computational Mathematics-A Celebration of the 80th Birthday of Ian Sloan*. Springer, pp. 455–481.

Grigoriu, Mircea (2006). "Evaluation of Karhunen–Loève, spectral, and sampling representations for stochastic processes". In: *Journal of engineering mechanics* 132.2, pp. 179–189.

Hadigol, Mohammad and Alireza Doostan (2018). "Least squares polynomial chaos expansion: A review of sampling strategies". In: *Computer Methods in Applied Mechanics and Engineering* 332, pp. 382–407.

Hammersley, John M (1960). "Monte Carlo methods for solving multivariable problems". In: *Annals of the New York Academy of Sciences* 86.3, pp. 844–874.

Haug, E, T Scharnhorst, and P Du Bois (1986). "FEM-Crash, Berechnung eines Fahrzeugfrontalaufpralls". In: *VDI Berichte* 613, pp. 479–505.

Hauke, Jan and Tomasz Kossowski (2011). "Comparison of values of Pearson's and Spearman's correlation coefficients on the same sets of data". In: *Quaestiones geographicae* 30.2, pp. 87–93.

Homma, Toshimitsu and Andrea Saltelli (1996). "Importance measures in global sensitivity analysis of nonlinear models". In: *Reliability Engineering & System Safety* 52.1, pp. 1–17.

Hosder, Serhat, Robert Walters, and Michael Balch (2007). "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*, p. 1939.

Hosder, Serhat, Robert Walters, and Rafael Perez (2014). "A Non-Intrusive Polynomial Chaos Method For Uncertainty Propagation in CFD Simulations". In: DOI: 10.2514/6.2006-891.

Jäckel, Peter (2005). "A note on multivariate Gauss-Hermite quadrature". In: *London: ABN-Amro. Re.*

Kaintura, Arun, Tom Dhaene, and Domenico Spina (2018). "Review of polynomial chaos-based methods for uncertainty quantification in modern integrated circuits". In: *Electronics* 7.3, p. 30.

Kleiber, Michał and Tran Duong Hien (1992). "The stochastic finite element method: basic perturbation technique and computer implementation". In:

Kodinariya, Trupti M and Prashant R Makwana (2013). "Review on determining number of Cluster in K-Means Clustering". In: *International Journal* 1.6, pp. 90–95.

Kucherenko, Sergei and Shufang Song (2017). "Different numerical estimators for main effect global sensitivity indices". In: *Reliability Engineering & System Safety* 165, pp. 222–238.

Lataniotis, Christos, Stefano Marelli, and Bruno Sudret (2018). “Extending classical surrogate modelling to ultrahigh dimensional problems through supervised dimensionality reduction: a data-driven approach”. In: *arXiv preprint arXiv:1812.06309*.

Le Maıtre, Olivier P et al. (2002). “A stochastic projection method for fluid flow: II. Random process”. In: *Journal of computational Physics* 181.1, pp. 9–44.

Li, Min, Ruo-Qian Wang, and Gaofeng Jia (2020). “Efficient dimension reduction and surrogate-based sensitivity analysis for expensive models with high-dimensional outputs”. In: *Reliability Engineering & System Safety* 195, p. 106725.

Likas, Aristidis, Nikos Vlassis, and Jakob J Verbeek (2003). “The global k-means clustering algorithm”. In: *Pattern recognition* 36.2, pp. 451–461.

Lin, G et al. (2005). “A computational design-of-experiments study of hemming processes for automotive aluminium alloys”. In: *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 219.10, pp. 711–722.

Liu, Wing Kam, Ted Belytschko, and A Mani (1986a). “Probabilistic finite elements for nonlinear structural dynamics”. In: *Computer Methods in Applied Mechanics and Engineering* 56.1, pp. 61–81.

— (1986b). “Random field finite elements”. In: *International journal for numerical methods in engineering* 23.10, pp. 1831–1845.

Liu, Zhangjun, Zixin Liu, and Yongbo Peng (2017). “Dimension reduction of Karhunen-Loeve expansion for simulation of stochastic processes”. In: *Journal of Sound and Vibration*. ISSN: 10958568. DOI: 10.1016/j.jsv.2017.07.016.

Lu, Y, N Blal, and A Gravouil (2018a). “Adaptive sparse grid based HOPGD: Toward a nonintrusive strategy for constructing space-time welding computational vademecum”. In: *International Journal for Numerical Methods in Engineering* 114.13, pp. 1438–1461.

Lu, Ye, Nawfal Blal, and Anthony Gravouil (2018b). “Multi-parametric space-time computational vademecum for parametric studies: Application to real time welding simulations”. In: *Finite Elements in Analysis and Design* 139, pp. 62–72.

Malvić, Tomislav and Davorin Balić (2009). “Linearity and Lagrange Linear Multiplicator in the Equations of Ordinary Kriging”. In: *Nafta: exploration, production, processing, petrochemistry* 60.1, pp. 31–43.

Mathelin, Lionel, M Yousuff Hussaini, and Thomas A Zang (2005). “Stochastic approaches to uncertainty quantification in CFD simulations”. In: *Numerical Algorithms* 38.1-3, pp. 209–236.

Matthies, Hermann G and Andreas Keese (2005). “Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations”. In: *Computer methods in applied mechanics and engineering* 194.12-16, pp. 1295–1331.

Moustapha, Maliki et al. (2014). “Metamodeling for crashworthiness design: comparative study of kriging and support vector regression”. In: *Uncertainties 2014-proceedings of the 2nd International Symposium on Uncertainty Quantification and Stochastic Modeling, July 7th to July 11th, 2014, Rouen, France.* ETH-Zürich.

Nagel, Joseph B, Jörg Rieckermann, and Bruno Sudret (2017). “Uncertainty quantification in urban drainage simulation: fast surrogates for sensitivity analysis and model calibration”. In: *arXiv preprint arXiv:1709.03283.*

Niederreiter, Harald (1978). “Quasi-Monte Carlo methods and pseudo-random numbers”. In: *Bulletin of the American Mathematical Society* 84.6, pp. 957–1041.

Nouy, Anthony (2009). “Recent developments in spectral stochastic methods for the numerical solution of stochastic partial differential equations”. In: *Archives of Computational Methods in Engineering* 16.3, pp. 251–285.

Oliver, MA and R Webster (2014). “A tutorial guide to geostatistics: Computing and modelling variograms and kriging”. In: *Catena* 113, pp. 56–69.

PAM-SCL - Theory Notes Manual (2000).

Phoon, KK, HW Huang, and ST Quek (2005). “Simulation of strongly non-Gaussian processes using Karhunen–Loeve expansion”. In: *Probabilistic engineering mechanics* 20.2, pp. 188–198.

Qiu, Na et al. (2018). “Crashworthiness optimization with uncertainty from surrogate model and numerical error”. In: *Thin-Walled Structures* 129, pp. 457–472.

Rifkin, Ryan M and Ross A Lippert (2007). “Notes on regularized least squares”. In:

Rocas, M et al. (2020). “Nonintrusive Stochastic Finite Elements for Crashworthiness with VPS/Pamcrash”. In: *Archives of Computational Methods in Engineering*, pp. 1–26.

Rocas, Marc et al. (2021). “Nonintrusive Uncertainty Quantification for automotive crash problems with VPS/Pamcrash”. In: *Finite Elements in Analysis and Design* 193, p. 103556.

Roy, Ranjit K (2001). *Design of experiments using the Taguchi approach: 16 steps to product and process improvement.* John Wiley & Sons.

Saltelli, Andrea et al. (2010). "Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index". In: *Computer physics communications* 181.2, pp. 259–270.

Saxena, Amit et al. (2017). "A review of clustering techniques and developments". In: *Neurocomputing* 267, pp. 664–681.

Schölkopf, Bernhard, Alexander Smola, and Klaus-Robert Müller (1998). "Nonlinear component analysis as a kernel eigenvalue problem". In: *Neural computation* 10.5, pp. 1299–1319.

Shinozuka, Masanobu and George Deodatis (1991). "Simulation of stochastic processes by spectral representation". In: *Applied Mechanics Reviews* 44.4, pp. 191–204.

Sobol, Ilya M (1993). "Sensitivity analysis for non-linear mathematical models". In: *Mathematical modelling and computational experiment* 1, pp. 407–414.

Spethmann, Philipp, Stefan H Thomke, and Cornelius Herstatt (2006). *The impact of crash simulation on productivity and problem-solving in automotive R&D.* Tech. rep. Working Paper.

Sraj, Ihab et al. (2017). "Quantifying uncertainties in fault slip distribution during the Tōhoku tsunami using polynomial chaos". In: *Ocean Dynamics* 67.12, pp. 1535–1551.

Stefanou, George (2009). "The stochastic finite element method: past, present and future". In: *Computer methods in applied mechanics and engineering* 198.9-12, pp. 1031–1051.

Stefanou, George and Manolis Papadrakakis (2007). "Assessment of spectral representation and Karhunen–Loève expansion methods for the simulation of Gaussian stochastic fields". In: *Computer methods in applied mechanics and engineering* 196.21-24, pp. 2465–2477.

Sudret, Bruno (2008). "Global sensitivity analysis using polynomial chaos expansions". In: *Reliability engineering & system safety* 93.7, pp. 964–979.

Sudret, Bruno and Armen Der Kiureghian (2000). *Stochastic finite element methods and reliability: a state-of-the-art report.* Department of Civil and Environmental Engineering, University of California ...

Sudret, Bruno and Chu V Mai (2015). "Computing derivative-based global sensitivity measures using polynomial chaos expansions". In: *Reliability Engineering & System Safety* 134, pp. 241–250.

Taguchi, Genichi and Seiso Konishi (1987). "Taguchi methods: orthogonal arrays and linear graphs; tools for quality engineering". In:

Tsui, Kwok-Leung (1992). "An overview of Taguchi method and newly developed statistical methods for robust design". In: *Iie Transactions* 24.5, pp. 44–57.

Van Der Maaten, Laurens, Eric Postma, and Jaap Van den Herik (2009). "Dimensionality reduction: a comparative". In: *J Mach Learn Res* 10.66-71, p. 13.

Wang, Quan (2012). "Kernel principal component analysis and its applications in face recognition and active shape models". In: *arXiv preprint arXiv:1207.3538*.

Wang, Tao et al. (2018). "Crashworthiness analysis and multi-objective optimization of a commercial vehicle frame: A mixed meta-modeling-based method". In: *Advances in mechanical engineering* 10.5, p. 1687814018778480.

Wasserstein, Ronald L (1997). "Monte carlo: Concepts, algorithms, and applications". In:

Wong, Tien-Tsin, Wai-Shing Luk, and Pheng-Ann Heng (1997). "Sampling with Hammersley and Halton points". In: *Journal of graphics tools* 2.2, pp. 9–24.

Xiu, Dongbin (2009). "Fast numerical methods for stochastic computations: a review". In: *Communications in computational physics* 5.2-4, pp. 242–272.

— (2010). *Numerical methods for stochastic computations: a spectral method approach*. Princeton university press.

Xiu, Dongbin and George Em Karniadakis (2002). "The Wiener–Askey polynomial chaos for stochastic differential equations". In: *SIAM journal on scientific computing* 24.2, pp. 619–644.

Zang, C, MI Friswell, and JE Mottershead (2005). "A review of robust optimal design and its application in dynamics". In: *Computers & structures* 83.4-5, pp. 315–326.

Zhang, Yi (2013). "Efficient uncertainty quantification in aerospace analysis and design". In:

Zheng, Wei-Shi, JianHuang Lai, and Pong C Yuen (2010). "Penalized preimage learning in kernel principal component analysis". In: *IEEE Transactions on Neural Networks* 21.4, pp. 551–570.