Chaos polynomial creux et adaptatif pour la propagation d'incertitudes et l'analyse de sensibilité

Géraud Blatman

N° d'ordre : 1955
EDSPIC : 446

Université BLAISE PASCAL - Clermont II

École Doctorale Sciences pour l'Ingénieur de Clermont-Ferrand

THÈSE

présentée par

Géraud BLATMAN

pour obtenir le grade de

Docteur d'Université

Spécialité : Génie Mécanique

Adaptive sparse polynomial chaos expansions for uncertainty propagation and sensitivity analysis

soutenue publiquement le 8 Octobre 2009 devant le jury composé de Messieurs :

Pr. Gilles FLEURY	Supélec	Rapporteur
Dr. Anthony NOUY	Université de Nantes	Rapporteur
Pr. Anestis ANTONIADIS	Université Joseph Fourier	Examinateur
Dr. Didier LUCOR	Université Paris VI	Examinateur
Pr. Maurice LEMAIRE	Institut Français de Mécanique Avancée	Correspondant universitaire
Dr. Marc BERVEILLER	EDF R&D	Responsable industriel
Dr. Bruno SUDRET	Phimeca Engineering, associé LaMI	Directeur de thèse

Laboratoire de Mécanique et Ingénieries,
Institut Français de Mécanique Avancée et Université Blaise Pascal

"La connaissance progresse en intégrant en elle l'incertitude, non en l'exorcisant."
Edgar Morin, *La Méthode*

À mes parents, Anne et Michel.
À mon frère, Romain.
En souvenir de mes grands-parents,
Paulette, Madeleine, Marcel et Jean.

Acknowledgements

First and foremost I offer my most sincer gratitude to my thesis supervisor, Dr. Bruno Sudret, for his exceptional support and guidance throughout my research work. I attribute the level of my Ph.D degree to his permanent encouragement and involvement, which allowed among others the publication of many journal and conference papers. I could just not expect a better and friendlier supervisor.

I wish to thank the members of the jury, namely Pr. Anestis Antoniadis for having accepted to be its president, Pr. Gilles Fleury and Dr. Anthony Nouy for their careful reading and rating of my thesis report. I also thank Pr. Maurice Lemaire and Dr. Didier Lucor for having accepted to be part of the jury and for their relevant questions after my presentation.

In my daily work I was helped much by the members of the probabilistic analysis team, namely Marc Berveiller and Arsène Yameogo. They provided me great advices without fail, such as how to deal with Fortran or to couple simulation codes. We had fruitful discussions on stochastic methods as well as interesting debates on politics and society while sipping a Nespresso coffee.

I would like to thank the head of MMC department, Christophe Varé, as well as the former head of group T24, Stéphane Bugat, who supported my application to become a permanent research engineer at EDF. My greetings are also addressed to all my colleagues from T24 and T25 for their good mood and for creating a great atmosphere at work. The group assistants, Lydie Blanchard and Dominique Maligot, often guided me through the labyrinth of the EDF procedures with lots of patience and devotion. They also helped me reduce dramatically my stress level in October by fixing the last technical details for the D-day of my PhD defense. Besides, Clarisse Messelier-Gouze and Anna Dahl very nicely spent some time to explain me the problem of the integrity of a nuclear powerplant vessel.

Throughout my three years of Ph.D I have had the great pleasure to meet a very friendly group of fellow students and young employees. Some of them helped me regain some sort of fitness playing basketball. Gabrielle made sure none of us starved, Mary and the Glasgow staff made sure none of us went thirsty.

Last not least, I thank my brother for helping me move out from Auvergne to Fontainebleau: driving this big truck was quite an odyssey! I also wish to thank my parents for supporting me throughout all my studies. Without them this thesis work would not have been achieved.

Contents

Chapter 1

Introduction

1 Context

Mathematical models are widely used in many science disciplines, such as physics, biology and meteorology. They are aimed at better understanding and explaining real-world phenomena. These models are also extensively employed in an industrial context in order to analyze the behaviour of structures and products. This allows the engineers to design systems with ever increasing performance and reliability at an optimal cost.

In structural mechanics, mathematical models may range from simple analytical formulæ (*e.g.* simple applications in beam theory) to sets of partial differential equations (*e.g.* a general problem in continuum mechanics). Characterizing the behaviour of the system (*e.g.* identifying the stress or displacement fields of an elastic structure) may be not an easy task since a closed-form solution is generally not available. Then numerical solving schemes have to be employed, such as the finite difference or the finite element method.

From this viewpoint, the recent considerable improvements in *computer simulation* have allowed the analysts to handle models of ever increasing complexity. Therefore taking into account quite realistic constitutive laws as well as particularly fine finite element meshes have become affordable. Nonetheless, in spite of this increase in the accuracy of the models, computer simulation never predicts exactly the behaviour of a real-world complex system.

Three possible sources for such a discrepancy between simulation and experimental observations may be distinguished, namely:

- *Model inadequacy.* Any mathematical model is a simplification of a real-world system. Indeed, a model relies upon a certain number of more or less realistic hypotheses. Moreover, some significant physical phenomena of the real system behaviour may have been neglected.

- *Numerical errors.* They are directly related to the level of refinement of the employed discretization scheme (typically the density of a finite element mesh). Note that for many classes of mathematical models (*e.g.* an elliptic boundary value problem), such an error can be mastered by means of *a priori* or *a posteriori* estimates.

- *Input parameter uncertainty.* In practice the estimation of the input parameters of a model may be difficult or inaccurate, *i.e. uncertain.* Such an input uncertainty naturally leads to an uncertainty in the computer predictions. *Aleatoric* and *epistemic* uncertainty are commonly distinguished. Aleatoric uncertainty corresponds to inherent variability in a system (*e.g.* the number-of-cycles-to-failure of a sample specimen submitted to fatigue loading), whereas epistemic uncertainty is due to imperfect knowledge and may be reduced by gathering additional information (*e.g.* the compressive strength of concrete determined from few measurements).

Only the uncertainty in input parameters is addressed in this thesis. Thus it is assumed that both the model and the numerical solving scheme are sufficiently accurate to predict the behaviour of the system, which means that the equations are relevant to describe the underlying physical phenomena and that the approximations introduced in the computational scheme, if any, are mastered.

Various methods for dealing with uncertainty in the input parameters in the field of structural mechanics have been proposed, *e.g.* interval analysis (Moore, 1979; Dessombz et al., 2001), fuzzy logic (Zadeh, 1978; Rao and Sawyer, 1995), Dempster-Schefer random intervals (Dempster, 1968; Shafer, 1976). However, the present work focuses on the *probabilistic methods*, which are widely used and rely upon a sound mathematical framework. Probabilistic approaches in civil and mechanical engineering have received an increasing interest since the 1970's although pioneering contributions date back to the first half of the twentieth century (Mayer, 1926; Freudenthal, 1947; Lévi, 1949). They are based on the representation of uncertain input parameters by *random variables* or *random fields*.

2 General framework for probabilistic analysis

This section aims at presenting a general scheme for probabilistic analysis. The proposed framework, which is quite classical, has been formalized in the past few years at the R&D Division of EDF (De Rocquigny, 2006a,b; Sudret, 2007) together with various companies and academic research groups. It is sketched in Figure 1.1, see also various illustrations in De Rocquigny et al. (2008).

Several steps are identified in this framework:

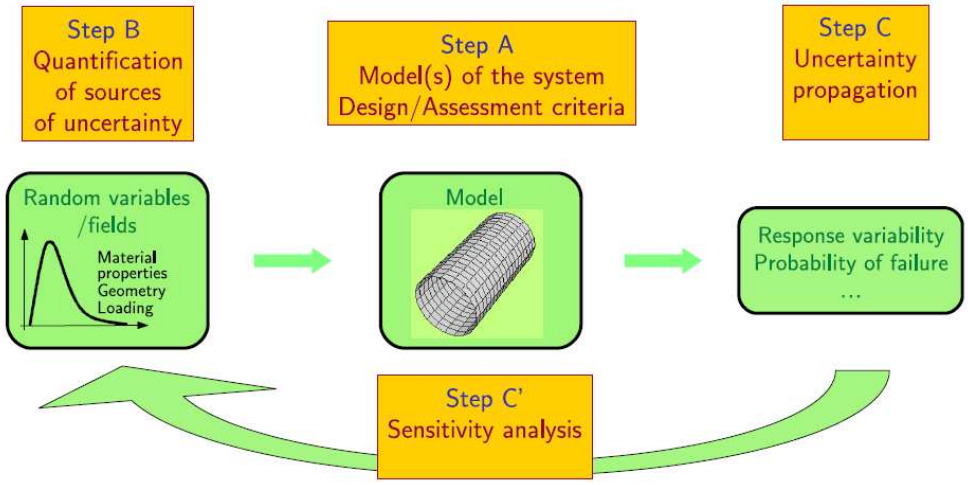


Figure 1.1: *General sketch for probabilistic uncertainty analysis (after Sudret (2007))*

- Step A consists in *defining the model* as well as associated criteria (*e.g.* safety criteria) that should be used in order to assess the system under consideration. This step gathers all the ingredients used for a classical *deterministic* analysis of the physical system to be analyzed.

- Step B consists in *identifying the uncertain input parameters* and modelling them by random variables or random fields.

- Step C consists in *propagating the uncertainty* in the input parameters through the deterministic model, *i.e.* characterizing the statistical properties of the output quantities of interest.

- Step C' consists in *hierarchizing the input parameters* according to their respective impact on the output variability. This study, known as *sensitivity analysis*, is carried out by post-processing the results obtained at the previous step.

The present work is focused on the uncertainty propagation problem (Step C), and also addresses the problem of sensitivity analysis (Step C').

3 Problem statement

Uncertainty propagation and quantification is a challenging problem in engineering. Indeed, the analyst often makes use of complex models in order to assess the reliability or to perform a robust design of industrial structures. This has the two following consequences:

- the relationship between the input and output parameters is often too complicated to be characterized prior to evaluating the model;
- each evaluation of the model (*i.e.* each run of the computer code) may be time-demanding.

The first point leads to consider the model of interest as a *black-box*, *i.e.* a function which can be known only through evaluations. Thus any *intrusive* method, *i.e.* any scheme consisting in adaptating the governing equations of the deterministic model, cannot be used. *Non intrusive* methods, *i.e.* methods which only make use of a series of calls to the deterministic model, have to be employed instead. The second point implies to adopt a method that minimizes the number of calls to the model. Therefore standard methods based on an intensive simulation of the model, such as so-called *Monte Carlo simulation*, should be avoided.

An interesting alternative is the *response surface methodology*. It basically consists in substituting the possibly complex model by a simple approximation, named *response surface* or *metamodel*, that is fast to evaluate. In this setup, the computational cost is focused on the fitting of the metamodel. The so-called *spectral methods* that are well-known in functional analysis (Boyd, 1989) appear to provide a polynomial metamodel onto a basis that is well suited to post-processing in uncertainty and sensitivity analysis. Such a metamodel is commonly referred to as *polynomial chaos* (PC) *expansion* (Ghanem and Spanos, 1991; Soize and Ghanem, 2004).

Metamodelling techniques generally reveal efficient and accurate provided that the model under consideration involves a moderate number of input parameters, say not greater than 10. However, the required computational cost may grow up for a larger number of variables. This may be a problem in practice, especially for applications featuring random fields, the discretization of whom possibly leading to a parametrization of the model in terms of many random variables, say ten to hundreds.

A second difficulty related to metamodels lies in the estimation of the approximation error. Such an assessment is of crucial importance since the accuracy of the results of an uncertainty or a sensitivity analysis will directly depend on the goodness-of-fit of the response surface. Furthermore, a reliable error estimate will allow to design an adaptive refinement of the metamodel until some prescribed accuracy is reached. It has to be noted that the error estimation should not require additional model evaluations (since the latter may be computationally expensive), but rather reuse the already performed computer experiments.

4 Objectives and outline of the thesis

Taking into account all the previous statements, the present work is aimed at devising a metamodelling technique in such a way that:

- quantities of interest of uncertainty and sensitivity analysis can be straightforwardly obtained by post-processing the response surface;
- the metamodel may be built up using a small number of model evaluations, even in the case of a high-dimensional problem. The dimensions under consideration range from 10 to 100;
- the approximation error is monitored;
- given a certain measure of accuracy, which is prescribed by the analyst, the metamodel is *adaptively* built and enriched until the target accuracy is attained.

From these objectives, the document is organized in five chapters that are now presented.

Chapter 2 contain a review of well-known methods for uncertainty propagation and sensitivity analysis. Various techniques for second moment, reliability and global sensitivity analysis are first addressed. Then a special focus is given to *metamodelling approaches*, and two recent methods are briefly described.

Chapter 3 introduces the so-called *spectral methods* that have emerged in the late 80's in probabilistic mechanics, and for which interest has dramatically grown up in the last five years. The spectral methods consist in representing the random model response as a series expansion onto a suitable basis, called *polynomial chaos* (PC) *expansion*. The coefficients of this expansion contain the whole probabilistic content of the random response. The so-called Galerkin (*i.e.* *intrusive*) computational scheme is presented. The following of the chapter reports the personal viewpoint of the author on the *non intrusive* approaches: the various methods are investigated in detail, related to each other and compared theoretically. Lastly techniques for post-processing the PC expansion are described.

The next chapters are devoted to the author's contributions in the field of uncertainty propagation and sensitivity analysis. Chapter 4 deals with the problem of the noticeable increase of the required number of model evaluations (*i.e.* the computational cost) when increasing either the degree of the PC expansion or the number of input parameters. To bypass this difficulty, new strategies for truncating the PC representation are introduced. They aim at favoring the PC terms that are related to the main effects and the low-order interactions. In order to further reduce the computational cost, an adaptive cut-off procedure close to stepwise regression is devised in order to automatically detect the most significant PC coefficients, resulting in a *sparse* representation. Inexpensive robust error estimates are investigated in order to assess the PC approximations.

Chapter 5 considers the building of a sparse PC expansion as a *variable selection* problem, well-known in statistics. Several variable selection methods are reviewed. A particularly efficient

approach known as *Least Angle Regression* (LAR) is given a special focus. LAR provides a set of more or less sparse PC approximations at the cost of an ordinary least-square regression scheme. Then criteria for selecting the best metamodel are investigated. Similarly to the stepwise regression scheme developed in the previous chapter, an adaptive version of LAR is devised in order to automatically increase the degree of the approximation until some target accuracy is reached.

Chapter 6 finally addresses various academic examples which show the potential of the methods proposed in the various chapters. In particular, examples featuring a large number of input variables resulting from random field discretization are tackled. An industrial application of the proposed methods is finally considered, namely the mechanical integrity of the vessel of a nuclear reactor.

Chapter 2

Metamodel methods for uncertainty propagation

1 Introduction

Physical systems are often represented by mathematical models, which may range from simple analytical formulæ to sets of partial differential equations. The latter may be solved using specific numerical schemes such as finite difference or finite element methods, which are implemented as simulation computer codes. In this work, the mathematical model of a physical system is described by a *deterministic* mapping $\mathcal{M}$ from $\mathbb{R}^M$ to $\mathbb{R}^Q$, where $M, Q \geq 1$. The function $\mathcal{M}$ has generally no explicit expression and can be known only point-by-point, *e.g.* when the computer code is run. In this sense $\mathcal{M}$ can be referred to as a *black-box* function. The model function depends on a set of M input parameters denoted by $\boldsymbol{x} \equiv \{x_1, \ldots, x_M\}^\mathsf{T}$, which typically represent material properties, geometrical properties or loading in structural mechanics. Each model evaluation $\mathcal{M}(\boldsymbol{x})$ returns a set of output quantities $\boldsymbol{y} \equiv \{y_1, \ldots, y_Q\}^\mathsf{T}$, called the model response vector. Vector $\boldsymbol{y}$ may for instance represent a set of displacements, strains or stresses at the nodes of a finite element mesh.

As the input vector $\boldsymbol{x}$ is assumed to be affected by uncertainty, a probabilistic framework is now introduced. Let $(\Omega, \mathcal{F}, \mathcal{P})$ be a probability space, where Ω is the event space equipped with σ-algebra $\mathcal{F}$ and probability measure $\mathcal{P}$. Throughout this document, random variables are denoted by upper case letters $X(\omega) : \Omega \to \mathcal{D}_\mathcal{X} \subset \mathbb{R}$, while their realizations are denoted by the corresponding lower case letters, *e.g.* x. Moreover, bold upper and lower case letters are used to denote random vectors (*e.g.* $\boldsymbol{X} = \{X_1, \ldots, X_M\}^\mathsf{T}$) and their realizations (*e.g.* $\boldsymbol{x} = \{x_1, \ldots, x_M\}^\mathsf{T}$), respectively.

Let us denote by $\mathcal{L}^2_\mathbb{R} \equiv \mathcal{L}^2(\Omega, \mathcal{F}, \mathcal{P}; \mathbb{R})$ the space of real random variables X with finite second

moments:

$$\mathbb{E}\left[X^2\right] \equiv \int_\Omega X^2(\omega)d\mathcal{P}(\omega) = \int_{\mathcal{D}_X} x^2 f_X(x)dx < +\infty \tag{2.1}$$

where $\mathbb{E}[\cdot]$ denotes the mathematical expectation operator and f_X (resp. $\mathcal{D}_X$) represents the probability density function (PDF) (resp. the support) of X (usual probability density functions are presented in Appendix A). This space is an Hilbert space with respect to the inner product:

$$\langle X_1, X_2\rangle_{\mathcal{L}^2_\mathbb{R}} \equiv \mathbb{E}[X_1 X_2] \equiv \int_\Omega X_1(\omega)X_2(\omega)d\mathcal{P}(\omega) \tag{2.2}$$

This inner product induces the norm $\|X\|_{\mathcal{L}^2_\mathbb{R}} \equiv \sqrt{\mathbb{E}[X^2]}$.

The input vector of the physical model $\mathcal{M}$ is represented as a random vector $\boldsymbol{X}(\omega), \omega \in \Omega$ with prescribed joint PDF $f_{\boldsymbol{X}}(\boldsymbol{x})$. The model response is also a random variable $Y(\omega) = \mathcal{M}(\boldsymbol{X}(\omega))$ by uncertainty propagation through the model $\mathcal{M}$, as illustrated in Figure 2.1.

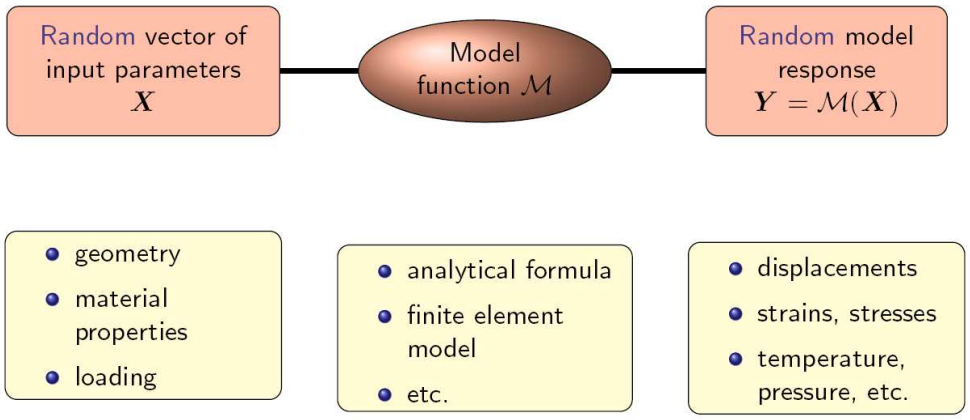


Figure 2.1: *General sketch for uncertainty propagation (after Sudret (2007))*

For the sake of simplicity, a scalar random response Y is considered in the sequel. The response Y can be completely characterized by its joint PDF denoted by $f_Y(y)$. However the latter has generally no analytical expression since it depends on the black box function $\mathcal{M}$. Consequently specific numerical methods have to be applied, depending on the type of study that is carried out. Four kinds of analyses are distinguished in this work, namely:

- *second moment analysis*, in which the mean value μ_Y and standard deviation σ_Y of the response Y are of interest;
- *sensitivity analysis*, which aims at quantifying the impact of each input random variable X_i on the response variability;
- *reliability analysis*, which consists in estimating the probability that the response exceeds a given threshold (here the tails of the distribution of random variable Y are given a special focus);

- *distribution analysis*, which is dedicated to the estimation of the *whole* PDF of Y.

Well-known methods used to solve these problems are reviewed in Section 2. They include in particular Monte Carlo simulation (MCS) that may be applied to solve each of the problems outlined above.

However MCS requires to perform a large number of evaluations of the model $\mathcal{M}$. This may lead to intractable calculations in case of a computationally demanding model, such as a finite element model of a complex industrial system. The so-called *metamodel* methods are intended to overcome this problem. They consist in substituting the model $\mathcal{M}$ for a simple (say analytical) approximation $\widehat{\mathcal{M}}$ which is fast to evaluate. Two recent methods are presented in Section 3, namely *Gaussian Process modelling* (Sacks et al., 1989; Santner et al., 2003) and *Support Vector Regression* (Vapnik et al., 1997; Smola and Schölkopf, 2006).

2 Standard methods for uncertainty propagation problems

In this section, classical methods for carrying out second moment, distribution, sensitivity and reliability analyses of a response quantity $Y \equiv \mathcal{M}(\boldsymbol{X})$ are addressed.

2.1 Methods for second moment analysis

Methods for computing the mean value and standard deviation of the response quantity are first adressed. The specific use of Monte Carlo simulation in this context is first presented. Confidence intervals on the results are derived. Then the so-called quadrature method is presented.

2.1.1 Monte Carlo simulation

Monte Carlo simulation can be used in order to estimate the mean value μ_Y and standard deviation σ_Y of a response quantity $Y \equiv \mathcal{M}(\boldsymbol{X})$. Consider a random sample $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$ drawn from the joint probability density function $f_{\boldsymbol{X}}$ of the input random vector $\boldsymbol{X}$. The usual estimators of the second moments read:

$$\hat{\mu}_Y \equiv \frac{1}{N} \sum_{i=1}^{N} \mathcal{M}(\boldsymbol{x}^{(i)}) \tag{2.3}$$

$$\hat{\sigma}_Y^2 \equiv \frac{1}{N-1} \sum_{i=1}^{N} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \hat{\mu}_Y \right)^2 \tag{2.4}$$

From a statistical point of view, the Monte Carlo estimates are random due to the randomness in the choice of the $\boldsymbol{x}^{(i)}$'s. According to the central limit theorem, the estimator in Eq.(2.3) is

asymptotically Gaussian with mean μ_Y (unbiased estimator) and standard deviation $\sigma_Y/\sqrt{N}$. This allows one to derive confidence intervals on $\hat{\mu}_Y$. It is also possible to compute confidence intervals on $\hat{\sigma}_Y$ as shown in Saporta (1990, Chapter 13).

Nonetheless, the convergence rate of the Monte Carlo-based estimators is quite slow, say $\propto N^{-1/2}$. Thus many model evaluations are usually required in order to reach a good accuracy. This may lead to intractable calculations in case of computationally demanding models. In the context of spectral methods developed in the next chapter, more efficient simulation schemes are proposed, namely *latin hypercube sampling* (LHS) (McKay et al., 1979) and *quasi-Monte Carlo* (QMC) methods (Niederreiter, 1992).

2.1.2 Quadrature method

By definition, the mean and variance of the model response may be cast as the following integrals:

$$\mu_Y \;\equiv\; \mathbb{E}\left[\mathcal{M}(\boldsymbol{X})\right] \;\equiv\; \int_{\mathcal{D}_{\boldsymbol{X}}} \mathcal{M}(\boldsymbol{x})\; f_{\boldsymbol{X}}(\boldsymbol{x})\; d\boldsymbol{x} \tag{2.5}$$

$$\sigma_Y^2 \;\equiv\; \mathbb{E}\left[\Big(\mathcal{M}(\boldsymbol{X}) - \mu_Y\Big)^2\right] \;\equiv\; \int_{\mathcal{D}_{\boldsymbol{X}}} \left(\mathcal{M}(\boldsymbol{x}) - \mu_Y\right)^2\; f_{\boldsymbol{X}}(\boldsymbol{x})\; d\boldsymbol{x} \tag{2.6}$$

This motivates the use of numerical integration techniques such as *quadrature* (Abramowitz and Stegun, 1970) in order to estimate the response second moments.

Let us describe first quadrature in a one-dimensional setting. One considers a scalar random variable X with prescribed probability density function (PDF) $f_X(x)$ and support $\mathcal{D}_X$. Let $X \mapsto h(X)$ be a given function of the random variable X, which is assumed to be square-integrable with respect to the PDF $f_X(x)$. Univariate quadrature allows one to approximate the mathematical expectation of the random variable $h(X)$ by a weighted sum as follows:

$$\mathbb{E}\left[h(X)\right] \;\equiv\; \int_{\mathcal{D}_X} h(x)\; f_X(x)\; dx \;\approx\; \sum_{i=1}^{n} w^{(i)}\; h(x^{(i)}) \tag{2.7}$$

In this expression, n is the *level* of the quadrature scheme and $\{(x^{(i)}, w^{(i)}), i = 1, \ldots, n\}$ are the quadrature nodes and weights, respectively.

Consider now a random vector $\boldsymbol{X} \equiv \{X_1, \ldots, X_M\}^{\mathsf{T}}$ with independent components. The joint PDF $f_{\boldsymbol{X}}$ of $\boldsymbol{X}$ reads:

$$f_{\boldsymbol{X}}(\boldsymbol{x}) \;=\; f_{X_1}(x_1) \times \cdots \times f_{X_M}(x_M) \tag{2.8}$$

Univariate quadrature formulæ may be derived for each marginal PDF f_{X_i}. Considering a $f_{\boldsymbol{X}}$-square-integrable mapping $\boldsymbol{X} \mapsto h(\boldsymbol{X})$, the quantity $\mathbb{E}\left[h(\boldsymbol{X})\right]$ can be estimated using a so-called *tensor-product quadrature* scheme as follows:

$$\mathbb{E}\left[h(\boldsymbol{X})\right] \;\equiv\; \int_{\mathcal{D}_{\boldsymbol{X}}} h(\boldsymbol{x})\; f_{\boldsymbol{X}}(\boldsymbol{x})\; d\boldsymbol{x} \;\approx\; \sum_{i_1=1}^{n_1} \cdots \sum_{i_M=1}^{n_M} w^{(i_1)} \cdots w^{(i_M)}\; h(x^{(i_1)}, \ldots, x^{(i_M)}) \tag{2.9}$$

Tensor-product quadrature may be straightforwardly applied to compute the mean value (resp. the variance) of the model response $\mathcal{M}(\boldsymbol{X})$ by setting $h(\boldsymbol{x}) \equiv \mathcal{M}(x)$ (resp. $h(\boldsymbol{x}) \equiv (\mathcal{M}(x) - \mu_Y)^2$).

The main drawback of this approach is the so-called *curse of dimensionality*. Suppose indeed that a n-level quadrature scheme is retained for each dimension. Then the M nested summations in Eq.(2.9) comprise n^M terms, which exponentially increases with the number of input variables M. In order to bypass this issue, *sparse* quadrature schemes (also known as Smolyak quadrature) are introduced in the next chapter.

2.2 Probability density functions of response quantities

In order to obtain a graphical representation of the response PDF, the random response Y may be simulated using a Monte Carlo scheme. This provides a sample set of response quantities $\{y^{(1)}, \dots, y^{(N)}\}$. An histogram may be built from this sample. Smoother representations may be obtained using *kernel smoothing* techniques, see *e.g.* Wand and Jones (1995).

Broadly speaking, the *kernel density approximation* of the response PDF is given by:

$$\hat{f}_Y(y) = \frac{1}{N_K\, h_K} \sum_{i=1}^{N_K} K\left(\frac{y - y^{(i)}}{h_K}\right) \tag{2.10}$$

In this expression, $K(x)$ is a suitable positive function called kernel, and h_K is the bandwith parameter. Well-known kernels are the Gaussian kernel (which is the standard normal PDF) and the Epanechnikov kernel $K_E(x) = \frac{3}{4}(1 - x^2)\, \mathbf{1}_{|x|\leq 1}$. Several values for h_K were proposed in Seather and Jones (1991). In practice one may select the following value when using a Gaussian kernel (Silverman, 1986):

$$h_K^* = \left(\frac{4}{3N_K}\right)^{1/5} \min\left(\widehat{\sigma}_Y, \widehat{iqr}_Y\right) \tag{2.11}$$

where $\widehat{\sigma}_Y$ and $\widehat{iqr}_Y$ respectively denote the empirical standard deviation and interquartile of the observed sample. Indeed, it is shown that h_K^* is optimal in the sense that it minimizes the approximation error $\|\hat{f}_Y - f_Y\|_{\mathcal{L}^2}$ of the response PDF.

In practice, a large sample set is necessary in order to obtain an accurate approximation, say $N = 10,000 - 100,000$.

2.3 Methods for reliability analysis

2.3.1 Introduction

Structural reliability analysis aims at computing the probability of failure of a mechanical system with respect to a prescribed failure criterion by accounting for uncertainties arising in the model

description (*e.g.* geometry, material properties) or the environment (*e.g.* loading). The reader is referred to classical textbooks for a comprehensive presentation (*e.g.* Ditlevsen and Madsen (1996); Melchers (1999); Lemaire (2005) among others). This section summarizes some well-established methods to solve reliability problems.

2.3.2 Problem statement

The mechanical system is supposed to fail when some requirements of safety or serviceability are not fulfilled. For each failure mode, a failure criterion is set up. It is mathematically represented by a *limit state function* $g(\boldsymbol{X}, \mathcal{M}(\boldsymbol{X}), \boldsymbol{X}')$. As shown in this expression, the limit state function may depend on input parameters, response quantities that are obtained from the model and possibly additional random variables and parameters gathered in $\boldsymbol{X}'$. For the sake of simplicity, the sole notation $\boldsymbol{X}$ is used in the sequel to refer to all random variables involved in the analysis. Let M be the size of $\boldsymbol{X}$.

Conventionnally, the limit state function is defined in such a way that:

- $\mathcal{D}_f \equiv \{\boldsymbol{x} \in \mathcal{D}_{\boldsymbol{X}} \,:\, g(\boldsymbol{x}) \,>\, 0\}$ is the *safe domain* in the space of parameters;
- $\mathcal{D}_f \equiv \{\boldsymbol{x} \in \mathcal{D}_{\boldsymbol{X}} \,:\, g(\boldsymbol{x}) \,\leq\, 0\}$ is the *failure domain.*

The set of points $\{\boldsymbol{x} \in \mathcal{D}_{\boldsymbol{X}} : g(\boldsymbol{x}) = 0\}$ defines the *limit state surface.* Denoting by $f_{\boldsymbol{X}}(\boldsymbol{x})$ the joint probability density function (PDF) of $\boldsymbol{X}$, the probability of failure of the system reads:

$$P_f \;\equiv\; \mathbb{P}\left[g(\boldsymbol{X}) \,\leq\, 0\right] \;\equiv\; \int_{\mathcal{D}_{\boldsymbol{X}}} \mathbf{1}_{\{g(\boldsymbol{x})\leq 0\}}(\boldsymbol{x})\; f_{\boldsymbol{X}}(\boldsymbol{x})\; d\boldsymbol{x} \;=\; \int_{\mathcal{D}_f} f_{\boldsymbol{X}}(\boldsymbol{x})\; d\boldsymbol{x} \tag{2.12}$$

where $\mathbf{1}_{\{g(\boldsymbol{x})\leq 0\}}(\boldsymbol{x}) = 1$ if $\{g(\boldsymbol{x}) \leq 0\}$ and 0 otherwise. As $\mathcal{M}$ is assumed to be a black box model, the failure domain $\mathcal{D}_f$ is not known explicitely and the integral in the previous equation cannot be computed directly. Instead numerical schemes are employed in order to obtain estimates of P_f.

2.3.3 Monte Carlo simulation

As shown in Section 2.1.1, Monte Carlo simulation may be used to approximate integrals such as in Eq.(2.12). Thus an estimator of the probability of failure is obtained by:

$$\widehat{P}_f \;\equiv\; \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{\{g(\boldsymbol{x})\leq 0\}}(\boldsymbol{x}^{(i)}) \;=\; \frac{N_{fail}}{N} \tag{2.13}$$

where the $\boldsymbol{x}^{(i)}$'s are randomly drawn from the joint input PDF $f_{\boldsymbol{X}}(\boldsymbol{x})$, and N_{fail} denotes the number of model evaluations that correspond to the failure of the system. $\widehat{P}_f$ is an unbiased

estimator of P_f, *i.e.* $\mathbb{E}\left[\widehat{P}_f\right] = P_f$. Moreover, the variance of $\widehat{P}_f$ is given by:

$$\mathbb{V}\left[\widehat{P}_f\right] \;=\; \frac{P_f\ (1-P_f)}{N} \tag{2.14}$$

For common values of the probability of failure (say $P_f << 1$), the coefficient of variation of the estimator asymptotically reads:

$$CV_{\widehat{P}_f} \;=\; \frac{\sqrt{\mathbb{V}\left[\widehat{P}_f\right]}}{P_f} \;\sim\; \frac{1}{\sqrt{P_f\ N}} \qquad , \quad N \to \infty \tag{2.15}$$

Suppose that P_f has the order of magnitude 10^{-k} and a coefficient of variation of 5% is required in its computation. The above equation shows that a number of model evaluations $N > 4 \cdot 10^{k+2}$ is required, which is clearly big when small values of P_f are sought.

Note that more efficient simulation schemes have been proposed in order to decrease the number of computer experiments, such as *directional simulation*, *importance sampling* (Ditlevsen and Madsen, 1996) or more recently *subset simulation* (Au and Beck, 2001) and some variants (Ching et al., 2005; Katafygiotis and Cheung, 2005).

2.3.4 First Order Reliability Method (FORM)

The *First Order Reliability Method* (FORM) has been introduced in order to get an estimate of P_f by means of a limited number of evaluations of the limit state function compared to crude Monte Carlo simulation.

First of all, the problem is recast in the standard normal space, *i.e.* the input random vector $\boldsymbol{X}$ is transformed into a Gaussian random vector $\boldsymbol{\xi}$ with independent components. This is achieved using an isoprobabilistic transform $\boldsymbol{X} \mapsto T(\boldsymbol{X}) \equiv \boldsymbol{\xi}$, such as the Nataf transform or the Rosenblatt transform (Ditlevsen and Madsen, 1996, Chap.7)[1]. Thus the probability of failure in Eq.(2.12) rewrites:

$$P_f \;\equiv\; \int_{\{\boldsymbol{x}:g(\boldsymbol{x})\leq 0\}} f_{\boldsymbol{X}}(\boldsymbol{x})\ d\boldsymbol{x} \;=\; \int_{\{\boldsymbol{\xi}:g(T^{-1}(\boldsymbol{\xi}))\leq 0\}} \varphi_M(\boldsymbol{\xi})\ d\boldsymbol{\xi} \tag{2.16}$$

where φ_M is the standard multinormal PDF defined by:

$$\varphi_M(\boldsymbol{\xi}) \;\equiv\; (2\pi)^{-M/2}\ \exp\left(-\frac{\|\boldsymbol{\xi}\|_2^2}{2}\right) \tag{2.17}$$

This PDF shows a peak at the origin $\boldsymbol{\xi} = \boldsymbol{0}$ and decreases exponentially with $\|\boldsymbol{\xi}\|_2^2$. Thus the points that contribute at most to the integral in Eq.(2.16) are those in the failure domain that are closest to the origin of the space (Figure 2.2).

[1]Note that such a transformation can be carried out *exactly* only in specific cases. For instance the Nataf transform may be only applied to random vectors $\boldsymbol{X}$ with an *elliptical copula*, such as the Gaussian one (see Lebrun and Dutfoy (2009) for more details). Otherwise one may seek an approximation $\widehat{T}$ of T.

As a consequence, the second step of FORM consists in determining the so-called *design point*, *i.e.* the point in the failure domain closest to the origin of the standard space. This point $\boldsymbol{\xi}^*$ is the solution of the following optimization problem:

$$P^* \equiv \boldsymbol{\xi}^* = \arg\min_{\boldsymbol{\xi}} \left\{ \|\boldsymbol{\xi}\|_2^2 : g\left(T^{-1}(\boldsymbol{\xi})\right) \leq 0 \right\} \tag{2.18}$$

Once the design point has been found, it is possible to define the *reliability index* by:

$$\beta \equiv \text{sign}\left[g\left(T^{-1}(\mathbf{0})\right)\right] \cdot \|\boldsymbol{\xi}^*\|_2 \tag{2.19}$$

It corresponds to the algebraic distance of the design point to the origin, which is counted as positive if the origin is in the safe domain, and negative otherwise.

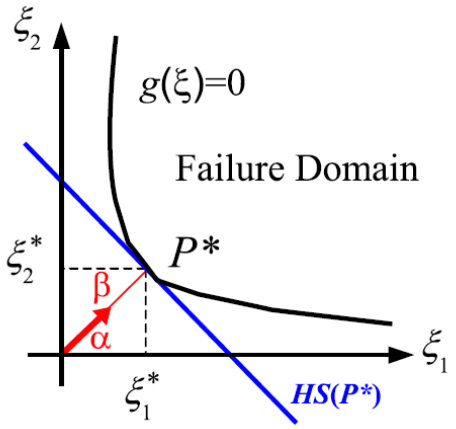


Figure 2.2: *Principle of the First Order Reliability Method (FORM)*

The third step of FORM consists in substituting the failure domain for the half space $HS(\boldsymbol{\xi}^*)$ defined by means of the hyperplane that is tangent to the limit state surface at the design point. The equation of this hyperplane may be cast as:

$$\beta - \boldsymbol{\alpha} \cdot \boldsymbol{\xi} = 0 \tag{2.20}$$

where the unit vector $\boldsymbol{\alpha} = \boldsymbol{\xi}^*/\beta$ is normal to the limit state surface at the design point (see Figure 2.2):

$$\boldsymbol{\alpha} = -\frac{\nabla g(T^{-1}(\boldsymbol{\xi}^*))}{\|\nabla g(T^{-1}(\boldsymbol{\xi}^*))\|} \tag{2.21}$$

This leads to:

$$P_f \equiv \int_{\{\boldsymbol{\xi}: g(T^{-1}(\boldsymbol{\xi}))\leq 0\}} \varphi_M(\boldsymbol{\xi})\, d\boldsymbol{\xi} \approx \int_{HS(\boldsymbol{\xi}^*))} \varphi_M(\boldsymbol{\xi})\, d\boldsymbol{\xi} \tag{2.22}$$

The latter integral can be evaluated analytically and provides the first order approximation of the probability of failure:

$$P_f \approx P_{f,\text{FORM}} \equiv \Phi(-\beta) \tag{2.23}$$

where Φ denotes the standard normal cumulative distribution function.

A more accurate estimation of the probability of failure may be obtained by deriving a second-order Taylor series expansion of the limit state function around the design point. This is essentially what the so-called *second-order reliability method* (SORM) (Breitung, 1984) does.

It should be noted that the FORM and SORM approaches may provide erroneous results if the optimization problem in Eq.(2.18) is non-convex. Indeed, the two following problems could then arise:

- the adopted solving scheme could thus converge to a *local* minima and then miss the region of dominant contribution to the probability of failure;
- even if the global design point is detected, there could be significant contributions to the probability of failure from the vicinity of the local design points.

A simple and heuristic method based on series system reliability analysis has been proposed in Der Kiureghian and Dakessian (1998) in order to tackle those problems featuring multiple design points.

2.3.5 Importance sampling

FORM allows the analyst to compute an estimate of the probability of failure at a low computational cost compared to Monte Carlo simulation (MCS). However, the FORM estimate in Eq.(2.23) might reveal inaccurate in case of a complex limit state function. Furthermore, no error estimate is associated with the FORM results, in contrast to MCS. To bypass these difficulties, a method that both uses FORM and simulation may be employed, namely *importance sampling* (IS) (Harbitz, 1983; Shinozuka, 1983).

Consider that a FORM analysis has been carried out and that the design point $\boldsymbol{\xi}^*$ has been identified. Let us define an auxiliary PDF, called *importance sampling density*, by:

$$\psi(\boldsymbol{\xi}) \;\equiv\; \varphi_M(\boldsymbol{\xi} - \boldsymbol{\xi}^*) \tag{2.24}$$

The probability of failure may be recast as:

$$P_f \;=\; \int_{\{\boldsymbol{\xi}: g(T^{-1}(\boldsymbol{\xi}))\leq 0\}} \frac{\varphi_M(\boldsymbol{\xi})}{\psi(\boldsymbol{\xi})}\, \psi(\boldsymbol{\xi})\, d\boldsymbol{\xi} \tag{2.25}$$

The corresponding Monte Carlo estimate of P_f reads:

$$P_{f,\mathrm{IS}} \;=\; \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{\{g(T^{-1}(\boldsymbol{\xi}))\leq 0\}}(\boldsymbol{\xi}^{(i)})\, \frac{\varphi_M(\boldsymbol{\xi}^{(i)})}{\psi(\boldsymbol{\xi}^{(i)})} \tag{2.26}$$

where the $\boldsymbol{\xi}^{(i)}$'s are now randomly generated from the auxiliary PDF $\psi(\boldsymbol{\xi})$. This expression rewrites:

$$P_{f,\mathrm{IS}} \;=\; \frac{\exp[\beta^2/2]}{N}\sum_{i=1}^{N} \mathbf{1}_{\{g(T^{-1}(\boldsymbol{\xi}))\leq 0\}}(\boldsymbol{\xi}^{(i)})\, \exp\left[-\boldsymbol{\xi}^{(i)}\cdot\boldsymbol{\xi}^*\right] \tag{2.27}$$

The IS estimate converges more rapidly than the MC one. Moreover, as any random sampling method, IS comes with confidence intervals on the result. In practice, this allows the monitoring of the simulation according to the coefficient of variation of the estimate.

2.4 Methods for sensitivity analysis

2.4.1 Introduction

Quantifying the contribution of each input parameter X_i to the variability of the response Y is of major interest in practical applications. This may not be an easy task in practice though since the relationship between the input and the output variables of a complex model is not straightforward. Sensitivity analysis (SA) methods allow the analyst to address this problem. A great deal of SA methods may be found in the literature, see a review in Saltelli et al. (2000) and recent advances in Xu and Gertner (2008). They are typically separated into two groups:

- *local* sensitivity analysis, which studies how little variations of the input parameters in the vicinity of given values influence the model response;
- *global* sensitivity analysis, which is related with quantifying the output uncertainty due to changes of the input parameters (which are taken singly or in combination with others) over their entire domain of variation.

Many papers have been devoted to the latter topic in the last two decades. The state-of-the-art review available in Saltelli et al. (2000) gathers the related methods into two groups:

- *regression-based methods*, which exploit the results of the linear regression of the model response on the input vector. These approaches are useful to measure the effects of the input variables if the model is linear or almost linear. However, they fail to produce satisfactory sensitivity measures in case of significant nonlinearity (Saltelli and Sobol', 1995).
- *variance-based methods*, which rely upon the decomposition of the response variance as a sum of contributions of each input variables, or combinations thereof. They are known as *ANOVA* (for *ANalysis Of VAriance*) techniques in statistics (Efron and Stein, 1981). The Fourier amplitude sensitivity test (FAST) indices (Cukier et al., 1978; Saltelli et al., 1999) and the Sobol' indices (Sobol', 1993; Saltelli and Sobol', 1995; Archer et al., 1997) enter this category, see also the reviews in McKay (1995); Sobol' and Kucherenko (2005).

Other global sensitivity analysis techniques are available, such as the Morris method (Morris, 1991), sampling methods (Helton et al., 2006) and methods based on experimental designs,

e.g. fractional factorial designs (Saltelli et al., 1995) and Plackett-Burman designs (Beres and Hawkins, 2001). In the sequel, the attention is focused on Sobol' indices.

2.4.2 Global sensitivity analysis

It is assumed that Y has a finite variance and that the components of $\boldsymbol{X}$ are *independent*. As shown first in Hoeffding (1948) and later in Efron and Stein (1981), the model response $\mathcal{M}(\boldsymbol{X})$ may be decomposed into main effects and interactions[2]:

$$Y = \mathcal{M}(\boldsymbol{X}) = \mathcal{M}_0 + \sum_{i=1}^{M} \mathcal{M}_i(X_i) + \sum_{1\leq i<j\leq M} \mathcal{M}_{i,j}(X_i, X_j) + \cdots + \mathcal{M}_{1,2,\ldots,M}(\boldsymbol{X}) \quad (2.28)$$

The uniqueness of the decomposition is ensured by choosing summands that satisfy the following properties (Sobol', 1993):

$$\mathcal{M}_0 = \int_{\mathcal{D}_{\boldsymbol{X}}} \mathcal{M}(\boldsymbol{x})\, f_{\boldsymbol{X}}(\boldsymbol{x})\, d\boldsymbol{x} \quad (2.29)$$

$$\int_{\mathcal{D}_{X_k}} \mathcal{M}_{i_1,\ldots,i_s}(\boldsymbol{x}_{i_1}, \ldots, \boldsymbol{x}_{i_s})\, f_{X_k}(x_k)\, dx_k = 0 \qquad 1 \leq i_1 < \cdots < i_s \leq M \qquad k \in \{i_1, \ldots, i_s\} \quad (2.30)$$

where $\mathcal{D}_{\boldsymbol{X}}$ is the support of random vector $\boldsymbol{X}$ and $\mathcal{D}_{X_k}$ (resp. $f_{X_k}(x_k)$) is the support (resp. margin PDF) of random variable X_k. The decomposition is then often referred to as ANOVA-decomposition (for ANalysis Of VAriance).

Remark 1. *Other types of decompositions may be obtained by substituting the image probability measure $d\mathcal{P}_{\boldsymbol{X}} \equiv f_{\boldsymbol{X}}(\boldsymbol{x})d\boldsymbol{x}$ for another measure in Eqs.(2.29),(2.30). For instance the multivariate Dirac measure $\delta_{\boldsymbol{x}_0} d\boldsymbol{x} \equiv \prod_{i=1}^{M} \delta_{x_{0,i}} dx_i$ located at some point $\boldsymbol{x}_0 = \{x_{0,1}, \ldots, x_{0,M}\}$ was considered in Rabitz et al. (1999) under the name cut-HDMR (for High Dimensional Model Representation). In this setup, the summands in Eq.(2.28) are given by:*

$$\begin{aligned} \mathcal{M}_0 &= \mathcal{M}(\boldsymbol{x}_0) \\ \mathcal{M}_i(x_i) &= \mathcal{M}(x_i, \boldsymbol{x}_0^{-i}) - \mathcal{M}_0 \\ \mathcal{M}_{i,j}(x_i, x_j) &= \mathcal{M}(x_i, x_j, \boldsymbol{x}_0^{-i,j}) - \mathcal{M}_i(x_i) - \mathcal{M}_j(x_j) - \mathcal{M}_0 \end{aligned} \quad (2.31)$$

and so on, where:

$$\begin{aligned} \boldsymbol{x}_0^{-i} &\equiv \{x_{0,1}, \ldots, x_{0,i-1}, x_i, x_{0,i+1}, \ldots, x_{0,M}\} \\ \boldsymbol{x}_0^{-i,j} &\equiv \{x_{0,1}, \ldots, x_{0,i-1}, x_i, x_{0,i+1}, \ldots, x_{0,j-1}, x_j, x_{0,j+1}, \ldots, x_{0,M}\} \end{aligned} \quad (2.32)$$

It may be shown that for a differentiable function $\mathcal{M}$, such a decomposition corresponds to a rearrangement of the terms of the multivariate Taylor expansion of $\mathcal{M}$ around $\boldsymbol{x}_0$ (Rabitz et al., 1999; Griebel, 2006).

[2]It may be shown that the decomposition holds for any absolutely integrable function in L^1, see *e.g.* Griebel (2006, Section 1.3.1).

Properties (2.29),(2.30) correspond to the following summands in the ANOVA-decomposition:

$$\begin{aligned} \mathcal{M}_0 &= \mathbb{E}\left[\mathcal{M}(\boldsymbol{X})\right] \\ \mathcal{M}_i(X_i) &= \mathbb{E}\left[\mathcal{M}(\boldsymbol{X})|X_i\right] - \mathcal{M}_0 \\ \mathcal{M}_{i,j}(X_i,X_j) &= \mathbb{E}\left[\mathcal{M}(\boldsymbol{X})|X_i,X_j\right] - \mathcal{M}_i(X_i) - \mathcal{M}_j(X_j) - \mathcal{M}_0 \end{aligned} \tag{2.33}$$

and so on, where $\mathbb{E}\left[\cdot|\cdot\right]$ denotes the conditional expectation operator. It may be shown that the summands except $\mathcal{M}_0$ are mutually orthogonal, that is:

$$\begin{gathered} \mathbb{E}\left[\mathcal{M}_{i_1,\ldots,i_s}(X_{i_1},\ldots,X_{i_s})\mathcal{M}_{j_1,\ldots,j_t}(X_{j_1},\ldots,X_{j_t})\right] = 0 \\ s,t = 1,\ldots,M \quad , \quad \{i_1,\ldots,i_s\} \neq \{j_1,\ldots,j_t\} \end{gathered} \tag{2.34}$$

This allows one to derive directly the variance of the decomposition (2.28) as follows:

$$\mathbb{V}\left[\mathcal{M}(\boldsymbol{X})\right] \equiv D = \sum_{i=1}^{M} D_i + \sum_{1\leq i<j\leq M} D_{i,j} + \cdots + D_{1,\ldots,M} \tag{2.35}$$

where the $D_{i_1,\ldots,i_s}$'s are referred to as the *partial variances* and are defined by:

$$D_{i_1,\ldots,i_s} = \mathbb{V}\left[\mathcal{M}_{i_1,\ldots,i_s}(X_{i_1},\ldots,X_{i_s})\right] \quad , \quad s = 1,\ldots,M \tag{2.36}$$

This leads to the following definition of the *sensitivity index* of the model response to the subset of input random variables $\{X_{i_1},\ldots,X_{i_s}\}$ (Sobol', 1993):

$$S_{i_1,\ldots,i_s} = \frac{D_{i_1,\ldots,i_s}}{D} \tag{2.37}$$

Moreover the *total sensitivity indices* S_i^T have been defined in order to evaluate the total effect of an input parameter (Homma and Saltelli, 1996). They are defined from the sum of all partial sensitivity indices $D_{i_1\ldots i_s}$ involving parameter i:

$$S_i^T = \sum_{\{i_1\ldots i_s\}\in\mathcal{I}_i} D_{i_1\ldots i_s} \,/\, D \quad , \quad \mathcal{I}_i \equiv \{\{i_1,\ldots,i_s\} \supset \{i\}\} \tag{2.38}$$

It is easy to show that:

$$S_i^T = 1 - \frac{D_{-i}}{D} \tag{2.39}$$

where D_{-i} is the sum of all $D_{i_1,\ldots,i_s}$ that do *not* include index i.

2.4.3 Estimation by Monte Carlo simulation

In practice, the sensitivity indices (2.37)-(2.38) are estimated using Monte Carlo simulation (see *e.g.* Sobol' (1993)). As shown in Saltelli et al. (2000), the mean value and the total variance may be estimated using N random samples of $\boldsymbol{X}$:

$$\widehat{\mathcal{M}_0} \equiv \frac{1}{N}\sum_{i=1}^{N} \mathcal{M}(\boldsymbol{x}^{(i)}) \tag{2.40}$$

$$\widehat{D} \equiv \frac{1}{N}\sum_{i=1}^{N} \mathcal{M}(\boldsymbol{x}^{(i)})^2 - \widehat{\mathcal{M}}_0^2 \tag{2.41}$$

The estimation of the total variances D_i associated with a single variable X_i requires *two* independent sample sets $\mathcal{X} \equiv \{\boldsymbol{x}^{(i)}, i = 1, \ldots, N\}$ and $\mathcal{Z} \equiv \{\boldsymbol{z}^{(i)}, i = 1, \ldots, N\}$. Using the following notation:

$$\boldsymbol{x}^{(k)} \equiv \{x_1^{(k)}, \ldots, x_M^{(k)}\} \quad , \quad \boldsymbol{z}^{(k)} \equiv \{z_1^{(k)}, \ldots, z_M^{(k)}\} \quad , \quad k = 1, \ldots, N \tag{2.42}$$

the Monte Carlo estimate of D_i reads:

$$\widehat{D}_i \equiv \frac{1}{N}\sum_{k=1}^{N} \mathcal{M}\left(x_1^{(k)}, \ldots, x_M^{(k)}\right) \mathcal{M}\left(z_1^{(k)}, \ldots, z_{i-1}^{(k)}, x_i^{(k)}, z_{i+1}^{(k)}, \ldots, z_M^{(k)}\right) - \widehat{\mathcal{M}}_0^2 \tag{2.43}$$

Formulæ similar to Eq.(2.44) may be obtained for deriving the partial variances of higher order, see Homma and Saltelli (1996).

The quantity D_{-i} that is needed for computing the total index S_i^T (Eq.(2.39)) is estimated by:

$$\widehat{D}_{-i} \equiv \frac{1}{N}\sum_{k=1}^{N} \mathcal{M}\left(x_1^{(k)}, \ldots, x_M^{(k)}\right) \mathcal{M}\left(x_1^{(k)}, \ldots, x_{i-1}^{(k)}, z_i^{(k)}, x_{i+1}^{(k)}, \ldots, x_M^{(k)}\right) - \widehat{\mathcal{M}}_0^2 \tag{2.44}$$

Hence the Monte Carlo estimates of the S_i^T's:

$$\widehat{S}_i^T = 1 - \frac{\widehat{D}_{-i}}{\widehat{D}} \tag{2.45}$$

The Sobol' indices are known to be good descriptors of the sensitivity of the model response to its input parameters, since they do not suppose any kind of linearity or monotonicity in the model. However, a limitation of the method is that 2^M Monte Carlo integrals are necessary in order to obtain a full description of the model, which is not practically feasible unless M is low (Saltelli et al., 2000). In practice, the analyst often only computes the first-order and the total sensitivity indices, and sometimes the second-order ones.

3 Methods based on metamodels

3.1 Introduction

As shown in the previous section, standard methods for uncertainty propagation and sensitivity analysis rely upon a large number of calls to the model under consideration. This may lead to intractable calculations if the model function $\mathcal{M}$ is expensive to evaluate. To bypass this problem, it is possible to approximate $\mathcal{M}$ by some analytical function whose evaluation in inexpensive. Such an approximation is often referred to as a *surrogate model*, *response surface*

or *metamodel* in the literature. In this setting, once an accurate metamodel has been determined, the classical intensive simulation schemes may be applied at a negligible computational cost. Various metamodelling techniques have been proposed in the literature, such as polynomial response surfaces, Gaussian Process modelling, Multivariate Adaptive Regression Splines (MARS), regression trees, artificial neural network, etc. The reader is referred to Chen et al. (2006) for a comprehensive review.

In this section, emphasis is put on two metamodelling techniques which revealed particularly efficient in some comparative studies (Jin et al., 2001; Clarke et al., 2003). *Gaussian process modelling* (Sacks et al., 1989; Santner et al., 2003) is presented first (Section 3.2). Then a specific regression technique known as *Support Vector Regression* (Vapnik et al., 1997; Smola and Schölkopf, 2006) is reviewed (Section 3.3).

3.2 Gaussian process modelling

Kriging (Matheron, 1967) is a well-established method in geostatistics in order to predict the value of a physical random field $\mathsf{M}(\boldsymbol{x}, \omega)$ at an unobserved location $\boldsymbol{x}$, from a set of observed values at given locations $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$. In this setting, $\boldsymbol{x}$ is a set of spatial coordinates.

The method has been recently generalized as a tool for approximating the response $y \equiv \mathcal{M}(\boldsymbol{x})$ of deterministic computer models by assuming that $y \equiv \mathcal{M}(\boldsymbol{x})$ is a sample path of an underlying *random field* $\mathsf{M}(\boldsymbol{x}, \omega)$, see *e.g.* Sacks et al. (1989); Welch et al. (1992); Koehler and Owen (1996); Santner et al. (2003); Vazquez and Walter (2003); O'Hagan (2006); Vazquez (2005); Vazquez et al. (2006). In this context, $\boldsymbol{x}$ is a vector of input parameters and kriging is often referred to as *Gaussian process* (GP) *modelling*. The stochastic framework of GP might seem irrelevant for approximating deterministic functions. Actually the statistical assumptions must be viewed as a way to represent the uncertainty associated with the prediction $y \equiv \mathcal{M}(\boldsymbol{x})$ of the model at a new set of input parameters $\boldsymbol{x}$. GP takes benefit of this mathematical foundation to build up the most probable metamodel given the set of already performed computer experiments, as shown in the sequel.

3.2.1 Stochastic representation of the model function

Let us consider the deterministic model $\boldsymbol{x} \mapsto \mathcal{M}(\boldsymbol{x})$ of a physical system. Assume that the function $\mathcal{M}$ in only known in N points $\{(\boldsymbol{x}^{(i)}, y^{(i)}), i = 1, \ldots, N\}$. The model output $y \equiv \mathcal{M}(\boldsymbol{x})$ is considered to be a sample path of an underlying Gaussian random field $\mathsf{M}(\boldsymbol{x}, \omega)$. Denoting by $\mu(\boldsymbol{x})$ the mean of the random field $\mathsf{M}(\boldsymbol{x})$, the latter may be cast as:

$$\mathsf{M}(\boldsymbol{x}, \omega) \;\equiv\; \mu(\boldsymbol{x}) \,+\, \mathsf{Z}(\boldsymbol{x}, \omega) \tag{2.46}$$

where $\mathsf{Z}(\boldsymbol{x})$ is a zero-mean random field. The mean $\mu(\boldsymbol{x})$ is usually cast as a linear combination of known deterministic functions $\{r_j(\boldsymbol{x}), j = 1, \ldots, m\}$ as follows:

$$\mu(\boldsymbol{x}) \;=\; \sum_{j=0}^{m} \beta_j \, r_j(\boldsymbol{x}) \;\equiv\; \boldsymbol{r}^{\mathsf{T}}(\boldsymbol{x})\boldsymbol{\beta} \tag{2.47}$$

In this setting, the mean $\mu(\boldsymbol{x})$ may represent a large-scale trend of the model output, *e.g.* a constant, a linear or a polynomial behaviour. In contrast, the random field $\mathsf{Z}(\boldsymbol{x}, \omega)$ may correspond to small-scale fluctuations of the model response. In practice separating both effects is not an easy task, and the choice of the complexity of the regression term is often arbitrary. A constant or a linear mean function $\mu(\boldsymbol{x})$ are commonly selected in practice (Sacks et al., 1989). Moreover, it is assumed that the random field $\mathsf{Z}(\boldsymbol{x}, \omega)$ satisfies a second-order stationarity property, *i.e.* its variance fullfils $\sigma^2(\boldsymbol{x}) = \sigma^2$ for all $\boldsymbol{x}$. Lastly, $\mathsf{Z}(\boldsymbol{x}, \omega)$ is supposed to be a Gaussian random field. Hence it is completely determined by its variance σ^2 and its autocorrelation function denoted by $\rho(\boldsymbol{x}, \boldsymbol{x}')$.

3.2.2 Conditional distribution of the model response

Let us consider a set of realizations $\mathcal{X} \equiv \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}^{\mathsf{T}}$ of the input random vector $\boldsymbol{X}$. $\mathcal{X}$ is referred to as the experimental design. Let $\mathcal{Y} \equiv \{\mathcal{M}(\boldsymbol{x}^{(1)}), \ldots, \mathcal{M}(\boldsymbol{x}^{(N)})\}^{\mathsf{T}} = \{y^{(1)}, \ldots, y^{(N)}\}^{\mathsf{T}}$ be the corresponding set of computed model evaluations. The GP approach consists in building a metamodel (*i.e.* an analytical approximation of $\mathcal{M}(\boldsymbol{x})$) from the conditional distribution of $\mathsf{M}(\boldsymbol{x}, \omega)$ with respect to the observations in $\mathcal{X}$.

Let us introduce the following vector/matrix notation:

$$\boldsymbol{k}(\boldsymbol{x}) \;\equiv\; \left\{\rho\left(\boldsymbol{x}, \boldsymbol{x}^{(1)}\right), \ldots, \rho\left(\boldsymbol{x}, \boldsymbol{x}^{(N)}\right)\right\}^{\mathsf{T}} \tag{2.48}$$

$$\boldsymbol{R} \;\equiv\; \left(r_j(\boldsymbol{x}^{(i)})\right)_{1 \leq i,j \leq N} \qquad , \qquad \boldsymbol{K} \;\equiv\; \left(\rho(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})\right)_{1 \leq i,j \leq N} \tag{2.49}$$

It may be shown that the conditional model response at untried $\boldsymbol{x}$ follows a Gaussian distribution with mean and variance respectively given by:

$$\mu_{\mathsf{M}}(\boldsymbol{x}) \;=\; \boldsymbol{r}^{\mathsf{T}}(\boldsymbol{x})\boldsymbol{\beta} \,+\, \boldsymbol{k}^{\mathsf{T}}(\boldsymbol{x})\boldsymbol{K}^{-1}\left(\mathcal{Y} \,-\, \boldsymbol{R}\boldsymbol{\beta}\right) \tag{2.50}$$

$$\sigma^2_{\mathsf{M}}(\boldsymbol{x}) \;=\; \sigma^2 \,-\, \boldsymbol{k}^{\mathsf{T}}(\boldsymbol{x})\, \boldsymbol{K}^{-1}\, \boldsymbol{k}^{\mathsf{T}}(\boldsymbol{x}) \tag{2.51}$$

Nonetheless such derivations rely upon the assumption that the properties of the random field $\mathsf{M}(\boldsymbol{x})$ are perfectly known, which is *not* the case in practical applications.

A specific family of parametrized autocorrelation functions is often selected *a priori*. In the literature on computer experiments, one usually uses *tensorized stationary* autocorrelation functions of the form:

$$\rho(\boldsymbol{x}, \boldsymbol{x}') \;\equiv\; \prod_{i=1}^{M} \rho_i(x_i - x'_i) \tag{2.52}$$

In particular, the following *exponential-type* autocorrelation functions have received great interest:

$$\rho(\boldsymbol{x}, \boldsymbol{x}') \;\equiv\; \prod_{i=1}^{M} e^{-(\theta_i |x_i - x_i'|)^{\gamma_i}} \tag{2.53}$$

where the θ_i are non-negative constants and $0 < \gamma_i \leq 2$. Note that setting $\gamma_1 = \cdots = \gamma_M = k$ with $k = 1$ (resp. $k = 2$) yields the *exponential* (resp. *Gaussian*) autocorrelation function. From now on, the parameters of the autocorrelation function are gathered in a vector denoted by $\boldsymbol{\theta}$.

In this setting, characterizing the metamodel leads to estimate the following parameters:

- the parameters of the mean function $\mu(\boldsymbol{x})$, namely the vector $\boldsymbol{\beta}$ (regression part);
- the variance σ^2;
- the autocorrelation parameters $\boldsymbol{\theta}$.

3.2.3 Estimation of the GP parameters

The *maximum likelihood* (ML) method is commonly applied to estimate the GP parameters from the data $(\mathcal{X}, \mathcal{Y})$. The ML estimate of $\boldsymbol{\beta}$ (denoted by $\widehat{\boldsymbol{\beta}}$) may be cast as a function of $\boldsymbol{\theta}$ as follows:

$$\widehat{\boldsymbol{\beta}} \;=\; \left(\boldsymbol{R}^{\mathsf{T}} \boldsymbol{K}_{\boldsymbol{\theta}}^{-1} \boldsymbol{R}\right)^{-1} \; \boldsymbol{R}^{\mathsf{T}} \boldsymbol{K}_{\boldsymbol{\theta}}^{-1} \mathcal{Y} \tag{2.54}$$

where the subscript $\boldsymbol{\theta}$ has been introduced to stress the dependency on the autocorrelation parameters. $\widehat{\boldsymbol{\beta}}$ is sometimes referred to as the *generalized least-square* estimate of $\boldsymbol{\beta}$. On the other hand, the ML estimate of the variance σ^2 reads as a function of $\boldsymbol{\theta}$:

$$\widehat{\sigma}^2 \;=\; \frac{1}{N} \left(\mathcal{Y} - \boldsymbol{R}\widehat{\boldsymbol{\beta}}^{\mathsf{T}}\right)^{\mathsf{T}} \boldsymbol{K}_{\boldsymbol{\theta}}^{-1} \left(\mathcal{Y} - \boldsymbol{R}\widehat{\boldsymbol{\beta}}^{\mathsf{T}}\right) \tag{2.55}$$

The MLE estimates of the autocorrelation parameters in vector $\boldsymbol{\theta}$ may be obtained by solving (Sacks et al., 1989):

$$\widehat{\boldsymbol{\theta}} \;=\; \arg\min_{\boldsymbol{\theta}} \; (\det \boldsymbol{K}_{\boldsymbol{\theta}})^{1/N} \; \widehat{\sigma}^2 \tag{2.56}$$

It appears that most of numerical solving schemes reveal some limitations when dealing with a large number M of input parameters $\boldsymbol{x} \equiv \{x_1, \ldots, x_M\}^{\mathsf{T}}$. Indeed, considering the exponential-type correlation model (2.53), the number of autocorrelation hyperparameters is equal to $2M$, hence an optimization problem of size $2(M+1)$. To circumvent this difficulty, a variable selection procedure has been proposed in Marrel et al. (2007); Marrel (2008), which allows a reduction of the computational cost.

3.2.4 GP metamodel

Once the GP parameters have been estimated, the mean function in Eq.(2.50) may be approximated by:

$$\widehat{\mu}_{\mathsf{M}}(\boldsymbol{x}) \equiv \widehat{\mathcal{M}}(\boldsymbol{x}) = \boldsymbol{r}^{\mathsf{T}}(\boldsymbol{x})\widehat{\boldsymbol{\beta}} + \boldsymbol{k}_{\hat{\boldsymbol{\beta}}}(\boldsymbol{x})^{\mathsf{T}}\boldsymbol{K}_{\hat{\boldsymbol{\theta}}}^{-1}\left(\mathcal{Y} - \boldsymbol{R}\widehat{\boldsymbol{\beta}}\right) \tag{2.57}$$

where $\hat{\boldsymbol{\theta}}$ denotes the set of autocorrelation hyperparameters computed by ML. $\widehat{\mathcal{M}}(\boldsymbol{x})$ may be used as a metamodel. It is shown that $\widehat{\mathcal{M}}(\boldsymbol{x})$ interpolates the observations in $\mathcal{Y}$. Moreover, one gets from Eq.(2.51) the following variance estimate of the metamodel:

$$\widehat{\sigma}^2_{\mathsf{M}}(\boldsymbol{x}) = \widehat{\sigma}^2 - \boldsymbol{k}_{\hat{\boldsymbol{\beta}}}(\boldsymbol{x})^{\mathsf{T}}\,\boldsymbol{K}_{\hat{\boldsymbol{\theta}}}^{-1}\,\boldsymbol{k}_{\hat{\boldsymbol{\beta}}}^{\mathsf{T}}(\boldsymbol{x}) \tag{2.58}$$

which may be used to derive confidence intervals on the predicted values $\widehat{\mathcal{M}}(\boldsymbol{x})$. The GP approach is illustrated in Figure 2.3 by the interpolation of the so-called Runge function defined by $f : x \mapsto y = (1+25x^2)^{-1}$. The Matlab toolbox *DACE* (for Design and Analysis of Computer Experiments) (Lophaven et al., 2002), which is dedicated to kriging metamodelling, has been used to this end.

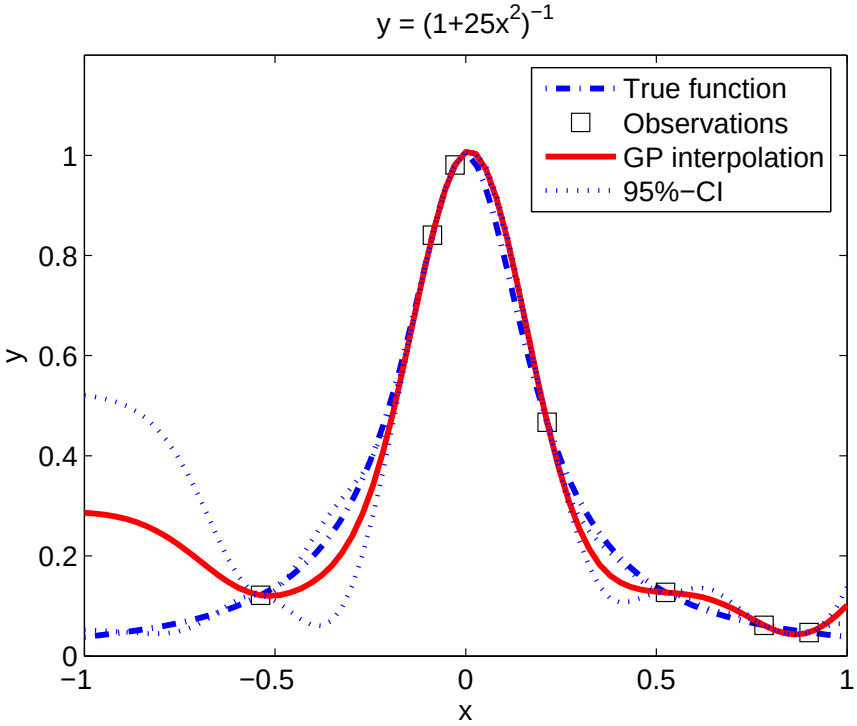


Figure 2.3: *Example of one-dimensional interpolation of the Runge function by GP modelling together with confidence intervals*

As an alternative, the GP metamodel may be obtained by seeking the *best linear unbiased predictor* (BLUP) in the form:

$$\widehat{\mathsf{M}}(\boldsymbol{x},\omega) = \sum_{i=1}^{N} \alpha_i(\boldsymbol{x})\ \mathsf{M}(\boldsymbol{x}^{(i)},\omega) \tag{2.59}$$

The BLUP minimizes the mean-square error $\mathbb{E}\left[(\widehat{\mathsf{M}}(\boldsymbol{x},\omega)-\mathsf{M}(\boldsymbol{x},\omega))^2\right]$ for all $\boldsymbol{x}$ under the unbiasedness condition $\mathbb{E}\left[\widehat{\mathsf{M}}(\boldsymbol{x},\omega)-\mathsf{M}(\boldsymbol{x},\omega)\right]=0$. The mean and the variance of the solution are the same as in Eqs.(2.57),(2.58). This corresponds to the original *kriging* point of view (Sacks et al., 1989; Vazquez, 2005).

Once the GP metamodel has been determined, the second moments and the distribution of the random model response may be computed inexpensively by crude Monte Carlo simulation. Note that analytical formulæ of the Sobol' indices have been derived in Oakley and O'Hagan (2004); Marrel (2008) together with confidence intervals.

3.2.5 Conclusion

The Gaussian Process (GP) modelling method has been described. It relies upon the assumption that the model response is a sample path of an underlying Gaussian random field. The properties of the latter may be estimated using the maximum likelihood approach. Then a metamodel may be derived that interpolates the observed model evaluations. This makes GP an attractive technique since there is no random error when considering a deterministic model function. Moreover, the stochastic framework of GP provides confidence intervals on the predictions, allowing an estimation of the approximation error. However the derivations in GP rely upon the knowledge of the parameters that characterize the underlying random field. In practice these parameters are estimated by maximum likelihood, which may reveal computationally expensive if the number of unknown parameters or the number of input variables is large.

3.3 Support Vector Regression

Support Vector Regression is a specific regression technique proposed in Vapnik et al. (1997); Smola and Schölkopf (2006) in the field of statistical learning. As in the previous sections one considers an experimental design $\mathcal{X}\equiv\{\boldsymbol{x}^{(1)},\ldots,\boldsymbol{x}^{(N)}\}^{\mathsf{T}}$ of the input random vector $\boldsymbol{x}$ and the corresponding model evaluations $\mathcal{Y}\equiv\{\mathcal{M}(\boldsymbol{x}^{(1)}),\ldots,\mathcal{M}(\boldsymbol{x}^{(N)})\}^{\mathsf{T}}=\{y^{(1)},\ldots,y^{(N)}\}^{\mathsf{T}}$.

3.3.1 Linear case

SVR formulation

Let us consider the following linear approximation of the model response $\mathcal{M}(\boldsymbol{x})$:

$$\widehat{\mathcal{M}}(\boldsymbol{x}) \;\equiv\; \sum_{i=1}^{M} a_i\, x_i \;+\; b \;\equiv\; \boldsymbol{a}^{\mathsf{T}}\,\boldsymbol{x} \;+\; b \tag{2.60}$$

where $\boldsymbol{a} \equiv \{a_1, \ldots, a_M\}^\mathsf{T}$. Parameters b and the a_i's are unknown deterministic coefficients. Regression methods yield a solution $\widehat{\mathcal{M}}$ that minimizes some *loss function* $\ell(\mathcal{M}, \widehat{\mathcal{M}}, \mathcal{X})$. Ordinary least-square regression consists in minimizing the following *quadratic* loss function:

$$\ell(\mathcal{M}, \widehat{\mathcal{M}}, \mathcal{X}) \;\equiv\; \sum_{i=1}^{N} \Big(\mathcal{M}(\boldsymbol{x}^{(i)}) \;-\; \widehat{\mathcal{M}}(\boldsymbol{x}^{(i)})) \Big)^2 \tag{2.61}$$

However, a too large number of unkown parameters compared to the number of observations may lead to an ill-posedness of the least-square regression problem and numerical unstabilities. This drawback may be circumvented by including a *regularization* term. Upon gathering the parameters a_i into vector $\boldsymbol{a}$, the regularized problem may be for instance cast as follows:

$$\text{Minimize} \quad J(b, \boldsymbol{a}) \;\equiv\; \|\boldsymbol{a}\|_2^2 \,+\, C \sum_{i=1}^{N} \Big(\mathcal{M}(\boldsymbol{x}^{(i)}) \;-\; \boldsymbol{a}^\mathsf{T} \boldsymbol{x}^{(i)} \;-\; b \Big)^2 \tag{2.62}$$

where $\|\boldsymbol{a}\|_2^2 \equiv \sum a_i^2$. C is a positive constant which controls the trade-off between regularization and data fitting.

In the context of physical experiments, the experiments are never perfectly reproducible, hence the presence of *random* residuals $\epsilon_i(\omega) \equiv \mathcal{M}(\boldsymbol{x}^{(i)}) - \widehat{\mathcal{M}}(\boldsymbol{x}^{(i)})$. The quadratic loss function in Eq.(2.61) is known to be optimal in case of Gaussian residuals (see *e.g.* Vazquez (2005, Section 3.6)). However it may lead to unsatisfactory results if the distribution of the residuals is non Gaussian. A solution to this problem is to select a loss function that is less sensitive to the extreme values of the residuals than the quadratic one. The originality of SVR lies in the choice of the so-called ε-*insensitive* loss function $[\cdot]_\varepsilon$ defined by:

$$[s]_\varepsilon \;\equiv\; \max(0, |s| - \varepsilon) \tag{2.63}$$

Thus SVR leads to the following regularized regression problem:

$$\text{Minimize} \quad J(b, \boldsymbol{a}) \;\equiv\; \|\boldsymbol{a}\|_2^2 \,+\, C \sum_{i=1}^{N} \Big[\mathcal{M}(\boldsymbol{x}^{(i)}) \;-\; \boldsymbol{a}^\mathsf{T} \boldsymbol{x}^{(i)} \;-\; b \Big]_\varepsilon \tag{2.64}$$

Such a formulation corresponds to seeking the flattest solution while tolerating the deviations not greater than ε and penalizing the others, as illustrated in Figure 2.4.

Solving of the SVR problem

Upon introducing auxiliary variables $\{(\zeta_i, \zeta_i^*), i = 1, \ldots, N\}$, called *slack variables*, the problem in Eq.(2.64) may be recast as:

$$\begin{aligned} &\underset{\boldsymbol{a}, b, \zeta_i, \zeta_i^*}{\text{Minimize}} \quad \frac{1}{2}\|\boldsymbol{a}\|_2^2 \,+\, C \sum_{i=1}^{N} (\zeta_i \,+\, \zeta_i^*) \\ &\text{subject to} \quad \begin{cases} \mathcal{M}(\boldsymbol{x}^{(i)}) \;-\; \boldsymbol{a}^\mathsf{T}\, \boldsymbol{x}^{(i)} \;-\; b \;\leq\; \varepsilon \,+\, \zeta_i \\ \boldsymbol{a}^\mathsf{T}\, \boldsymbol{x}^{(i)} \,+\, b \,-\, \mathcal{M}(\boldsymbol{x}^{(i)}) \;\leq\; \varepsilon \,+\, \zeta_i^* \qquad i = 1, \ldots, N \\ \zeta_i \,,\, \zeta_i^* \;\geq\; 0 \end{cases} \end{aligned} \tag{2.65}$$

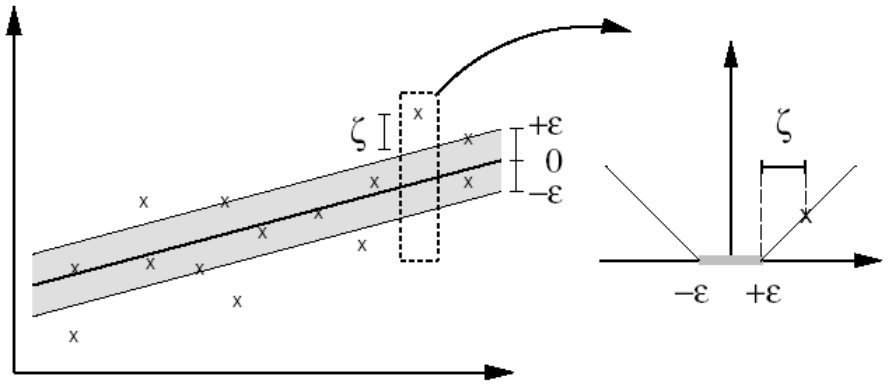


Figure 2.4: *Support vector regression in the linear case (after Smola and Schölkopf (2006))*

The introduction of the slack variables may be interpreted as follows. Requiring that the residuals do not exceed ε (*i.e.* $\zeta_i, \zeta_i^* = 0$) leads to an optimization problem which may not admit any solution. Thus deviations larger than ε are tolerated, (the two first constraints in Eq.(2.65)), but have to be minimized (the regularizing term in Eq.(2.65)).

The Lagrangian of the optimization problem in Eq.(2.65)) reads:

$$\begin{aligned} L(b, \boldsymbol{a}, \zeta_i, \zeta_i^*, \alpha_i, \alpha_i^*, \nu_i, \nu_i^*) \;\; = \;\; & \frac{1}{2}\|\boldsymbol{a}\|_2^2 + C \sum_{i=1}^{N} (\zeta_i + \zeta_i^*) \\ & - \sum_{i=1}^{N} \alpha_i (\varepsilon + \zeta_i - y^{(i)} + \boldsymbol{a}^{\mathsf{T}} \boldsymbol{x}^{(i)} + b) \\ & - \sum_{i=1}^{N} \alpha_i^* (\varepsilon + \zeta_i^* - y^{(i)} - \boldsymbol{a}^{\mathsf{T}} \boldsymbol{x}^{(i)} - b) \\ & - \sum_{i=1}^{N} (\nu_i \zeta_i + \nu_i^* \zeta_i^*) \end{aligned} \tag{2.66}$$

with $\alpha_i, \alpha_i^*, \nu_i, \nu_i^* \geq 0$.

Requiring that the partial derivatives of L with respect to the variables $b, \boldsymbol{a}, \zeta_i, \zeta_i^*$ be zero at the optimum, one obtains the following SVR metamodel:

$$\widehat{\mathcal{M}}(\boldsymbol{x}) \;\; = \;\; \sum_{i=1}^{N} \; (\widehat{\alpha}_i - \widehat{\alpha}_i^*) \; {\boldsymbol{x}^{(i)}}^{\mathsf{T}} \; \boldsymbol{x} + \widehat{b} \tag{2.67}$$

where $\widehat{\alpha}_i, \widehat{\alpha}_i^*$ are the solutions of the following quadratic optimization problem:

$$\begin{array}{ll} \underset{\alpha_i,\alpha_i^*}{\text{Maximize}} & -\frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*){\boldsymbol{x}^{(i)}}^{\mathsf{T}}\boldsymbol{x}^{(j)} \;-\; \varepsilon\sum_{i=1}^{N}(\alpha_i+\alpha_i^*) \;+\; \sum_{i=1}^{N}y^{(i)}(\alpha_i-\alpha_i^*) \\ \text{subject to} & \left\{\begin{array}{ll} \sum_{i=1}^{N}(\alpha_i-\alpha_i^*) \;=\; 0 & \\ 0 \;\leq\; \alpha_i,\alpha_i^* \;\leq\; C & \end{array}\right. \qquad i=1,\ldots,N \end{array} \tag{2.68}$$

An important feature of the solution of this optimization problem is that the quantities $(\widehat{\alpha}_i - \widehat{\alpha}_i^*)$ are zero for some i's. The observations for which to $(\widehat{\alpha}_i - \widehat{\alpha}_i^*) \neq 0$ are called *support vectors*. This property is obtained by deriving the Karush-Kuhn-Tucker conditions, which state that the products between the dual variables and the constraints are equal to zero at the optimum, that is:

$$\begin{array}{l} \widehat{\alpha}_i\left(\varepsilon \;+\; \widehat{\zeta}_i \;-\; y^{(i)} \;+\; \widehat{\mathcal{M}}(\boldsymbol{x}^{(i)})\right) \;=\; 0 \\ \widehat{\alpha}_i^*\left(\varepsilon \;+\; \widehat{\zeta}_i \;-\; y^{(i)} \;+\; \widehat{\mathcal{M}}(\boldsymbol{x}^{(i)})\right) \;=\; 0 \end{array} \tag{2.69}$$

As explained previously and as depicted in Figure 2.4, only those integers i associated with the data for which $|\mathcal{M}(\boldsymbol{x}^{(i)}) - \widehat{\mathcal{M}}(\boldsymbol{x}^{(i)})| \geq \varepsilon$ are such that ζ_i and ζ_i^* are non zero. Hence the data lying inside a tube of width ε around $\widehat{\mathcal{M}}(\boldsymbol{x})$ correspond to $\alpha_i = \alpha_i^* = 0$.

3.3.2 Extension to the non linear case

The SVR method can be extended in order to take into account a non linear relationship between $\mathcal{X}$ and $\mathcal{Y}$. The first step consists in defining a mapping Φ from the space of input parameters $\boldsymbol{x} \equiv \{x_1,\ldots,x_M\}$ to a high-dimensional space $\mathcal{F}$ called *feature space*. Then the SVR scheme described in the previous section will be applied to the new data $\{(\Phi(\boldsymbol{x}^{(i)}), y^{(i)}) : i = 1,\ldots,N\}$. Indeed the relationship between $\{\Phi(\boldsymbol{x}^{(i)}) : i = 1,\ldots,N\}$ and $\mathcal{Y}$ is expected to be close to linearity. Note that polynomial regression corresponds to $\mathcal{F} \equiv \mathcal{P}_p$, where $\mathcal{P}_p$ is the space of multivariate polynomials with total degree not greater than p. In this case the mapping Φ may be clearly identified *a priori*.

In contrast, SVR does not require an explicit knowledge of the function Φ. It is sufficient to know only an inner product over $\mathcal{F}$ of the form:

$$k(\boldsymbol{x},\boldsymbol{x}') \;\equiv\; \langle\Phi(\boldsymbol{x}),\Phi(\boldsymbol{x}')\rangle_{\mathcal{F}} \tag{2.70}$$

In this setting, the SVR dual optimization problem in Eq.(2.68) reads:

$$\begin{array}{ll} \underset{\alpha_i,\alpha_i^*}{\text{Maximize}} & -\frac{1}{2}\sum_{j,k=1}^{N}(\alpha_j-\alpha_j^*)(\alpha_k-\alpha_k^*)k(\boldsymbol{x}^{(j)},\boldsymbol{x}^{(k)}) \;-\; \varepsilon\sum_{j=1}^{N}(\alpha_j+\alpha_j^*) \;+\; \sum_{j=1}^{N}y^{(j)}(\alpha_j-\alpha_j^*) \\ \text{subject to} & \left\{\begin{array}{ll} \sum_{j=1}^{N}(\alpha_j-\alpha_j^*) \;=\; 0 & \\ 0 \;\leq\; \alpha_i,\alpha_i^* \;\leq\; C & \end{array}\right. \qquad i=1,\ldots,N \end{array} \tag{2.71}$$

and the solution is given by:

$$\widehat{\mathcal{M}}(\boldsymbol{x}) \;=\; \sum_{i=1}^{N}\;(\widehat{\alpha}_i-\widehat{\alpha}_i^*)\;k(\boldsymbol{x}^{(i)},\boldsymbol{x}) \;+\; \widehat{b} \tag{2.72}$$

The inner product $k(\cdot,\cdot)$ is called *kernel*. Many kinds of kernels may be selected in practice. In particular, it has been shown in Vazquez (2005) that covariance functions may be chosen (see for example the autocorrelation functions in Section 3.2.2)[3].

3.3.3 Illustration on the Runge function

The SVR method is applied to the interpolation of the Runge function defined by $f : x \mapsto y = (1+25x^2)^{-1}$. In this purpose, the so-called *Matlab Support Vector Machines Toolbox* (Gunn, 1998) freely available at www.isis.ecs.soton.ac.uk/resources/svminfo is used. The following Gaussian kernel function is selected:

$$k(x,x') \;\equiv\; \exp\left(-\frac{(x-x')^2}{\sigma^2}\right) \qquad , \qquad \sigma = 0.1 \tag{2.73}$$

Two parametric studies are carried out varying the parameters C and ε, respectively.

Figure 2.5 shows that the SVR metamodel tends to interpolate the observations when ε tends to zero. Moreover, the number of support vectors decrease when increasing ε. In particular, the SVR approximation is only affected by the observation at $x = 0$ when setting ε equal to 0.5. On the other hand, Figure 2.6 shows that a low value of the tuning parameter C yields a "flat" approximation which poorly fits the data. In contrast, a high value of C leads to a decrease of the metamodel smoothness and a better fit of the data.

3.3.4 Discussion

SVR is a metamodelling technique which provides the following advantages:

[3]Indeed, the Mercer theorem states that any continuous symmetric non-negative definite kernel $k(\boldsymbol{x},\boldsymbol{x}')$ may be expanded as $k(\boldsymbol{x},\boldsymbol{x}') = \sum_{i=1}^{\infty}\lambda_i e_i(\boldsymbol{x})e_i(\boldsymbol{x}')$ where the convergence is absolute and uniform. Using the notation $\boldsymbol{\Phi}(\boldsymbol{x}) \equiv \{\sqrt{\lambda_i}e_i(\boldsymbol{x}), i=1,2,\ldots\}^{\mathsf{T}}$, one gets the inner product representation $k(\boldsymbol{x},\boldsymbol{x}') = \langle\boldsymbol{\Phi}(\boldsymbol{x}),\boldsymbol{\Phi}(\boldsymbol{x}')\rangle$.

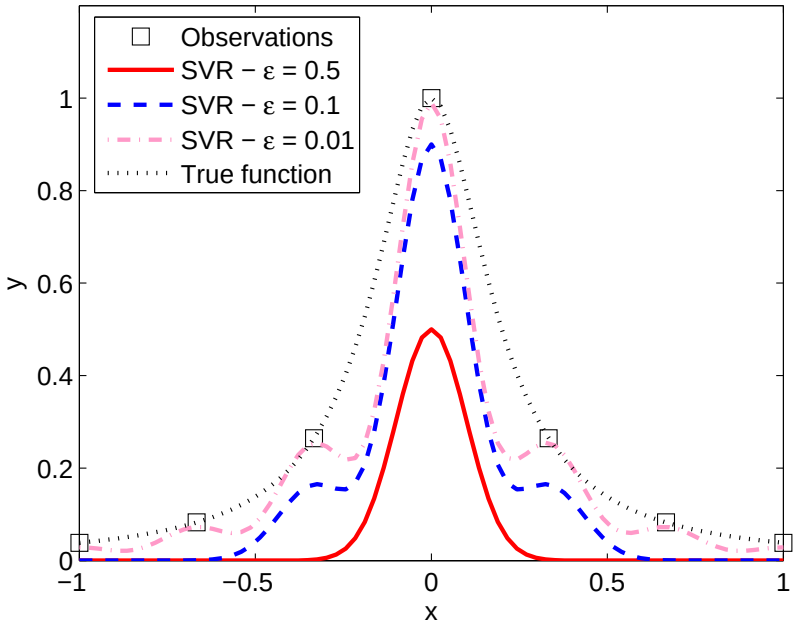


Figure 2.5: *Example of one-dimensional interpolation of the Runge function* $f : x \mapsto y = (1 + 25x^2)^{-1}$ *by SVR - Parametric study varying* ε *with* $C = 1$

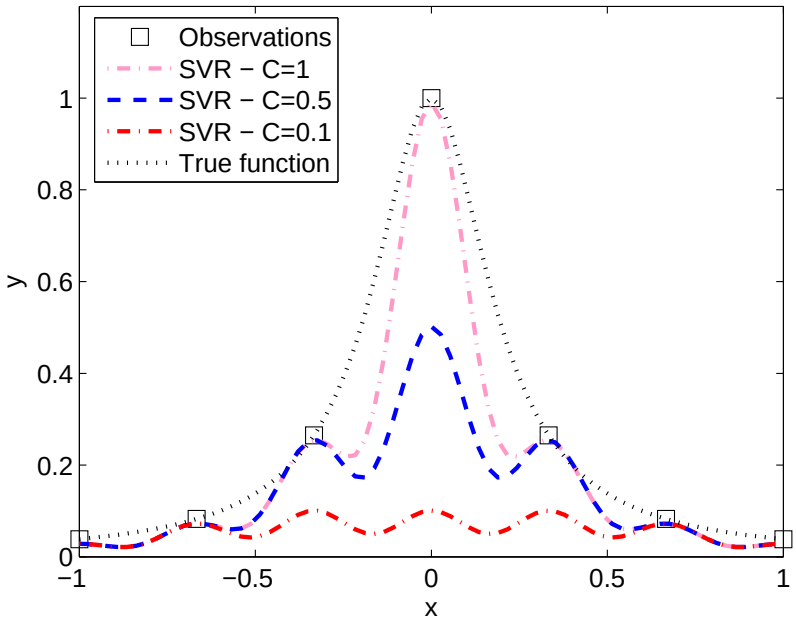


Figure 2.6: *Example of one-dimensional interpolation of the Runge function* $f : x \mapsto y = (1 + 25x^2)^{-1}$ *by SVR - Parametric study varying* C *with* $\varepsilon = 0.01$

- a numerical solving scheme of the SVR optimization problem may be easily implemented since it is a quadratic problem, see Eq.(2.71);

- SVR provides a *sparse* representation of the data, *i.e.* an approximation which depends on a small number of observations;
- the SVR approximation is robust with regard to outliers.

Concerning the first point, it has to be noted that the optimization problem to be solved corresponds to a particular choice of the parameters ε, C as well as the possible parameters of the selected kernel function. Selecting optimal values of the tuning parameters is not an easy task though. This may be done using a cross validation scheme, but this would require a large number of SVR calculations on a fine grid of tuning parameters.

The second point may be particularly interesting if a large number of model evaluations is allowed. Indeed, the sparsity of the SVR metamodel is defined in terms of observations (the approximation depends only on a small number of support vectors). However, the induced reduction of complexity may be insignificant if a small sample set is used, which is consistent with our objective of minimizing the number of model runs.

Lastly, the third point seems to be relevant in an experimental context, in which the observations may be affected by a large and non Gaussian random error. In such a setting, ordinary least-square regression would be unstable whereas SVR would provide an approximation that is robust to outliers. Nevertheless, this notion of outliers does not really make sense in the context of deterministic computer experiments. In other words, there is no reason to exclude a given observation (provided that the model has been correctly designed, so that any evaluation of the model at an irrealistic set of parameters is prohibited).

3.4 Use of metamodels for uncertainty propagation

Metamodelling techniques consist in building an analytical approximation $\widehat{\mathcal{M}}$ of the deterministic model function $\boldsymbol{x} \mapsto \mathcal{M}(\boldsymbol{x})$. Classical experimental designs such as Monte Carlo Sampling (MCS) or Latin Hypercube Sampling (LHS) are often used to this purpose (Kleijnen, 2004). These random designs are generally *uniformly* generated over the domain of variation of the input parameters.

Such an approach does not take into account a statistical distribution of the input random variables, which is considered though in the framework of uncertainty propagation. Indeed, it is recalled that the input parameters are modelled by random variables $\boldsymbol{X} \equiv \{X_1, \ldots, X_M\}$ in such a setting. The distribution, the second moments and the sensitivity indices of the model output are generally computed by direct Monte Carlo simulation of the metamodel (Mavris and Bandte, 1997; Sathyanarayanamurthy and Chinnam, 2009). These calculations are inexpensive since the metamodel is analytical. The Monte Carlo sampling set is randomly drawn from the joint probability density function (PDF) of the input parameters, and *not* uniform as the sample

that has been used to build up the metamodel. It is worth mentioning that analytical expressions of the response second-moments and Sobol' indices have been derived for the Gaussian process metamodel in Oakley and O'Hagan (2004); Marrel (2008).

The metamodelling approach described herein, which is aimed at obtaining an accurate approximation of the model function $\mathcal{M}$ *at any point* of the domain of variation of the input parameters, may excessively focus on subdomains associated with a low probability as depicted in Figure 2.7.

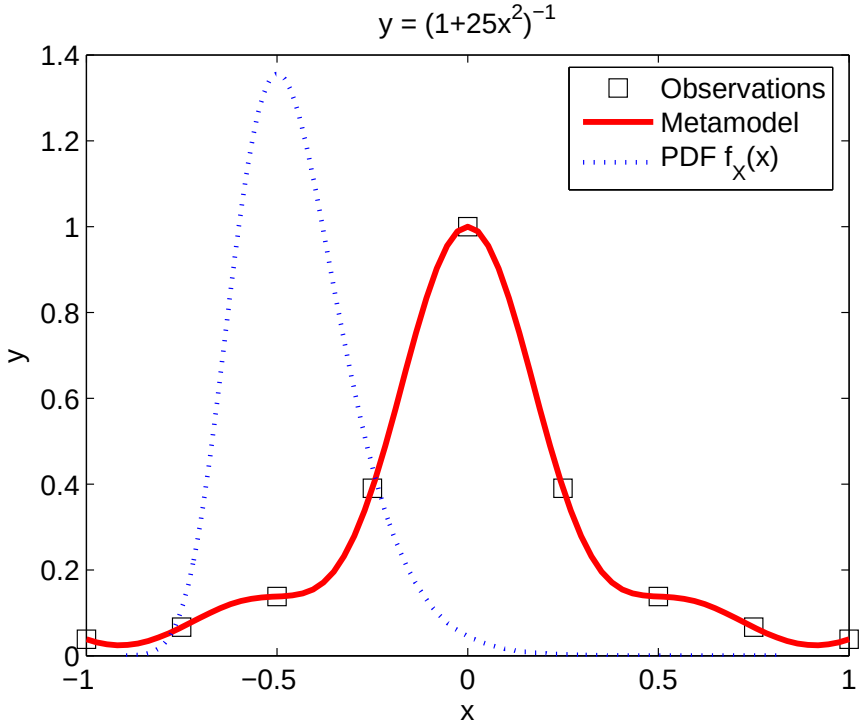


Figure 2.7: *Dilemma between function approximation and uncertainty analysis*

As an alternative, one may not approximate the sole function $\mathcal{M}$, but rather the random response $Y \equiv \mathcal{M}(\boldsymbol{X})$ of the model. This may be achieved using an appropriate *stochastic* metamodel $\widehat{\mathcal{M}}(\boldsymbol{X})$. The so-called *polynomial chaos* (PC) *expansions* (Wiener, 1938; Soize and Ghanem, 2004) seem well suited to this purpose. Indeed, they provide an explicit representation in terms of the input random variables $\{X_1, \ldots, X_M\}$. Furthermore, the second moments and the sensitivity indices of the random response Y may be obtained as simple byproducts of the PC metamodel.

4 Conclusion

Various well-known methods have been reviewed in this chapter for uncertainty propagation and sensitivity analysis. They are based on intensive Monte Carlo simulation, which may lead to intractable calculations in case of a computationally demanding model $\mathcal{M}$. To overcome this

difficulty, it is possible to substitute the model response by an analytical approximation, called metamodel.

Two recent metamodelling techniques have been briefly described, namely Gaussian Process (GP) modelling and Support Vector Regression (SVR). GP provides a nice statistical framework that allows the derivations of confidence intervals on the predictions. However the parameters of the GP metamodel have to be estimated accurately. This requires solving an optimization problem whose size may blow up in case of a large number of input parameters. On the other hand, SVR has received great interest in the statistical learning community. An attractive feature of the method is that it yields a metamodel which is robust and sparse in terms of the observations. However it involves tuning parameters whose optimization is not straightforward.

As many metamodelling techniques, GP and SVR are aimed at approximating the model response at any point in the domain of variation of the input parameters. This approach may not be efficient in the context of uncertainty propagation since all the points are not associated with the same probability. In other words, one may focus too much on the exploration of regions which are highly unlikely. To bypass this problem, it is possible to build up a *stochastic* metamodel of the model response regarded as a random vector. The so-called polynomial chaos expansions are well suited to this end and will be investigated in the next chapters.

Chapter 3

Polynomial chaos representations for uncertainty propagation

1 Introduction

Spectral expansions are well known in numerical analysis for solving various problems, such as differential and integral equations (Boyd, 1989). In a probabilistic context, the approach relies upon the approximation of the unknown random response of a model in a suitable finite-dimensional basis $(\psi_j(\boldsymbol{X}))_{0\leq j\leq P-1}$ as follows:

$$\mathcal{M}(\boldsymbol{X}) \approx \sum_{j=0}^{P-1} a_j \, \psi_j(\boldsymbol{X}) \tag{3.1}$$

In the present work, spectral expansions onto bases made of orthogonal polynomials, commonly referred to as *polynomial chaos* (PC) *expansions*, are of interest. In this setup, characterizing the model response is equivalent to computing the deterministic coefficients a_j's. To this end, the expansion may be typically inserted into the governing equations (*e.g.* a system of partial differential equations), and the coefficients may be obtained using a Galerkin scheme. Such an approach has been applied in the early 1990's to structural mechanics problems featuring spatially random parameters (Ghanem and Spanos, 1991). The approximation in Eq.(3.1), which is regarded as a discretization in the stochastic dimension, is then combined to a spatial finite element discretization, hence the name *spectral stochastic finite element method* (SSFEM). Although this approach is based on a sound mathematical foundation, it may lead to solve a huge matrix system, hence a high computational cost in time and memory. This problem has been adressed in Nouy (2007a) where an iterative building of a spectral decomposition is proposed. The approximation, called *generalized spectral decomposition* (GSD), is sought onto an optimal reduced basis, *i.e.* a basis which captures the main features of the response by means of a low

number of terms. The method only leads to solve a series of low-dimensional problems, hence a dramatical computational gain compared to the usual Galerkin scheme.

The abovementioned Galerkin-based methods are often labelled *intrusive* for they require a specific modification of the already existing deterministic computer code, depending on the nature of the problem (*e.g.* linear or nonlinear, elliptic or parabolic). Alternatively, the so-called *non intrusive* methods have emerged recently in stochastic finite element analysis. In this setup, the governing equations of the model under consideration are regarded as a black-box function, *i.e.* a function which can be only known through model evaluations. Non intrusive methods aim at obtaining a PC approximation such as in Eq.(3.1) by means of a series of well chosen calls to the deterministic model.

First of all, the mathematical framework of the PC representation is presented in Section 2 using the formalism proposed in Soize and Ghanem (2004). Then the intrusive SSFEM and GSD methods are described in Section 3. Lastly, the non intrusive methods are investigated in detail in Section 4.

2 Spectral representation of functionals of random vectors

2.1 Introduction

Let us consider a physical model represented by a deterministic mapping $\boldsymbol{y} = \mathcal{M}(\boldsymbol{x})$. Here $\boldsymbol{x} = \{x_1, ..., x_M\}^\mathsf{T} \in \mathbb{R}^M$, $M \geq 1$ is the vector of the input variables, and $\boldsymbol{y} = \{y_1, ..., y_Q\}^\mathsf{T} \in \mathbb{R}^Q$, $Q \geq 1$ is the vector of quantities of interest provided by the model, referred to as the *model response* in the sequel. As the input vector $\boldsymbol{x}$ is assumed to be affected by uncertainty, a probabilistic framework is now introduced.

Let $(\Omega, \mathcal{F}, \mathcal{P})$ be a probability space, where Ω is the event space equipped with σ-algebra $\mathcal{F}$ and probability measure $\mathcal{P}$. Random variables are denoted by upper case letters $X(\omega) : \Omega \to \mathcal{D}_\mathcal{X} \subset \mathbb{R}$, while their realizations are denoted by the corresponding lower case letters, *e.g.* x. Moreover, bold upper and lower case letters are used to denote random vectors (*e.g.* $\boldsymbol{X} = \{X_1, \ldots, X_M\}^\mathsf{T}$) and their realizations (*e.g.* $\boldsymbol{x} = \{x_1, \ldots, x_M\}^\mathsf{T}$), respectively.

Let us denote by $\mathcal{L}^2_\mathbb{R} \equiv \mathcal{L}^2(\Omega, \mathcal{F}, \mathcal{P}; \mathbb{R})$ the space of real random variables X with finite second moments:

$$\mathbb{E}\left[X^2\right] = \int_\Omega X^2(\omega) d\mathcal{P}(\omega) = \int_{\mathcal{D}_X} x^2 f_X(x) dx < +\infty \tag{3.2}$$

where $\mathbb{E}[\cdot]$ denotes the mathematical expectation operator and f_X (resp. $\mathcal{D}_X$) represents the probability density function (PDF) (resp. the support) of X. This space is an Hilbert space

with respect to the inner product:

$$< X_1, X_2 >_{\mathcal{L}^2_{\mathbb{R}}} \equiv \mathbb{E}\left[X_1 X_2\right] = \int_\Omega X_1(\omega) X_2(\omega) d\mathcal{P}(\omega) = \int_{\mathcal{D}_{\boldsymbol{X}}} x_1 x_2 \, f_{X_1,X_2}(x_1, x_2) \, dx_1 dx_2 \quad (3.3)$$

where f_{X_1,X_2} is the *joint* PDF of the random vector $\{X_1, X_2\}^\mathsf{T}$. This inner product induces the norm $\|X\|_{\mathcal{L}^2_{\mathbb{R}}} \equiv \sqrt{\mathbb{E}\left[X^2\right]}$.

The input vector of the physical model $\mathcal{M}$ is represented as a random vector $\boldsymbol{X}(\omega), \omega \in \Omega$ with prescribed joint PDF $f_{\boldsymbol{X}}$. The model response is also a random variable $Y(\omega) = \mathcal{M}(\boldsymbol{X}(\omega))$, which is assumed to be scalar for the sake of simplicity in this presentation, *i.e.* $Q = 1$. Note that in case of a vector-valued model response $\boldsymbol{Y}$, the following derivations hold componentwise. Moreover, It is assumed that the response is a square-integrable random variable, *i.e.* $Y \in \mathcal{L}^2_{\mathbb{R}}$.

Let us denote by $\mathcal{L}^2$ the space functionals of M-dimensional random vectors that are square-integrable with respect to the joint density $f_{\boldsymbol{X}}(\boldsymbol{x})$. Of interest is the spectral decomposition of Y onto a suitable complete orthogonal basis of $\mathcal{L}^2$ (orthogonality is defined with respect to the joint probability density function $f_{\boldsymbol{X}}$ of the input random vector $\boldsymbol{X}$). In the following we consider bases made of orthogonal polynomials. Note that other kinds of basis functions may be considered though, such as finite elements (Deb et al., 2001; Babuška et al., 2005; Frauenfelder et al., 2005), wavelets (Antoniadis, 1997; Antoniadis et al., 2001; Le Maître et al., 2004a,b) or multi-element bases (Wan and Karniadakis, 2005).

The simple case of independent random variables is addressed first. Then the case of random variables correlated with a Gaussian dependence structure (*Nataf distribution*) is considered. Lastly the case of input random fields is tackled.

2.2 Independent random variables

It is first assumed that the components of the input random vector are independent. It is shown that Y may be expanded onto an orthogonal polynomial basis as follows (Soize and Ghanem, 2004):

$$Y \equiv \mathcal{M}(\boldsymbol{X}) = \sum_{j=0}^{+\infty} a_j \psi_j(\boldsymbol{X}) \quad (3.4)$$

where the series converges in the sense of the $\mathcal{L}^2$-norm, that is:

$$\lim_{P \to +\infty} \left\| \mathcal{M}(\boldsymbol{X}) - \sum_{j=0}^{P-1} a_j \psi_j(\boldsymbol{X}) \right\|^2_{\mathcal{L}^2} \equiv \lim_{P \to +\infty} \mathbb{E}\left[\left(\mathcal{M}(\boldsymbol{X}) - \sum_{j=0}^{P-1} a_j \psi_j(\boldsymbol{X}) \right)^2 \right] = 0 \quad (3.5)$$

The $a_{\boldsymbol{\alpha}}$'s are unknown deterministic coefficients, and the $\psi_{\boldsymbol{\alpha}}$'s are multivariate polynomials. The series in Eq.(3.4) is usually referred to as *polynomial chaos* (PC) *expansion*. The principles of the building of the PC basis are described below.

Let us first notice that due to the independence of the input random variables, the input joint PDF may be cast as:

$$f_{\boldsymbol{X}}(\boldsymbol{x}) \;=\; \prod_{i=1}^{M} f_{X_i}(x_i) \tag{3.6}$$

where $f_{X_i}(x_i)$ is the marginal PDF of X_i. Let us consider a family $\{\pi_j^{(i)}, j \in \mathbb{N}\}$ of orthonormal polynomials with respect to f_{X_i}, *i.e.* :

$$\langle \pi_j^{(i)}(X_i)\ ,\ \pi_k^{(i)}(X_i) \rangle_{\mathcal{L}^2_{\mathbb{R}}} \;\equiv\; \mathbb{E}\left[\pi_j^{(i)}(X_i)\ \pi_k^{(i)}(X_i)\right] \;=\; \delta_{j,k} \tag{3.7}$$

It is assumed that the degree of $\pi_j^{(i)}$ is j for $j > 0$ and $\pi_0^{(i)} \equiv 1$ $(1 \leq i \leq M)$.

Upon tensorizing the M resulting families of univariate polynomials, one gets a set of orthonormal multivariate polynomials $\{\psi_{\boldsymbol{\alpha}}, \boldsymbol{\alpha} \in \mathbb{N}^M\}$ defined by:

$$\psi_{\boldsymbol{\alpha}}(\boldsymbol{x}) \;\equiv\; \pi_{\alpha_1}^{(1)}(x_1) \times \cdots \times \pi_{\alpha_M}^{(M)}(x_M) \tag{3.8}$$

where the multi-index notation $\boldsymbol{\alpha} \equiv \{\alpha_1, \ldots, \alpha_M\}$ has been introduced. The PC expansion was originally formulated with standard Gaussian random variables and Hermite polynomials as the *finite-dimensional Wiener polynomial chaos* (Wiener, 1938; Ghanem and Spanos, 2003). It was later extended to other classical random variables together with basis functions from the Askey family of hypergeometric polynomials (Xiu and Karniadakis, 2002; Xiu et al., 2002, 2003; Lucor and Karniadakis, 2004). The decomposition is then referred to as *generalized* PC expansion. In this setup, most common continuous distributions can be associated to a specific family of polynomials (Schoutens, 2000; Xiu and Karniadakis, 2002), as reported in Table 3.1.

Table 3.1: *Correspondence between usual continuous distributions and families of orthogonal polynomials*

Distribution	Support	Polynomial
Gaussian	$\mathbb{R}$	Hermite
Uniform	$[-1, 1]$	Legendre
Gamma	$(0, +\infty)$	Laguerre
Chebyshev	$(-1, 1)$	Chebychev
Beta	$(-1, 1)$	Jacobi

If other distribution types appear, then it is possible to employ a nonlinear mapping (namely an *isoprobabilistic transform*) such that the generalized PC expansion can be applied to the new variable. For instance, a lognormal variable will be recast as a function of a standard normal variable, which will be used in conjonction with Hermite polynomials. As an alternative, *ad hoc* orthogonal polynomial may be generated numerically for random variables with arbitrary distributions (Wan and Karniadakis, 2006; Witteveen et al., 2007).

2.3 Case of an input Nataf distribution

We now consider the case of input random variables that are correlated by means of a *Nataf distribution* (Nataf, 1962), *i.e.* whose joint cumulative density function (CDF) is given by:

$$F_{\boldsymbol{X}}(x_1, ..., x_M) = \Phi_{M,\boldsymbol{R}}\left[\Phi^{-1}(F_1(x_1)), ..., \Phi^{-1}(F_M(x_M))\right] \tag{3.9}$$

where $F_i(x_i)$ is the marginal CDF of the random variable X_i, $\Phi_{M,\boldsymbol{R}}$ is the standard Gaussian CDF of dimension M and correlation coefficient matrix $\boldsymbol{R}$, and Φ is the unidimensional standard Gaussian CDF. Let us denote by $\widehat{\boldsymbol{\xi}} \equiv \{\widehat{\xi_i} \equiv \Phi^{-1}(F_i(X_i)), i = 1, ..., M\}$ the *correlated* standard Gaussian random variables which appear in Eq.(3.9). In order to derive a classical Hermite PC approximation the model response shall be recast as a function of *independent* standard Gaussian random variables ξ_i. In this respect, $\widehat{\boldsymbol{\xi}}$ is expressed as a function of a standard Gaussian random vector $\boldsymbol{\xi}$ with uncorrelated components:

$$\widehat{\boldsymbol{\xi}} = \boldsymbol{\Gamma}\,\boldsymbol{\xi} \tag{3.10}$$

where the matrix $\boldsymbol{\Gamma}$ is obtained by the Cholesky decomposition of $\boldsymbol{R}$, that is:

$$\boldsymbol{R} = \boldsymbol{\Gamma}^{\mathsf{T}}\,\boldsymbol{\Gamma} \tag{3.11}$$

Eventually the model response may be recast as a function of *independent* standard Gaussian random variables as follows, *i.e.* $Y = \mathcal{M}[\boldsymbol{X}(\boldsymbol{\xi})]$. Hence it may be expanded onto a classical PC expansion made of normalized Hermite polynomials, as shown in Section 2.2.

The general case of dependent random variables (*e.g.* with a more complex dependence structure than the Nataf distribution, which may for instance involve a tail dependence) has been theoretically considered in Soize and Ghanem (2004). However building up an orthonormal basis may be not an easy task in practice since it requires a perfect knowledge of the joint PDF f_X. Note that the concept of *copulas* (Nelsen, 1999) provides an elegant framework in this context for parametrizing the relative contributions of the margins and the dependence structure (see Sudret (2007, Chapter 4) for the link with PC expansions).

2.4 Case of an input random field

Many stochastic finite elements studies involve an input random field, *e.g.* spatially variable material properties in mechanics (Ghanem and Spanos, 2003). Let us denote by $H(\boldsymbol{z}, \omega)$ such a random field, where $\boldsymbol{z}$ is a spatial variable in a bounded domain $\mathcal{D} \subset \mathbb{R}^d$ ($d \in \{1, 2, 3\}$) and ω is the elementary event of the probability space $(\Omega, \mathcal{F}, \mathcal{P})$. The random field $H(\boldsymbol{z}, \omega)$ is assumed to be square-integrable, with mean $\mu(\boldsymbol{z})$ and autocorrelation function $C_H(\boldsymbol{x}, \boldsymbol{x}')$. $H(\boldsymbol{z}, \omega)$ may be described by a *finite* number M of random variables after a proper discretization scheme,

e.g. the *Karhunen-Loève expansion* (Loève, 1977):

$$H(\boldsymbol{z},\omega) \;=\; \mu(\boldsymbol{z}) \,+\, \sum_{i=1}^{+\infty} \sqrt{\lambda_i}\, \xi_i(\omega)\, \varphi_i(\boldsymbol{x}) \tag{3.12}$$

where the series converges in the $\mathcal{L}^2$-norm. In this equation $(\xi_i(\omega))_{i\in\mathbb{N}}$ is a sequence of uncorrelated, zero-mean and unit-variance random variables, and $(\lambda_i)_{i\in\mathbb{N}}$ and $(\varphi_i(\boldsymbol{x}))_{i\in\mathbb{N}}$ are the solutions of the generalized eigenvalue problem:

$$\int_{\mathcal{D}} C_H(\boldsymbol{x},\boldsymbol{x}')\, \varphi_i(\boldsymbol{x}')\, d\boldsymbol{x}' \;=\; \lambda_i\, \varphi_i(\boldsymbol{x}) \qquad , \qquad \forall i \;\in \mathbb{N}^* \tag{3.13}$$

The eigenvalues are indexed in decreasing order (*i.e.* $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_M \geq \ldots$).

For computational purpose the series in Eq.(3.12) is truncated after M terms, the value of which may be selected *a priori* with respect to a target accuracy of discretization (Sudret and Der Kiureghian, 2000). The truncated Karhunen-Loève expansion is an optimal approximation of $H(\boldsymbol{z},\omega)$ in the sense of the $\mathcal{L}^2$-norm. The reader is referred to Appendix B for more details. Although the problem in Eq.(3.13) admits a closed-form solution for particular choices of $C_H(\boldsymbol{x},\boldsymbol{x}')$, it generally requires the implementation of a numerical solving scheme. In this purpose, a Galerkin scheme may be used together with the approximation of the autocorrelation function C_H onto a suitable basis, *e.g.* a finite element-like basis (Ghanem and Spanos, 2003) or spectral bases such as orthogonal polynomials (Zhang and Ellingwood, 1994) and wavelets (Phoon et al., 2002).

If the random field $H(\boldsymbol{z},\omega)$ is Gaussian, the ξ_i's form a set of independent standard Gaussian random variables. Then the model response may be expanded onto a basis made of normalized Hermite polynomials as shown in Section 2.2. A particular class of non Gaussian random fields $H(\boldsymbol{z},\omega)$ may be cast as a non-linear transformation of a Gaussian random field. Such random fields are known as *translation fields* (Grigoriu, 1998). For instance, input parameters such as material properties are often modelled by *lognormal* random fields:

$$H(\boldsymbol{z},\omega) \;=\; \mathrm{e}^{N(\boldsymbol{z},\omega)} \tag{3.14}$$

where $N(\boldsymbol{z},\omega)$ is a Gaussian random field. $H(\boldsymbol{z},\omega)$ may be recast in terms of independent Gaussian random variables by substituting $N(\boldsymbol{z},\omega)$ for its Karhunen-Loève expansion in Eq.(3.14). The reader is referred to Lagaros et al. (2005) for a comprehensive overview of the methods for simulating non-Gaussian random fields.

2.5 Practical implementation

For practical implementation, *finite dimensional* polynomial chaoses have to be built. The usual choice consists in selecting those multivariate polynomials $\psi_{\boldsymbol{\alpha}}$ of total degree $\sum_{i=1}^{M} \alpha_i$ not greater

than a maximal degree p. The size of this finite-dimensional basis is denoted by P and given by:

$$P = \binom{M+p}{p} \tag{3.15}$$

The full procedure requires the two following steps:

- the construction of the sets of univariate orthonormal polynomials associated with each marginal PDF of the components of $\boldsymbol{X}$;
- an algorithm that builds the set of indices $\boldsymbol{\alpha}$ corresponding to the P M-variate polynomials of degree not greater than p. Sudret and Der Kiureghian (2000) proposed a strategy based on a *ball sampling algorithm*. A more efficient approach, which relies upon two specific algorithms for generating and permuting the components of index sets, has been devised by the author and is used throughout this work. The method is detailed in Appendix C.

3 Galerkin solution schemes

3.1 Brief history

The spectral stochastic finite element method (SSFEM) was proposed in the early 1990's to solve linear structural mechanics problem featuring spatially random coefficients (Ghanem and Spanos, 1991). In this setup, the input quantities are represented by Gaussian random fields that are discretized by the Karhunen-Loève expansion (Eq.(3.12)). The model response, *i.e.* the vector of nodal displacements, is expanded onto a polynomial chaos basis made of Hermite polynomials. The solution is computed by a Galerkin projection scheme in the random dimension.

SSFEM was applied to various fields such as geotechnical problems (Ghanem and Brzkala, 1996), transport in random media (Ghanem and Dham, 1998; Ghanem, 1998), non linear random vibrations (Li and Ghanem, 1998) and heat conduction (Ghanem, 1999c), in which non Gaussian random fields were introduced (see also Ghanem (1999b)). A general framework that summarizes the various developments can be found in Ghanem (1999a). On the other hand, a considerable work in numerical analysis has been accomplished for studying the convergence of various Galerkin solving schemes of stochastic PDE's. In particular, the elliptic case has received much attention (Deb et al., 2001; Babuška and Chatzipantelidis, 2002; Frauenfelder et al., 2005; Bieri and Schwab, 2009). Moreover, approximation schemes based on wavelet bases (Le Maître et al., 2004a,b) and multi-element bases (Wan and Karniadakis, 2005) have been investigated for solving problems featuring long-term integration and/or stochastic discontinuities, such as fluid mechanics problems governed by non linear Navier-Stokes equations.

In the sequel, SSFEM is detailed in the simple yet relevant case of a linear stochastic elliptic problem. It is shown that the method results in a large system of coupled equations, hence a considerable computational cost. Then the so-called *generalized spectral decomposition* (GSD) method, which has been recently designed (Nouy, 2007a) in order to reduce the problem to a series of low-dimensional systems, is described.

3.2 Spectral Stochastic Finite Element Method

3.2.1 Stochastic elliptic boundary value problem

Mechanical systems are commonly governed by partial differential equations (PDE). In particular, the equations of elasticity (without inertial terms) or thermal diffusion are *elliptic* PDEs with suitable boundary conditions. A simple and relevant deterministic model is the linear case described below:

$$\left\{\begin{array}{l} \nabla\cdot(\kappa(\boldsymbol{z})\nabla u(\boldsymbol{z})) \;=\; -\,f(\boldsymbol{z}) \quad , \quad \forall \boldsymbol{z}\in\mathcal{D} \\ \\ u(\boldsymbol{z})\Big|_{\Gamma_0} \;=\; 0 \qquad \kappa(\boldsymbol{z})\nabla u(\boldsymbol{z})\Big|_{\Gamma_1}\cdot\boldsymbol{n}(\boldsymbol{z}) \;=\; \bar{f}(\boldsymbol{z}) \\ \\ \Gamma_0 \,\cup\, \Gamma_1 \;=\; \partial\mathcal{D} \end{array}\right. \tag{3.16}$$

where:

- $\mathcal{D}\subset\mathbb{R}^d$ ($d\in\{1,2,3\}$) is a bounded spatial domain, whose boundary is denoted by $\partial\mathcal{D}$;
- $\kappa(\boldsymbol{z})$ is a diffusion coefficient;
- $u(\boldsymbol{z})$ is the response field, which is assumed to be scalar (*e.g.* a temperature field) for the sake of simplicity[1];
- $f(\boldsymbol{z})$ and $\bar{f}(\boldsymbol{z})$ are volume and surface loadings, respectively;
- $\boldsymbol{n}(\boldsymbol{z})$ is the exterior normal to the boundary Γ_1 at point $\boldsymbol{z}$;
- Γ_0 (resp. Γ_1) is a part of the boundary of $\mathcal{D}$ on which Dirichlet (resp. Neumann) boundary conditions are applied.

In the following, the domain $\mathcal{D}$ is assumed to be known with sufficient accuracy and is therefore considered to be deterministic. In contrast, the diffusion coefficient $\kappa(\boldsymbol{z})$ is modelled as a random field $\kappa(\boldsymbol{z},\omega)$ since it may be affected by uncertainty. The loading is also represented by random

[1] A *vector* field $\boldsymbol{u}(\boldsymbol{z})$ (*e.g.* a displacement field) may also be considered upon redefining the problem.

fields $f(\boldsymbol{z},\omega)$ and $\bar{f}(\boldsymbol{z},\omega)$. As a consequence, the solution is also a random field $u(\boldsymbol{z},\omega)$. This leads to the following linear *stochastic* elliptic boundary value problem:

$$\left\{\begin{array}{l} \nabla\cdot(\kappa(\boldsymbol{z},\omega)\nabla u(\boldsymbol{z},\omega)) \;=\; -\,f(\boldsymbol{z},\omega) \quad , \quad \forall \boldsymbol{z}\in\mathcal{D} \\ \\ u(\boldsymbol{z},\omega)\Big|_{\Gamma_0} \;=\; 0 \qquad \kappa(\boldsymbol{z},\omega)\nabla u(\boldsymbol{z},\omega)\Big|_{\Gamma_1}\cdot\boldsymbol{n}(\boldsymbol{z}) \;=\; \bar{f}(\boldsymbol{z},\omega) \end{array}\right. \quad , \quad \text{almost surely} \tag{3.17}$$

The variational form of the stochastic problem (3.17) reads:

$$\text{Find } u(\boldsymbol{z},\omega)\in\mathcal{V}\otimes\mathcal{S}\text{:} \qquad \mathbb{E}\left[b(u,v,\omega)\right] \;=\; \mathbb{E}\left[l(v,\omega)\right] \qquad \forall v\in\mathcal{V}\otimes\mathcal{S} \tag{3.18}$$

where

$$b(u,v,\omega) \;\equiv\; \int_{\mathcal{D}} \nabla v(\boldsymbol{z},\omega)^{\mathsf{T}}\,\kappa(\boldsymbol{z},\omega)\,\nabla u(\boldsymbol{z},\omega)\,d\boldsymbol{z} \tag{3.19}$$

and

$$l(v,\omega) \;\equiv\; \int_{\mathcal{D}} f(\boldsymbol{z},\omega)\,v(\boldsymbol{z},\omega)\,d\boldsymbol{z} \;+\; \int_{\Gamma_1} \bar{f}(\boldsymbol{z},\omega)\,v(\boldsymbol{z},\omega)\,d\boldsymbol{z} \tag{3.20}$$

The solution of the problem is sought in the tensor-product space $\mathcal{V}\otimes\mathcal{S}$, where $\mathcal{V}$ is an appropriate set of real valued functions defined on $\mathcal{D}$ and $\mathcal{S}$ is a space of random variables ($\mathcal{L}^2(\Omega,\mathcal{F},\mathcal{P};\mathbb{R})$ is often a reasonable choice in practice).

3.2.2 Discretization of the problem

First of all, the random fields $\kappa(\boldsymbol{z},\omega)$, $f(\boldsymbol{z},\omega)$ and $\bar{f}(\boldsymbol{z},\omega)$ are discretized using a suitable method, *e.g.* the Karhunen-Loève expansion (Loève, 1977). Thus they can be cast as functions of a finite set of M independent random variables $\{X_1,\ldots,X_M\}$.

For fixed elementary event ω, the deterministic spatial functions $u(\boldsymbol{z},\omega)$ and $v(\boldsymbol{z},\omega)$ may be sought in a finite element-like subspace:

$$u(\boldsymbol{z},\omega) \;\approx\; \sum_{i=1}^{\mathcal{N}} u_i(\omega)\,N_i(\boldsymbol{z}) \;\equiv\; \boldsymbol{U}^{\mathsf{T}}(\omega)\boldsymbol{N}(\boldsymbol{z}) \tag{3.21}$$

where the $N_i(\boldsymbol{z})$'s are the usual shape functions and the following vector notation has been used:

$$\boldsymbol{U}^{\mathsf{T}}(\omega) \;\equiv\; \{u_1(\omega),\ldots,u_{\mathcal{N}}(\omega)\}^{\mathsf{T}} \qquad , \qquad \boldsymbol{N}(\boldsymbol{z}) \;\equiv\; \{N_1(\boldsymbol{z}),\ldots,N_{\mathcal{N}}(\boldsymbol{z})\}^{\mathsf{T}} \tag{3.22}$$

The bilinear and linear forms $b(u,v,\omega)$ and $l(v,\omega)$ are respectively approximated by:

$$b(u,v,\omega) \;\approx\; \boldsymbol{V}^{\mathsf{T}}(\omega)\,\underbrace{\int_{\mathcal{D}} \nabla\boldsymbol{N}(\boldsymbol{z})\,\kappa(\boldsymbol{z},\omega)\,\nabla\boldsymbol{N}^{\mathsf{T}}(\boldsymbol{z})\,d\boldsymbol{z}}_{\equiv\;\boldsymbol{K}(\omega)}\,\boldsymbol{U}(\omega) \tag{3.23}$$

$$l(v,\omega) \;\approx\; \boldsymbol{V}^{\mathsf{T}}(\omega)\,\underbrace{\left(\int_{\mathcal{D}} f(\boldsymbol{z},\omega)\,\boldsymbol{N}(\boldsymbol{z})\,d\boldsymbol{z} \;+\; \int_{\Gamma_1} \bar{f}(\boldsymbol{z},\omega)\boldsymbol{N}(\boldsymbol{z})\,d\boldsymbol{z}\right)}_{\equiv\;\boldsymbol{F}(\omega)} \tag{3.24}$$

Hence the following *semi-discretized* version of the variational problem in Eq.(3.18):

$$\begin{gathered}\text{Find } \boldsymbol{U}(\omega) \in \mathcal{S}^{\mathcal{N}} \quad \text{such that} \quad \forall\, \boldsymbol{U}(\omega) \in \mathcal{S}^{\mathcal{N}} \ : \\ \mathbb{E}\left[\boldsymbol{V}^{\mathsf{T}}(\omega)\ \boldsymbol{K}(\omega)\boldsymbol{U}(\omega)\right] \ = \ \mathbb{E}\left[\boldsymbol{V}^{\mathsf{T}}(\omega)\ \boldsymbol{F}(\omega)\right]\end{gathered} \tag{3.25}$$

In addition, for fixed $\boldsymbol{z}$, the random variables $U_i(\omega)$ in Eq.(3.21) may be sought in a suitable approximation subspace of $\mathcal{S}$, *e.g.* the space $\mathcal{S}_P$ spanned by a truncated polynomial chaos basis $\{\psi_j(\omega), j = 0, \ldots, P-1\}$ (see Section 2):

$$U_i(\omega) \ \approx \ \sum_{j=0}^{P-1} u_i^j \ \psi_j(\omega) \tag{3.26}$$

Hence the random vector $\boldsymbol{U}(\omega)$ rewrites:

$$\boldsymbol{U}(\omega) \ \approx \ \sum_{j=0}^{P-1} \boldsymbol{u}_j \ \psi_j(\omega) \qquad , \qquad \boldsymbol{u}_j \ \equiv \ \{u_j^1, \ldots u_j^{\mathcal{N}}\}^{\mathsf{T}} \tag{3.27}$$

Gathering the vectors $\{\boldsymbol{u}_j, j = 0, \ldots, P-1\}$ (resp. $\{\boldsymbol{v}_j, j = 0, \ldots, P-1\}$) into a block vector U (resp. V) of size $\mathcal{N} \times P$, the variational problem may be recast under the following *fully discretized* form:

$$\begin{gathered}\text{Find } \mathsf{U} \in \mathbb{R}^{\mathcal{N}\times P} \quad \text{such that} \quad \forall\, \mathsf{V} \in \mathbb{R}^{\mathcal{N}\times P} \ : \\ \sum_{i=0}^{P-1}\sum_{j=0}^{P-1} \boldsymbol{v}_i^{\mathsf{T}}\ \mathbb{E}\left[\boldsymbol{K}(\omega)\psi_i(\omega)\psi_j(\omega)\right]\ \boldsymbol{u}_j \ = \ \sum_{i=0}^{P-1} \boldsymbol{v}_i^{\mathsf{T}}\ \mathbb{E}\left[\boldsymbol{F}(\omega)\psi_i(\omega)\right]\end{gathered} \tag{3.28}$$

which reduces to a set of linear equations:

$$\sum_{j=0}^{P-1} \mathbb{E}\left[\boldsymbol{K}(\omega)\psi_i(\omega)\psi_j(\omega)\right]\ \boldsymbol{u_j} \ = \ \mathbb{E}\left[\boldsymbol{F}(\omega)\psi_i(\omega)\right] \qquad , \quad i = 0, \ldots, P-1 \tag{3.29}$$

These equations are usually arranged in a matrix linear system of size $\mathcal{N} \times P$:

$$\mathsf{K}\,\mathsf{U} \ = \ \mathsf{F} \tag{3.30}$$

where F is a block vector whose i-th block is $\mathsf{F}_i \equiv \mathbb{E}\left[\boldsymbol{F}(\omega)\psi_i\right]$ and K is a block matrix whose (i,j)-block is $\mathsf{K}_{i,j} \equiv \mathbb{E}\left[\boldsymbol{K}(\omega)\psi_i(\omega)\psi_j(\omega)\right]$.

3.3 Computational issues

As the linear system in Eq.(3.30) is usually very large and sparse, it is not recommended to use direct resolution techniques. Instead Krylov-type iterative solvers may be preferred, such as preconjugate gradient techniques (Ghanem and Kruger, 1996; Pellissetti and Ghanem, 2000;

Keese and Matthies, 2005; Chung et al., 2005). However these schemes may require a great computational cost as well as important memory requirements.

As an alternative, the cost associated with Galerkin solution schemes may be decreased by approximating the solution on an optimal reduced basis, *i.e.* which captures the main features of the unknown random field by means of a small number of basis functions. It is clear that such a basis cannot be determined *a priori* since the solution is unknown. It has been proposed in Ghanem et al. (2006) to first compute a crude approximation of the solution on a coarse mesh in order to obtain approximate response second moments. Then the accurate solution (*i.e.* on a fine mesh) is sought on a Karhunen-Loève decomposition of the approximate covariance kernel. A similar strategy may be found in Matthies and Keese (2005), where a coarse approximation of the response covariance is obtained using a Neumann series expansion. The so-called *reduced stochastic basis* method (Nair and Keane, 2002; Sachdeva et al., 2006) has also been developed to downsize the problem under consideration. In this approach, the model response is sought onto a reduced stochastic basis, which is a basis of a low-dimensional Krylov subspace. All these methods are dedicated to linear problems or problems featuring low nonlinearity.

Lastly, the so-called *generalized spectral decomposition* (GSD) method has been investigated in Nouy (2005, 2007b,a, 2008). In contrast to the other methods, it is aimed at building *iteratively* (and not *ab initio*) a reduced basis. The strategy has been recently extended to non linear problems (Nouy and Le Maître, 2009). Only slight modifications of the computer code at hand are then required compared to SSFEM, making GSD an attractive solving scheme. The method is briefly outlined for the linear case in the next section.

3.4 Generalized spectral decomposition (Nouy, 2007a)

The generalized spectral decomposition (GSD) aims at building up *iteratively* an *optimal* solution of the stochastic PDE under the form:

$$\boldsymbol{U}(\omega) \approx \sum_{i=1}^{m} \boldsymbol{u}_i \, \lambda_i(\omega) \tag{3.31}$$

where the $\boldsymbol{u}_i$'s are deterministic vectors and the $\lambda_i(\omega)$'s are random variables. Optimality means that the above approximation can reach a maximum accuracy in some sense for a given number of terms m. Instead of solving a large system such as in Eq.(3.30), GSD consists in solving several low-dimensional problems, hence a dramatic reduction of the computational cost and memory requirements.

3.4.1 Definition of the GSD solution

Let us first introduce the following vector/matrix notation:

$$\mathbf{U} \equiv \{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m\}^\mathsf{T} \quad , \quad \boldsymbol{\lambda}(\omega) \equiv \{\lambda_1(\omega), \ldots, \lambda_m(\omega)\}^\mathsf{T} \tag{3.32}$$

Then the GSD of $\boldsymbol{U}(\omega)$ rewrites:

$$\boldsymbol{U}(\omega) \approx \mathbf{U}^\mathsf{T} \boldsymbol{\lambda}(\omega) \tag{3.33}$$

The GSD method is aimed at solving the following problem:

$$\begin{gathered}\text{Find } (\mathbf{U}, \boldsymbol{\lambda}(\omega)) \in \mathbb{R}^{m\times\mathcal{N}} \times (\mathcal{S}_P)^m \quad \text{such that} \quad \forall\, (\mathbf{V}, \boldsymbol{\mu}(\omega)) \in \mathbb{R}^{m\times\mathcal{N}} \times (\mathcal{S}_P)^m \ : \\ \mathbb{E}\left[\left(\boldsymbol{\mu}^\mathsf{T}(\omega)\, \mathbf{U} + \boldsymbol{\lambda}(\omega)^\mathsf{T}\, \mathbf{V}\right)\, \boldsymbol{K}(\omega)\, \mathbf{U}^\mathsf{T}\, \boldsymbol{\lambda}(\omega)\right] = \mathbb{E}\left[\left(\boldsymbol{\mu}^\mathsf{T}(\omega)\, \mathbf{U} + \boldsymbol{\lambda}(\omega)^\mathsf{T}\, \mathbf{V}\right)\, \boldsymbol{F}(\omega)\right]\end{gathered} \tag{3.34}$$

For the sake of clarity, the dependence of the random variables and vectors on ω is dropped from now on.

3.4.2 Computation of the terms in the generalized spectral decomposition

The GSD approach consists in seeking a solution by *alternatively* solving the variational problem with respect to $\mathbf{U}$ and $\boldsymbol{\lambda}$. On the one hand, solving the variational problem with respect to $\boldsymbol{\lambda}$ for fixed $\mathbf{U}$ reads:

$$\begin{gathered}\text{Find } \boldsymbol{\lambda} \in (\mathcal{S}_P)^m \quad \text{such that} \quad \forall\, \boldsymbol{\mu} \in (\mathcal{S}_P)^m \ : \\ \mathbb{E}\left[\boldsymbol{\mu}^\mathsf{T}\, (\mathbf{U}\, \boldsymbol{K}\, \mathbf{U}^\mathsf{T})\, \boldsymbol{\lambda}\right] = \mathbb{E}\left[\boldsymbol{\mu}^\mathsf{T}\, \mathbf{U}\, \boldsymbol{F}\right]\end{gathered} \tag{3.35}$$

This can be interpreted as the natural way to find the best set of stochastic functions associated with given deterministic vectors. The problem in Eq.(3.35) leads to a linear system of size $m \times P$, which is computationally inexpensive compared to the $\mathcal{N} \times P$ problem associated to a classical decomposition (Eq.(3.30)).

On the other hand, solving the variational problem with respect to $\mathbf{U}$ for fixed $\boldsymbol{\lambda}$ reads:

$$\begin{gathered}\text{Find } \mathbf{U} \in \mathbb{R}^{m\times\mathcal{N}} \quad \text{such that} \quad \forall\, \mathbf{V} \in \mathbb{R}^{m\times\mathcal{N}} \ : \\ \mathbb{E}\left[\boldsymbol{\lambda}^\mathsf{T}\, (\mathbf{V}\, \boldsymbol{K}\, \mathbf{U}^\mathsf{T})\, \boldsymbol{\lambda}\right] = \mathbb{E}\left[\boldsymbol{\lambda}^\mathsf{T}\, \mathbf{V}\, \boldsymbol{F}\right]\end{gathered} \tag{3.36}$$

This corresponds to finding the best deterministic vectors associated with given stochastic functions. This leads to solve a $\mathcal{N} \times m$ linear system. Note that if the stochastic functions are chosen as the polynomial chaos basis functions $\boldsymbol{\psi} \equiv \{\psi_0, \ldots, \psi_{P-1}\}$, then solving the problem in Eq.(3.36) provides the classical P-term solution of the linear system (3.30). In contrast, the GSD method aims at obtaining a solution with a much smaller number of terms, *i.e.* $m << P$.

Let us denote by $\boldsymbol{\lambda} = \mathcal{F}(\mathbf{U})$ and $\mathbf{U} = \mathcal{G}(\boldsymbol{\lambda})$ the solutions of problems (3.35) and (3.36), respectively. So the GSD alternate solving procedure may be cast as a fixed-point problem in $\mathbf{U}$ as follows:

$$\mathbf{U} \;=\; \mathcal{T}(\mathbf{U}) \qquad , \qquad \mathcal{T} \;\equiv\; \mathcal{G} \circ \mathcal{F} \tag{3.37}$$

From properties of operator $\mathcal{T}$, this problem can be interpreted as a generalized eigenvalue problem (Nouy, 2008). This leads to the use of solving schemes inspired by algorithms dedicated to classical eigenvalue problems.

The basic algorithm is the *subspace iteration method*, outlined below:

1. Initialize $\mathbf{U}^{(0)} \in \mathbb{R}^{m \times \mathcal{N}}$
2. For $k = 1$ to k_{max}
   (a) Compute $\mathbf{U}^k = \mathcal{T}(\mathbf{U}^{k-1})$
   (b) Orthonormalize $\mathbf{U}^k$
3. End for
4. Set $\mathbf{U} = \mathbf{U}^{(k)}$ and compute $\boldsymbol{\lambda} = \mathcal{F}(\mathbf{U})$

At each iteration k, Step 2.(a) may be split into two substeps:

$$\boldsymbol{\lambda}^{(k-1)} \;=\; \mathcal{F}(\mathbf{U}^{(k-1)}) \qquad \text{then} \qquad \mathbf{U}^{(k)} \;=\; \mathcal{G}(\boldsymbol{\lambda}^{(k-1)}) \tag{3.38}$$

Other algorithms may be employed in order to solve problem (3.37), such as the so-called *Arnoldi type algorithm* (Nouy, 2008).

3.4.3 Step-by-step building of the generalized spectral decomposition

In practice, the number m of terms in the GSD in order to reach a prescribed accuracy is not known *a priori*. An iterative scheme is devised in order to determine each term one after another in the decomposition. Assume that a m-term approximation $\mathbf{U}_m^{\mathsf{T}}\boldsymbol{\lambda}_m$ has been built up, *e.g.* using the subspace iteration method. Let us define the following residual:

$$\boldsymbol{F}^m \;\equiv\; \boldsymbol{F} \;-\; \boldsymbol{K}\mathbf{U}_m^{\mathsf{T}}\boldsymbol{\lambda}_m \tag{3.39}$$

We denote by $\mathcal{F}^{(m)}$ and $\mathcal{G}^{(m)}$ the operators associated with problems in Eqs.(3.35) and (3.36) in which $\boldsymbol{F}$ has been replaced with $\boldsymbol{F}^m$. This allows the definition of the *deflated* operator $\mathcal{T}^{(m)} \equiv \mathcal{G}^{(m)} \circ \mathcal{F}^{(m)}$.

Thus the following algorithm may be devised:

1. Initialize $\boldsymbol{F}^{(0)} = \boldsymbol{F}$, $\mathbf{U}_0 = \emptyset$, $\boldsymbol{\lambda}_0 = \emptyset$
2. For $m = 1$ to m_{max}
   (a) Compute the solution $\widehat{\boldsymbol{u}}_m$ of the pseudo eigenproblem $\boldsymbol{u} = \mathcal{T}^{(m-1)}(\boldsymbol{u})$, *e.g.* using the subspace iteration method (or power iteration in this case)
   (b) Update the set of deterministic vectors: $\mathbf{U}_m = \{\mathbf{U}_{m-1}\ ,\ \widehat{\boldsymbol{u}}_m\}^{\mathsf{T}}$
   (c) Compute the corresponding new random variable: $\widehat{\lambda}_m = \mathcal{F}^{m-1}(\widehat{\boldsymbol{u}}_m)$
   (d) Update the set of random variables: $\boldsymbol{\lambda}_m = \{\boldsymbol{\lambda}_{m-1}\ ,\ \widehat{\lambda}_m\}^{\mathsf{T}}$
   (e) Stop if the target accuracy has been reached
3. End for

Note that it is also possible to determine several pairs $(\boldsymbol{u}_i, \lambda_i)$ in one shot in Step 2.(a). In this case the algorithm adds a number $r > 1$ of terms at each iteration. Besides, in order to obtain a more accurate solution, it has been proposed to substitute Steps 2.(c)-(d) for a *global updating* of the random variables in the GSD as follows:

$$\text{Compute } \boldsymbol{\lambda}_m \ = \ \mathcal{F}(\mathbf{U}_m) \tag{3.40}$$

The proposed step-by-step procedure requires to solve several small problems of size $\mathcal{N} \times 1$ and $1 \times \mathcal{P}$ (possibly $m \times \mathcal{P}$ if using global updating), instead of a single huge system of size $\mathcal{N} \times \mathcal{P}$ as in SSFEM.

The method has been successfully extended to nonlinear problems in Nouy and Le Maître (2009), making GSD an attractive strategy for solving a large class of models. A nice feature of GSD it that is only requires slight modifications of the already existing computer code compared to the usual Galerkin scheme. As the present work is aimed at addressing an even wider class of stochastic problems without adapting the governing equations, we focus our attention on the so-called *non intrusive* approaches in the sequel.

4 Non intrusive methods

4.1 Introduction

Of interest are the so-called *non intrusive* approaches in the current section. In contrast to the Galerkin-based schemes, they do not require any modification of the deterministic model, which is considered to be a black-box. Two categories of methods are distinguished, namely:

- an *interpolating* approach, known as the *stochastic collocation method*, in which the polynomial approximation is constrained to fit *exactly* the model response at a suitable point set. This corresponds to the so-called *pseudo-spectral methods* in Boyd (1989), which rely upon well-established results on Lagrange polynomial interpolation. The stochastic collocation method is described in Section 4.2;

- a *non interpolating* approach, in which the PC coefficients are computed by minimizing the mean-square error of approximation. Two kinds of coefficients estimates may be considered, *i.e.* estimates based on *spectral projection* (Section 4.3) and estimates based on *least-square regression* (Section 4.4).

4.2 Stochastic collocation method

The *stochastic collocation* (SC) method has received much interest in the past few years (Nobile et al., 2006; Xiu and Hesthaven, 2005; Ganapathysubramanian and Zabaras, 2007; Babuška et al., 2007; Foo et al., 2008; Lin and Tartakovsky, 2009; Bieri and Schwab, 2009). It relies upon a polynomial interpolation of the model response at a suitable set of realizations of the input random vector $\boldsymbol{X}$. SC takes benefit from the well-established theory of Lagrange interpolation. A rigorous convergence and error analysis of the method can be found in Babuška et al. (2007); Bieri and Schwab (2009) for linear elliptic boundary value problems. Other kinds of basis have also been considered for interpolation, such as piecewise linear functions (Klimke, 2006) and radial basis functions (McDonald et al., 2006). Is it worth mentioning that the Gaussian Process technique reviewed in Chapter 2, Section 3.2 may be viewed as a specific method for multivariate interpolation as well.

As pointed out in Barthelmann et al. (2000), two different interpolation problems can be distinguished:

- given a set of data of the form $\{(\boldsymbol{x}^{(i)}, y^{(i)}), i = 1, \ldots, N\}$, find a *smooth* function (*e.g.* a low-degree polynomial) $\widehat{\mathcal{M}}$ such that $\widehat{\mathcal{M}}(\boldsymbol{x}^{(i)}) = y^{(i)}$ for $i = 1, \ldots, N$;

- select a suitable point set $\mathcal{X} \equiv \{\boldsymbol{x}^{(i)}, i = 1, \ldots, N\}$ such that an accurate approximation of a given function $\mathcal{M}$ may be achieved by an appropriate interpolation on $\mathcal{X}$.

The present section is focused on the second problem. Indeed, we want to approximate the response of a model $\mathcal{M}(\boldsymbol{X})$ using a small set of optimally selected points. Classical results on univariate polynomial interpolation are first introduced. Then the multivariate problem is tackled.

4.2.1 Univariate Lagrange interpolation

Let us consider a model $Y = \mathcal{M}(X)$ that only depends on a single random variable X with prescribed PDF $f_X(x)$ and support $\mathcal{D}_X$. One first considers the case of a random variable with a *bounded* support $\mathcal{D}_X$. For the sake of simplicity, variable X is rescaled in such a way that its support is $[-1, 1]$.

Let $\mathcal{X} \equiv \{x^{(1)}, \ldots, x^{(n)}\}$ be some univariate point set or *experimental design* (ED) (we will see below how to choose these points optimally). Let $\mathcal{Y} \equiv \{y^{(1)}, \ldots, y^{(n)}\}$ be the associated model evaluations. The polynomial interpolation problem reads as follows:

$$\text{Find } \mathcal{M}_n \in \mathcal{P}_{n-1} \quad : \qquad \mathcal{M}_n(\boldsymbol{x}^{(i)}) \;=\; y^{(i)} \qquad , \qquad i = 1, \ldots, n \tag{3.41}$$

where $\mathcal{P}_{n-1}$ denotes the space of one-dimensional polynomials of degree less than or equal to $n-1$. This problem always admits a unique solution.

Let us define the *Lagrange basis* $(\ell_i)_{1 \leq i \leq n}$ related to $\mathcal{X}$ by:

$$\ell_i(x) \;=\; \prod_{j \neq i} \frac{x - x^{(j)}}{x^{(i)} - x^{(j)}} \qquad , \qquad i = 1, \ldots, n \tag{3.42}$$

The *Lagrange polynomials* ℓ_i satisfy $\ell_i(x^{(j)}) = \delta_{i,j}$. The interpolating polynomial $\mathcal{M}_n$ may be cast in the Lagrange basis as follows:

$$\mathcal{M}_n(X) \;=\; \sum_{i=1}^{n} y^{(i)}\, \ell_i(X) \tag{3.43}$$

The uniform convergence of the approximation $\mathcal{M}_n$ to the model function $\mathcal{M}$ when increasing n is strongly affected by the choice of $\mathcal{X}$. Indeed, one gets the following result (see *e.g.* Smith (2006)):

$$\|\mathcal{M} \;-\; \mathcal{M}_n\|_\infty \;\leq\; (1 + \Lambda_n)\; \|\mathcal{M} - \mathcal{M}_n^*\|_\infty \tag{3.44}$$

where $\|\cdot\|_\infty$ denotes the maximum norm. $\mathcal{M}_n^*$ is the (unknown) best approximating polynomial of $\mathcal{M}$ in the sense of this norm. Λ_n is the so-called *Lebesgue constant* defined by:

$$\Lambda_n \;\equiv\; \max_{-1 \leq x \leq 1} \sum_{i=1}^{n} |\ell_i(x)| \tag{3.45}$$

The inequality in Eq.(3.44) shows that the interpolation error is uniformly bounded by a product of two factors, namely a factor that only depends on the smoothness of the model function $\mathcal{M}$ (which cannot be controlled) and a factor that only depends on the experimental design $\mathcal{X}$. In particular, if we choose equally spaced points $x^{(i)}$ (uniform grid), the Lebesgue constant grows exponentially:

$$\Lambda_n \;\sim\; \frac{2^{n+1}}{\mathrm{e}\; n \log n} \qquad , \qquad n \to +\infty \tag{3.46}$$

where e $\equiv$ exp(1). Hence the interpolating polynomial $\mathcal{M}_n$ may not converge to the model function $\mathcal{M}$ when increasing the resolution n of the grid $\mathcal{X}$. This phenomenon is illustrated in Figure 3.1 by the well-known *Runge function* (Runge, 1901) defined by $\mathcal{M} : x \mapsto y \equiv (1 + 25x^2)^{-1}$, for all $x \in [-1, 1]$.

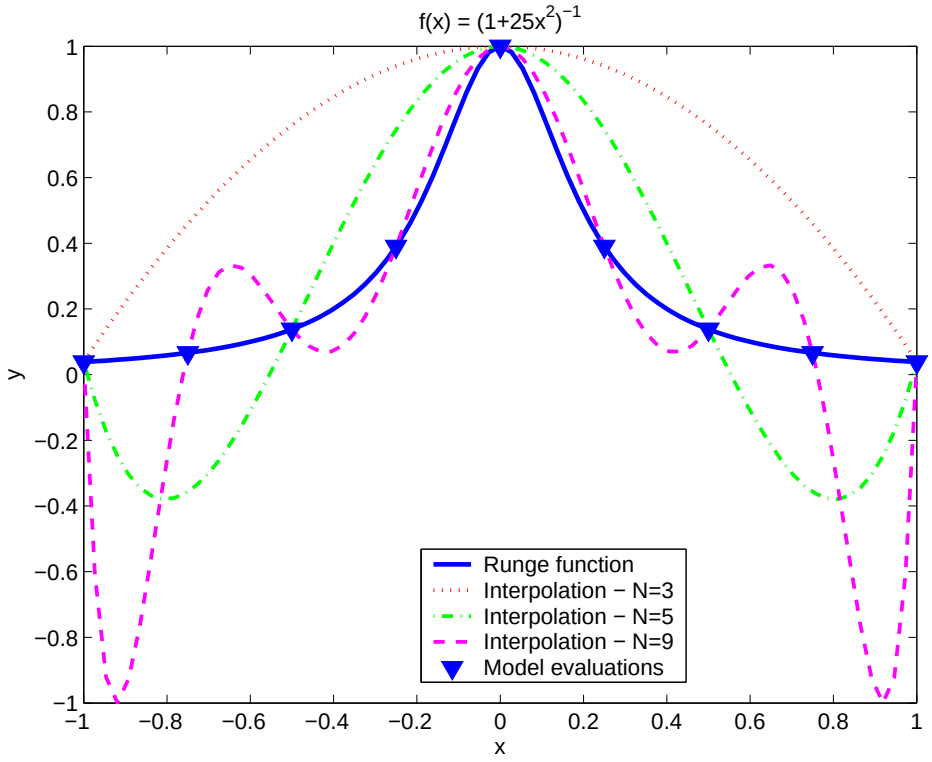


Figure 3.1: *Runge function - Illustration of the Runge phenomenon: the absolute error of approximation increases when refining the sample grid*

The so-called *Cauchy theorem* may help shed light on this phenomenon. It states that if function $\mathcal{M}$ is sufficiently smooth to have continuous derivatives at least up to order n, *i.e.* $\mathcal{M} \in \mathcal{C}^n([-1, 1])$, then:

$$\mathcal{M}(x) - \mathcal{M}_n(x) = \frac{\mathcal{M}^{(n)}(\xi_x)}{n!} w_n^{\mathcal{X}}(x) \quad , \quad \xi_x \in [-1, 1] \tag{3.47}$$

where $w_n^{\mathcal{X}}(x)$ is the *nodal polynomial associated with the grid* $\mathcal{X}$ defined by:

$$w_n^{\mathcal{X}}(x) = \prod_{i=1}^{n} (x - x^{(i)}) \tag{3.48}$$

In Eq.(3.47), we have no control on $\mathcal{M}^{(n)}$, which may take large values. For instance, for the Runge function we have $\|\mathcal{M}^{(n)}\|_\infty = n!5^n$. So one should select a grid $\mathcal{X}$ ensuring a small $\|w_n^{\mathcal{X}}\|_\infty$. The smallest possible value is obtained when using the n zeros of the Chebyshev polynomial of degree n, which leads to $\|w_n^{\mathcal{X}}\|_\infty = 2^{1-n}$. The associated grid is called the *Gauss-Chebyshev* (CG) grid. It has much better properties than the uniform grid considered so far. In

particular, according to Eq.(3.47), for any model function $\mathcal{M} \in \mathcal{C}^n([-1,1])$:

$$\|\mathcal{M}(x) - \mathcal{M}_n(x)\|_\infty \leq \frac{1}{2^{n-1}n!} \left\|\mathcal{M}^{(n)}\right\|_\infty \tag{3.49}$$

Then the interpolating polynomial converges very rapidly towards $\mathcal{M}$ if its derivative of order n is uniformly bounded. Also, the Lebesgue constant associated with the CG grid is small:

$$\Lambda_n(CG) \sim \frac{2}{\pi} \log n \quad , \quad n \to \infty \tag{3.50}$$

Grids based on the *extrema* rather than the roots of Chebyshev polynomials are also often selected. Such grids are known as *Gauss-Chebyshev-Lobatto* (GCL) or *Clenshaw-Curtis* grids in the literature (Brutman, 1978). This choice has the advantage to reuse the points of the grid when doubling the number of points. Then the GCL point sets are said to be *nested*. Alternatively, grids based on the roots or the extrema of Legendre polynomials may be used. They are referred to as *Gauss-Legendre* and *Gauss-Legendre-Lobatto* grids, respectively. Their associated Legendre constant is $\mathcal{O}(\sqrt{n})$. For the sake of illustration, interpolation of the Runge function is carried out using a CGL grid (Figure 3.2). The Runge phenomenon is no more observed, and the interpolating polynomials does converge toward the target function when refining the grid.

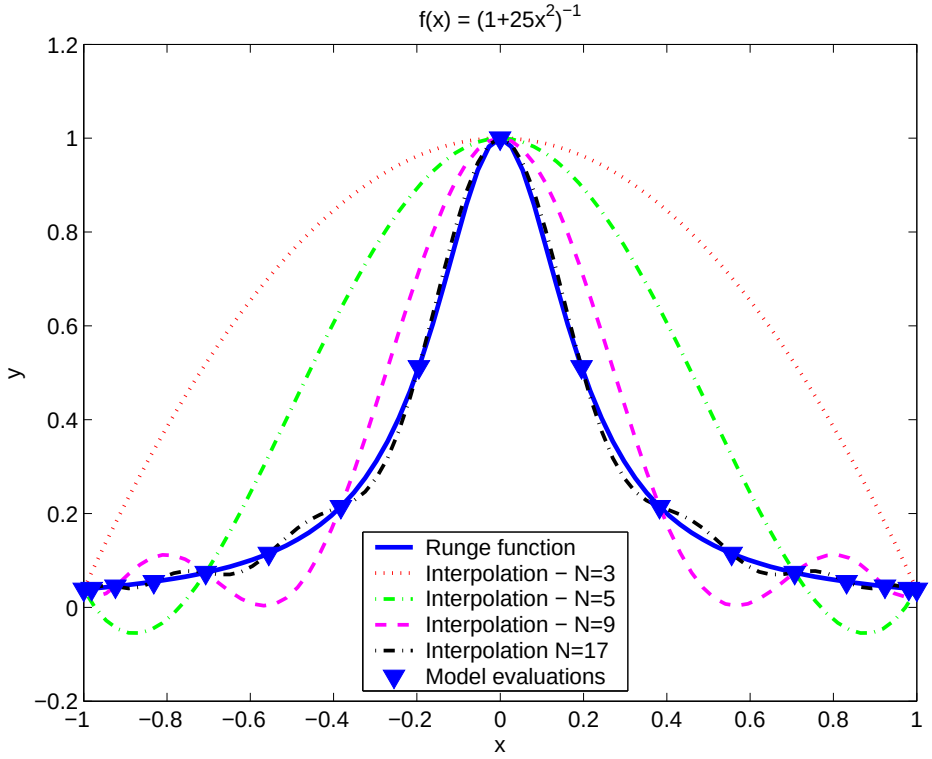


Figure 3.2: *Runge function - Polynomial interpolation based on nested Gauss-Chebyshev-Lobatto grids: the approximation error decreases when refining the sample grid*

In the framework of uncertainty propagation, one deals with a model $\mathcal{M}$ depending on a random variable X. All the results in polynomial interpolation that have been presented in this section

are directly applicable to *uniform* random variables X. As shown in Babuška et al. (2007), other kinds of random variables may be handled in conjonction with appropriate Gauss grids, *i.e.* grids made of roots of a family of orthogonal polynomials with respect to the probability density function of X. For instance, a Gauss-Hermite grid should be used for a Gaussian random variable. More generally, the same correspondence between the type of the distribution and the kind of orthogonal polynomials as in Table 3.1 (Section 2.2) can be exploited.

4.2.2 Multivariate Lagrange interpolation

The Lagrange interpolation problem always admits a unique solution in the univariate case. However multivariate polynomial interpolation is more complicated since the existence of a solution depends on the choice of the point set $\mathcal{X}$ (see Gasca and Sauer (2000) for more details). So there has been interest in identifying point sets and polynomial subspaces for which the interpolation problem can be solved.

It is tempting to extend the one-dimensional Lagrange interpolation to the multivariate case using a tensor-product formulation. In this setting, the points must have a grid structure and the approximation space is the set of M-variate polynomials of *partial* degree not greater than $N-1$. Let us define the *tensor-product Lagrange basis* by:

$$\mathcal{L}_{\boldsymbol{\alpha}}(\boldsymbol{x}) \;=\; \prod_{k=1}^{M} \ell_{\alpha_i,k}(x_k) \qquad , \qquad i = 1, \ldots, N \tag{3.51}$$

where the multi-index notation $\boldsymbol{\alpha} \equiv \{\alpha_1, \ldots, \alpha_M\}$ is used. Similarly to the univariate case, one gets $\mathcal{L}_i(\boldsymbol{x}^{(j)}) = \delta_{i,j}$. Hence the interpolating polynomial may be cast as:

$$\mathcal{M}_N(\boldsymbol{x}) \;=\; \sum_{i=1}^{N} y^{(i)}\, \mathcal{L}_i(\boldsymbol{x}) \tag{3.52}$$

On the other hand, the experimental design $\mathcal{X}$ is selected as the tensor product of M suitable one-dimensional point sets (*e.g.* GC or GCL grids) $\{(x_i^{(1)}, \ldots, x_i^{(n_i)}), i = 1, \ldots, M\}$. The model response $y \equiv \mathcal{M}(\boldsymbol{x})$ is approximated as follows:

$$\mathcal{M}(\boldsymbol{x}) \;\approx\; \mathcal{M}_{n_1,\ldots,n_M}(\boldsymbol{x}) \;\equiv\; \sum_{i_1=1}^{n_1} \ldots \sum_{i_M=1}^{n_M} \mathcal{M}(x_1^{(i_1)}, \ldots, x_M^{(i_M)})\, \ell_{i_1}(x_1^{(i_1)}) \ldots \ell_{i_M}(x_M^{(i_M)}) \tag{3.53}$$

In the case of an *isotropic* formula (*i.e.* $n_1 = \cdots = n_M = n$), the required number of model evaluations (*i.e.* the computational cost) grows exponentially with the number M of input random variables, *i.e.* $N = n^M$. Hence the tensor-product scheme suffers the curse of dimensionality.

In order to bypass this difficulty, the use of *Smolyak formulæ* (Smolyak, 1963) has been recommended in Barthelmann et al. (2000). These formulæ are linear combinations of tensor-products

in Eq.(3.53) which make use of a small number of model evaluations. The Smolyak formula of level l is defined by:

$$\mathcal{M}_l^{Smolyak}(\boldsymbol{x}) \equiv \sum_{l \leq |\boldsymbol{k}| \leq l+M-1} (-1)^{l+M-|\boldsymbol{k}|-1} \binom{M-1}{|\boldsymbol{k}|-l} \mathcal{M}_{n(k_1),\ldots,n(k_M)}(\boldsymbol{x}) \tag{3.54}$$

In the above equation, $n(k_i)$ $(i = 1, \ldots, M)$ denotes the number of nodes associated to the one-dimensional quadrature rule of level k_i. For instance, one gets $n(k_i) = k_i$ (resp. $n(k_i) = 2^{k_i-1}+1$) in case of a GC (resp. GCL) quadrature rule. The grid $\mathcal{X}$ that contains all the points used to evaluate the sum in the previous equation is called *sparse grid* (Figure 4.2.2).

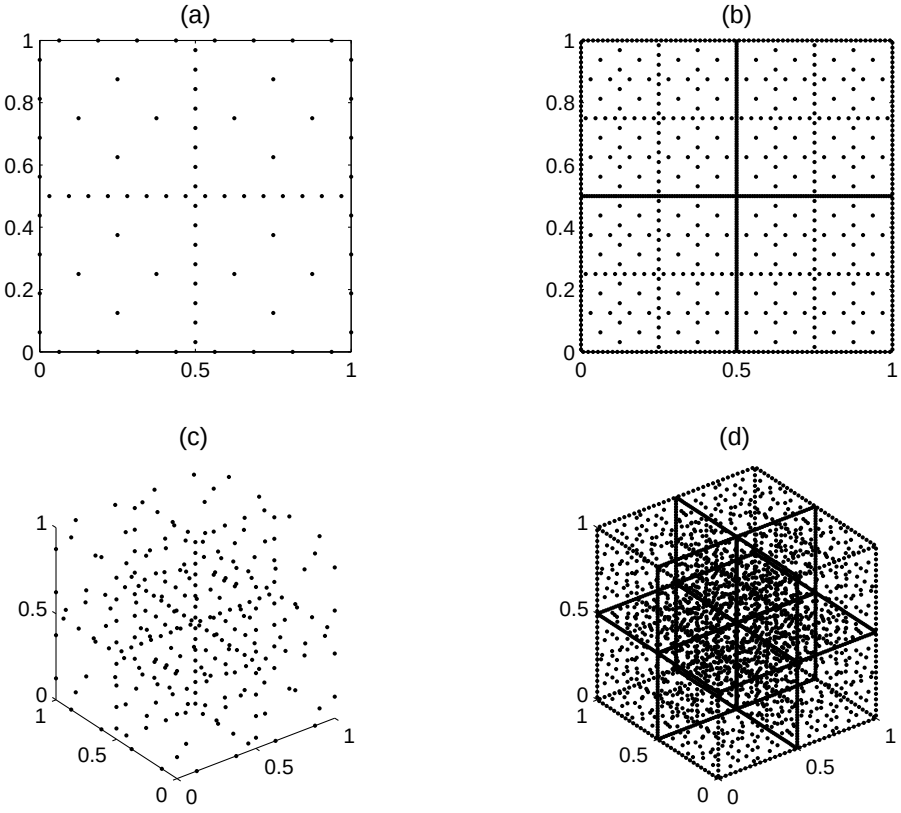


Figure 3.3: *Sparse grids based on Gauss-Chebyshev-Lobatto point sets. (a) Two-dimensional sparse grid of level $l = 3$. (b) Two-dimensional sparse grid of level $l = 5$. (c) Three-dimensional sparse grid of level $l = 3$. (d) Three-dimensional sparse grid of level $l = 5$.*

It is proven in Barthelmann et al. (2000, Proposition 2) that the Smolyak formula interpolates the data on the sparse grid $\mathcal{X}$ provided that the one-dimensional formulæ that are tensorized use *nested* point sets $\mathcal{X}^i$ and interpolate data on these sets. So the use of the Smolyak algorithm in conjonction with GCL point sets is often recommended. According to Novak and Ritter (1999), the required number of model evaluations is then given by:

$$N_l \sim \frac{2^{l-1}}{(l-1)!} M^{l-1} \quad , \quad M \to \infty \tag{3.55}$$

which is small compared to the exponential growth associated with the tensor-product scheme. The number of evaluations may be further reduced using an *adaptive anisotropic* sparse grid scheme (Gerstner and Griebel, 2003; Nobile et al., 2006; Klimke, 2006; Ganapathysubramanian and Zabaras, 2007).

However, if several families of nested point sets are available for random variables $\{X_i, i = 1, \ldots, M\}$ with *bounded support* (*e.g.* Gauss-Chebyshev-Lobatto, Gauss-Kronrod-Patterson (Elhay and Kautsky, 1992)), this is not the case for unbounded variables. A solution to handle unbounded variables (*e.g.* Gaussian variables) is to perform tensor-product Lagrange interpolation on *hyperrectangles* rather than hypercubes, using appropriate Gauss point sets. This is achieved in Bieri and Schwab (2009) in an adaptive fashion, in such a way that the most significant input random variables are favored.

4.2.3 Post-processing of the metamodel

Let us consider the interpolating polynomial $\mathcal{M}_N$ in Eq.(3.52). Provided that a sufficient number N of interpolation points has been used, the metamodel $\mathcal{M}_N(\boldsymbol{X})$ should be a fair approximation of the model response $\mathcal{M}(\boldsymbol{X})$. Then the statistical moments of the latter can be estimated by post-processing the surrogate $\mathcal{M}_N(\boldsymbol{X})$.

For instance, the mean of the model response can be approximated as follows:

$$\mathbb{E}\left[\mathcal{M}(\boldsymbol{X})\right] \;\approx\; \mathbb{E}\left[\mathcal{M}_N(\boldsymbol{X})\right] \;=\; \sum_{i=1}^{N} \mathcal{M}(\boldsymbol{x}^{(i)})\, \mathbb{E}\left[\mathcal{L}_i(\boldsymbol{X})\right] \tag{3.56}$$

We have seen that the use of a suitable point set $\mathcal{X} \equiv \{\boldsymbol{x}^{(i)}, i = 1, \ldots, N\}^{\mathsf{T}}$, such as a Gauss or a Gauss-Lobatto point set, is recommended in order to avoid the Runge effect and to improve the convergence of the interpolating polynomial toward function $\mathcal{M}$. These point sets correspond to *quadrature nodes* in numerical integration (see the brief presentation in Chapter 2, Section 2.1). By definition, the mathematical expectations $\mathbb{E}\left[\mathcal{L}_i(\boldsymbol{X})\right]$ are the associated *quadrature weights*:

$$w^{(i)} \;\equiv\; \int_{\mathcal{D}_{\boldsymbol{X}}} \mathcal{L}_i(\boldsymbol{x})\; f_{\boldsymbol{X}}(\boldsymbol{x})\; d\boldsymbol{x} \;\equiv\; \mathbb{E}\left[\mathcal{L}_i(\boldsymbol{X})\right] \tag{3.57}$$

These weights are available as tabulated values in the literature, see *e.g.* Abramowitz and Stegun (1970). So the mean estimate in Eq.(3.56) reduces to:

$$\mathbb{E}\left[\mathcal{M}_N(\boldsymbol{X})\right] \;=\; \sum_{i=1}^{N} \mathcal{M}(\boldsymbol{x}^{(i)})\; w^{(i)} \tag{3.58}$$

Two methods may be employed in order to estimate the r-th statistical moment $\mathbb{E}\left[\mathcal{M}^r(\boldsymbol{X})\right]$, $r > 1$. First, one may evaluate the metamodel $\mathcal{M}_N$ at a large set of points $\boldsymbol{x}^{(i)}$ randomly drawn from the PDF $f_{\boldsymbol{X}}(\boldsymbol{x})$ of $\boldsymbol{X}$, and then compute the empirical r-th moment of the produced output

sample. An estimation of the r-th moment of the metamodel (and *not* of the model) with an arbitrary accuracy can be obtained when increasing the size of the sample.

The other approach seems more attractive since it is based on an analytical formula such as in Eq.(3.58). It relies upon the interpolation of the function $\mathcal{M}^r(\cdot)$ rather than $\mathcal{M}(\cdot)$. The trick consists in reusing the point set $\mathcal{X}$ to this end, such that no additional (possibly costly) model evaluation is performed. Let us denote by $\mathcal{M}_N^r$ the interpolating polynomial, which reads:

$$\mathcal{M}_N^r(\boldsymbol{X}) \;=\; \sum_{i=1}^{N} \mathcal{M}^r(\boldsymbol{x}^{(i)})\; \mathcal{L}_i(\boldsymbol{X}) \tag{3.59}$$

Then taking the expectation of this metamodel provides the following estimate of the r-th statistical moment of the model response:

$$\mathbb{E}\left[\mathcal{M}_N^r(\boldsymbol{X})\right] \;=\; \sum_{i=1}^{N} \mathcal{M}^r(\boldsymbol{x}^{(i)})\; w_i \tag{3.60}$$

However, the accuracy of the moment estimate directly depends on the interpolation error of the function $\mathcal{M}^r$. This error will tend to increase with r, since the "degree" of $\mathcal{M}^r$ will be accordingly augmented by r, leading to a decrease of the smoothness of the function. Then more interpolation points should be employed.

Note that quantities of interest in sensitivity analysis, such as sensitivity indices, cannot be carried out analytically from the stochastic collocation metamodel. Such a post-processing requires the statistical treatment of a large sample of realizations of the metamodel $\mathcal{M}_N(\boldsymbol{X})$. In the following, we investigate *non-interpolating* approaches which provide approximations on an *explicit* functional basis. It will be shown that such a formulation is well suited to straightforward post-processing.

4.3 Spectral projection method

In the sequel, one considers directly the following PC expansion of the model response:

$$Y \;=\; \mathcal{M}(\boldsymbol{X}) \quad \approx \quad \mathcal{M}_p(\boldsymbol{X}) \;\equiv\; \sum_{|\boldsymbol{\alpha}|\leq p} a_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{3.61}$$

The *spectral projection method* aims at estimating the PC coefficients $a_{\boldsymbol{\alpha}}$ by exploiting the orthonormality of the truncated basis $\{\psi_{\boldsymbol{\alpha}}, |\boldsymbol{\alpha}| \leq p\}$. Indeed, by premultiplying the series in Eq.(3.61) by $\psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$ and by taking its expectation, one gets the theoretical expression of each coefficient $a_{\boldsymbol{\alpha}}$:

$$a_{\boldsymbol{\alpha}} \;=\; \mathbb{E}\left[\mathcal{M}(\boldsymbol{X})\; \psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\right] \;\equiv\; \int_{\mathcal{D}_{\boldsymbol{X}}} \mathcal{M}(\boldsymbol{x})\, \psi_{\boldsymbol{\alpha}}(\boldsymbol{x})\, f_{\boldsymbol{X}}(\boldsymbol{x})\, d\boldsymbol{x} \tag{3.62}$$

In practice, it is necessary to estimate the above mathematical expectation using numerical integration techniques, which approximate (3.62) by a weighted sum:

$$a_{\boldsymbol{\alpha}} \;\approx\; \sum_{i=1}^{N} w^{(i)}\; \mathcal{M}(\boldsymbol{x}^{(i)})\; \psi_{\boldsymbol{\alpha}}(\boldsymbol{x}^{(i)}) \tag{3.63}$$

Various approximation schemes may be employed, which differ in the choice of the integration *weights* $w^{(i)}$ and *nodes* $\boldsymbol{x}^{(i)}$. Two categories of techniques are proposed in the sequel, namely the *simulation* and the *quadrature* schemes.

4.3.1 Simulation technique

The *simulation* technique relies upon the choice of N *random* integration nodes $\boldsymbol{x}^{(i)}$ and integration weights defined by $w^{(1)} = \cdots = w^{(N)} = 1/N$, which leads to:

$$a_{\boldsymbol{\alpha}} \approx \widetilde{a}_{\boldsymbol{\alpha}} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{M}(\boldsymbol{x}^{(i)})\, \psi_{\boldsymbol{\alpha}}(\boldsymbol{x}^{(i)}) \tag{3.64}$$

The accuracy of the estimates $\widetilde{a}_{\boldsymbol{\alpha}}$ depends on the choice of the *sampling scheme* of the nodes $\boldsymbol{x}^{(i)}$. Three sampling methods are reviewed below.

Monte Carlo simulation

Monte Carlo (MC) simulation corresponds to the choice of *pseudo-random* nodes $\boldsymbol{x}^{(i)}$ with respect to the PDF $f_{\boldsymbol{X}}(\boldsymbol{x})$ of the input random vector $\boldsymbol{X}$. The performance of the method follows from elementary statistical derivations on Eq.(3.64). In this respect, $\widetilde{a}_{\boldsymbol{\alpha}}$ is regarded as a random variable due to the randomness in the generation of the $\boldsymbol{x}^{(i)}$'s. It appears that $\widetilde{a}_{\boldsymbol{\alpha}}$ is an estimate of $a_{\boldsymbol{\alpha}}$ whose mean and variance are respectively given by:

$$\mathbb{E}\left[\widetilde{a}_{\boldsymbol{\alpha}}^{MC}\right] = a_{\boldsymbol{\alpha}} \tag{3.65}$$

$$\mathbb{V}\left[\widetilde{a}_{\boldsymbol{\alpha}}^{MC}\right] = \frac{\sigma^2}{N} \quad , \quad \sigma^2 \equiv \mathbb{V}\left[\mathcal{M}(\boldsymbol{X})\psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\right] \tag{3.66}$$

hence a standard error $\varepsilon \equiv \sqrt{\mathbb{V}\left[\widetilde{a}_{\boldsymbol{\alpha}}^{MC}\right]}$ that decreases in $N^{-1/2}$. This induces a particularly low convergence rate, which is a well-known drawback of Monte Carlo simulation. Note furthermore that the convergence is all the slower as the total degree p of the PC expansion is high, since the variance σ^2 increases with the total degree $|\boldsymbol{\alpha}|$ of $\psi_{\boldsymbol{\alpha}}$. As a consequence more efficient simulation schemes have been investigated.

Latin Hypercube Sampling

The Latin Hypercube Sampling (LHS) method (McKay et al., 1979) aims at generating pseudo-random numbers that are more representative of the joint distribution of the input random vector $\boldsymbol{X}$ than those generated by MC simulation. LHS processes as follows:

1. Generate N realizations $\{\boldsymbol{e}^{(1)}, \ldots, \boldsymbol{e}^{(N)}\}$ of a uniform random vector over $[0,1]^M$.

2. Define the vectors $\boldsymbol{u}^{(i)} \equiv (\boldsymbol{e}^{(i)} - i)/N$, $i = 1, \ldots, N$: each component $u_j^{(i)}$ of $\boldsymbol{u}^{(i)}$ is located in the interval $[(i-1)N\ ,\ iN]$.

3. Randomly pair without replacement the N components of vector $\boldsymbol{u}^{(1)}$ with those of vector $\boldsymbol{u}^{(2)}$. The resulting N pairs are then randomly combined with the N components of $\boldsymbol{u}^{(3)}$, and so on until a set of N M-dimensional samples is formed.

4. The obtained set is finally transformed into a set of pseudo-random numbers $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$ that are distributed according to the input joint PDF $f_{\boldsymbol{X}}(\boldsymbol{x})$.

The LHS technique is explained in the simple case of two independent uniform random variables over $[-1, 1]$. As shown in Figure 3.4, the domain $[-1, 1]^2$ is first split into N^2 equiprobable cells. Then N points are randomly generated in such a way that there is only one point in each column and in each row.

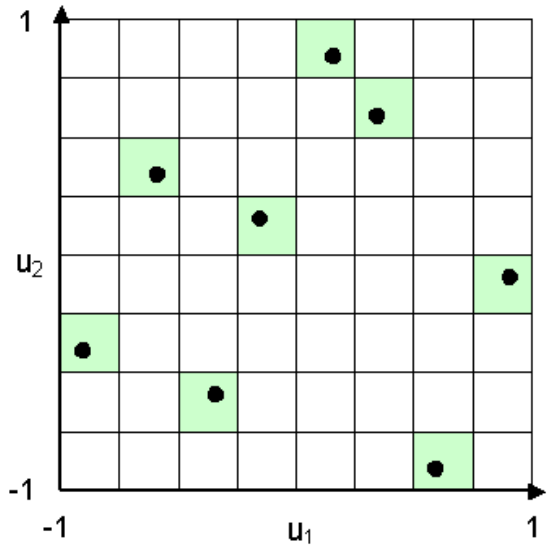


Figure 3.4: *Latin hypercube sample*

LHS is expected to perform much better than MC simulation in case of a quasi linear function $\boldsymbol{x} \mapsto \mathcal{M}(\boldsymbol{x})\psi_{\boldsymbol{\alpha}}(\boldsymbol{x})$. Indeed, it is shown in Owen (1992) that using the LHS integration nodes $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$ for the evaluation of the sum in Eq.(3.64) provides estimates $\widetilde{a}_{\boldsymbol{\alpha}}^{LHS}$ that satisfy:

$$\mathbb{E}\left[\widetilde{a}_{\boldsymbol{\alpha}}^{LHS}\right] \ = \ a_{\boldsymbol{\alpha}} \qquad \text{(unbiasedness)} \tag{3.67}$$

$$\mathbb{V}\left[\widetilde{a}_{\boldsymbol{\alpha}}^{LHS}\right] \ = \ \frac{1}{N}\left(\sigma^2 \ - \ \sigma_{add}^2\right) \ + \ o\left(\frac{1}{N}\right) \tag{3.68}$$

where σ_{add}^2 denotes the variance of the additive part closest to $\boldsymbol{x} \mapsto \mathcal{M}(\boldsymbol{x})\psi_{\boldsymbol{\alpha}}(\boldsymbol{x})$ in a mean square sense. Anyway the error associated with LHS is never much worse than the one corresponding to MC simulation since (Owen, 1992):

$$\mathbb{V}\left[\widetilde{a}_{\boldsymbol{\alpha}}^{LHS}\right] \ \leq \ \frac{N}{N-1}\,\mathbb{V}\left[\widetilde{a}_{\boldsymbol{\alpha}}^{MC}\right] \tag{3.69}$$

In other words LHS using N nodes is never worse than MC simulation using $N-1$ nodes. LHS has been used in a stochastic analysis framework for computing the PC coefficients, see *e.g.* Ghiocel and Ghanem (2002). However, LHS suffers from a major difficulty. Indeed, the accuracy of LHS-based estimates cannot be increased incrementally, *i.e.* by adding new points to the already existing LHS sample, since the new set is not a Latin hypercube anymore.

Quasi-Monte Carlo method

The convergence rate of the estimates can be often increased by the use of *deterministic* sequences known as *quasi-random* or *low discrepancy sequences* (Niederreiter, 1992). Such sequences have been used in Blatman et al. (2007) for computing the PC coefficients. In this work we focus on the so-called *Sobol' sequence* which generally reveals efficient to estimate high-dimensional (say $M \geq 10$) integrals (Morokoff and Caflisch, 1995).

The Sobol' sequence in one dimension is generated by expanding the set of integers $\{1, 2, \ldots, N\}$ into base 2 notation. The i-th term of the sequence is defined by:

$$u^{(i)} \;=\; \frac{b_0}{2} \,+\, \frac{b_1}{2^2} \,+\, \cdots \,+\, \frac{b_m}{2^{m+1}} \tag{3.70}$$

where the b_k's are integers taken from the base 2 expansion of the number $i-1$, that is:

$$[i-1]_2 \;=\; b_m \, b_{m-1} \, \cdots \, b_1 \, b_0 \tag{3.71}$$

with $b_k \in \{0, 1\}$. Figure 3.5 shows the space-filling process of $[0, 1]$ using this technique. The M-dimensional Sobol' sequence is built by pairing M permutations of the unidimensional sequences. Figure 3.6 shows the space-filling process of $[0, 1]^2$ using a two-dimensional Sobol' sequence, compared to MCS and LHS, from which the better uniformity of the former is obvious.

Let us denote by $(\boldsymbol{u}^1, \ldots, \boldsymbol{u}^N)$ the obtained set of N quasi-random numbers. It is necessary to transform the latter into realizations of the input random vector $\boldsymbol{X}$. Here the input random variables $\{X_1, \ldots, X_M\}$ are assumed to be independent for the sake of simplicity. Thus one applies the following change of variables componentwise:

$$x_j^{(i)} \;=\; F_X^{-1}\left[F_U(u_j^{(i)})\right] \quad , \quad i = 1, \ldots, N \quad , \quad j = 1, \ldots, M \tag{3.72}$$

where F_U is the cumulative density dunction of the uniform distribution over [0,1].

The estimates of the PC coefficients that are obtained by injecting the quasi-random integration nodes $\boldsymbol{x}^{(i)}$ in Eq.(3.64) are referred to as the *quasi-Monte Carlo (QMC) estimates* and are denoted by $\widetilde{a}_{\boldsymbol{\alpha}}^{QMC}$. Due to the deterministic nature of the quasi-random numbers it is not possible to derive statistical properties of the QMC estimates contrary to their MC and LHS counterparts. Instead a deterministic upper bound of the absolute error may be provided as shown in Morokoff and Caflisch (1995), which guarantees a *worst case* convergence rate in

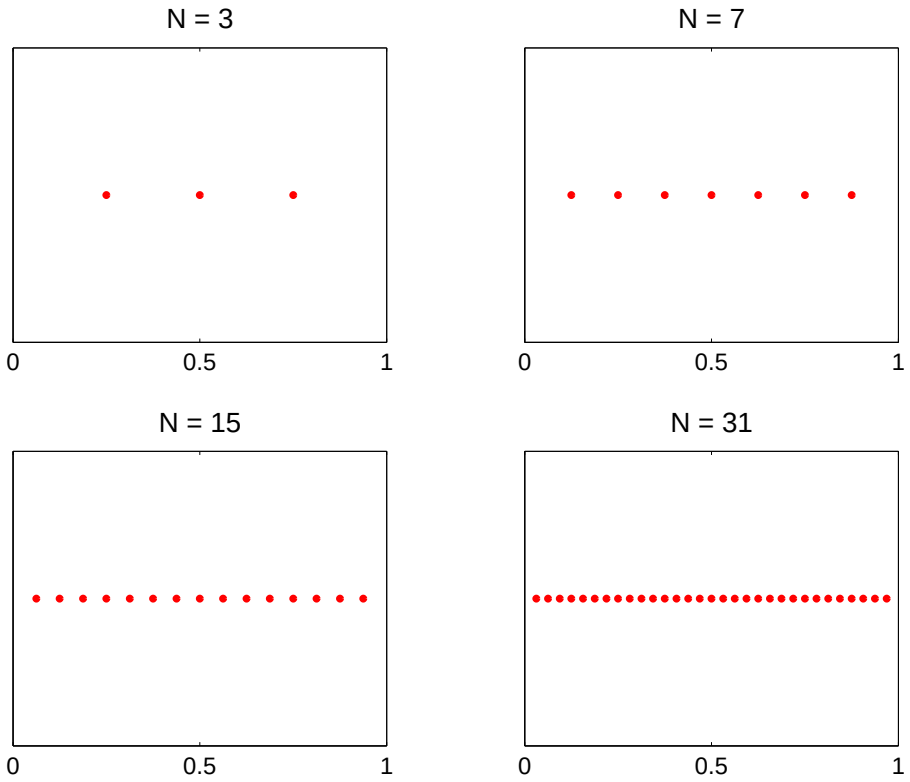


Figure 3.5: *Space-filling process of [0, 1] using a unidimensional Sobol' sequence*

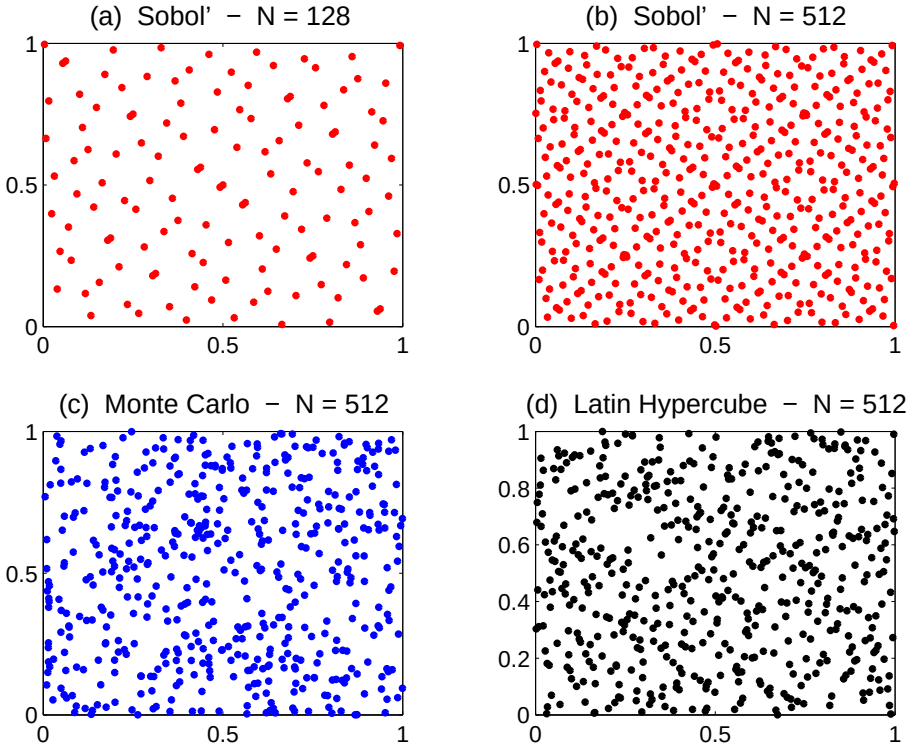


Figure 3.6: *Space-filling process of $[0, 1]^2$ using a two-dimensional Sobol' sequence, compared to MCS and LHS.*

$\mathcal{O}(N^{-1}\log^M(N))$. Thus for M small the convergence rate of QMC is faster than MC but for M large (say $M \geq 10$) the efficiency of QMC might be considerably reduced. However it has been shown in Caflisch et al. (1997) that QMC noticeably overperforms MC in practice for integrating high-dimensional functions of low *effective dimension*, *i.e.* which satisfy the two following statements:

- they only depend on a low number of input variables (effective dimension in the *superposition* sense);
- they only depend on low order interactions of input variables (effective dimension in the *truncation* sense).

The efficiency of the three sampling methods, namely Monte Carlo (MC), Latin Hypercube Sampling (LHS) and quasi-Monte Carlo (QMC), have been compared in Blatman et al. (2007). It has been shown that QMC overperforms MC and LHS, with a mean computational gain factor of 10 in order to reach a given accuracy.

4.3.2 Quadrature technique

An alternative to simulation techniques for selecting the integration nodes $\boldsymbol{x}^{(i)}$ and weights $w^{(i)}$ is *quadrature*, which has been briefly presented in Chapter 2, Section 2.1.2.

a) Tensor-product quadrature

The multidimensional integral in Eq.(3.62) may be approximated using the following tensor-product quadrature formula:

$$a_{\boldsymbol{\alpha}} \approx a_{\boldsymbol{\alpha}}^{n_1,\dots,n_M} \equiv \sum_{i_1=1}^{n_1} \cdots \sum_{i_M=1}^{n_M} \nu_1^{(i_1)} \cdots \nu_M^{(i_M)} \, \psi_{\boldsymbol{\alpha}}\left(x_1^{(i_1)},\dots,x_M^{(i_M)}\right) \, \mathcal{M}\left(x_1^{(i_1)},\dots,x_M^{(i_M)}\right) \tag{3.73}$$

One commonly uses an *isotropic* quadrature formula, *i.e.* a formula in which $n_1 = \cdots = n_M = n$. When using Gauss quadrature rules, this scheme allows one to integrate *exactly* any multivariate polynomial with *partial* degree not greater than $2n-1$. On the other hand, upon substituting the model function $\mathcal{M}(\boldsymbol{X})$ for its PC-based approximation $\mathcal{M}_p(\boldsymbol{X})$ in Eq.(3.62), one gets:

$$a_{\boldsymbol{\alpha}} \approx a_{\boldsymbol{\alpha},p} \equiv \int_{\mathcal{D}_{\boldsymbol{X}}} \mathcal{M}_p(\boldsymbol{x}) \, \psi_{\boldsymbol{\alpha}}(\boldsymbol{x}) \, f_{\boldsymbol{X}}(\boldsymbol{x}) \, d\boldsymbol{x} \tag{3.74}$$

The integrand in this equation is a multivariate polynomial of total degree $p+\|\boldsymbol{\alpha}\|_1$. As the multi-indices $\boldsymbol{\alpha}$ have a total degree less or equal to p, the maximal total degree of the integrands such as in Eq.(3.74) is $2p$. As a result, $a_{\boldsymbol{\alpha},p}$ may be computed exactly using an isotropic tensor-product

quadrature scheme based on a $(p+1)$-point Gauss quadrature rule, *i.e.* $a_{\boldsymbol{\alpha},p} = a_{\boldsymbol{\alpha}}^{p+1,\ldots,p+1}$. This requires to perform $N = (p+1)^M$ model evaluations. This exponential growth (known as the *curse of dimensionality*) may lead to intractable calculations in case of a computational demanding model function $\mathcal{M}$.

b) Smolyak sparse quadrature

An alternative to tensor-product quadrature that avoids the curse of dimensionality is of interest. Let us consider a univariate quadrature rule such that each rule of level $l \geq 1$ contains n_l points and allows one to integrate exactly any one-dimensional polynomial with degree at most m_l. Of interest is a multivariate quadrature formula made of linear combinations of product formulæ, referred to as the *Smolyak construction* (see Section 4.2.2). The Smolyak formula of level l applied to the estimation of the PC coefficients is given by:

$$a_{\boldsymbol{\alpha}} \approx a_{\boldsymbol{\alpha},p}^{Smolyak} \equiv \sum_{l \leq |\boldsymbol{k}| \leq l+M-1} (-1)^{l+M-|\boldsymbol{k}|-1} \binom{M-1}{|\boldsymbol{k}|-l} a_{\boldsymbol{\alpha}}^{k_1,\ldots,k_M} \tag{3.75}$$

In this expression, only products with a relatively small number of integration points are used. These points form a so-called *sparse grid*.

Let us denote by $\mathcal{P}_n^1$ the space of unidimensional polynomials of degree at most n. According to Novak and Ritter (1999, Lemma 1), a Smolyak formula of level l allows the exact integration of any polynomial belonging to the following space:

$$E(l,M) \equiv \sum_{\substack{|\boldsymbol{\alpha}|=l+M-1 \\ \alpha_1,\ldots,\alpha_M>0}} \left(\mathcal{P}_{m_{\alpha_1}}^1 \otimes \cdots \otimes \mathcal{P}_{m_{\alpha_M}}^1 \right) \tag{3.76}$$

Note that this result corresponds to a "non classical" polynomial space though. More interest is given to results for the classical space $\mathcal{P}_k^M$ of multivariate polynomials of total degree not greater than k.

As shown in Novak and Ritter (1999, Corollary 2), a Smolyak formula of level $l = p+1$ using classical grids (*e.g.* Gauss, CGL) integrates exactly any polynomial with total degree *at least* $2p+1$. This result may be directly applied to the estimation of the PC coefficients. Indeed, it is recalled that this calculation leads to integrate a polynomial of total degree $2p$ (Eq.(3.74)). The asymptotic required number of model evaluations is given by (Novak and Ritter, 1999, Corollary 2):

$$N \sim \frac{2^p}{p!} M^p \quad , \quad M \to \infty \tag{3.77}$$

Hence the computational cost only grows polynomially with M, which is much less than the exponential increase in n^M when using tensor-product quadrature.

4.4 Link between quadrature and stochastic collocation

The formalism developed in the previous section is really similar to the one used in the context of polynomial interpolation (Section 4.2). Indeed, the same concepts are exploited, namely orthogonal polynomials, Gauss point sets, tensor-products and Smolyak sparse grids. The present section is aimed at stressing the links between quadrature and interpolation, and consequently between the spectral projection and stochastic collocation methods. The following derivations are mainly inspired by Boyd (1989, Chapter 4).

4.4.1 Univariate case

We first consider a model $Y = \mathcal{M}(X)$ that only depends on a single random variable X with prescribed PDF $f_X(x)$ and support $\mathcal{D}_X$. Let $\mathcal{X} \equiv \{x^{(1)}, \ldots, x^{(n)}\}$ be a univariate experimental design, and let $\mathcal{Y} \equiv \{y^{(1)}, \ldots, y^{(n)}\}$ be the associated model evaluations. We denote by $\widetilde{\mathcal{M}}_n(X)$ the corresponding interpolating polynomial:

$$\widetilde{\mathcal{M}}_n(X) \equiv \sum_{i=1}^{n} \mathcal{M}(x^{(i)}) \, \ell_i(X) \tag{3.78}$$

where the ℓ_i's are univariate Lagrange polynomials. As $\widetilde{\mathcal{M}}_n$ is a polynomial of degree equal to $n-1$, it may be expanded onto a univariate PC basis as follows:

$$\widetilde{\mathcal{M}}_n(X) = \sum_{j=0}^{n-1} b_j \, \psi_j(X) \tag{3.79}$$

On the other hand, let us consider the following truncated PC decomposition of the model response $\mathcal{M}(X)$:

$$\mathcal{M}_{n-1}(X) \equiv \sum_{j=0}^{n-1} a_j \, \psi_j(X) \tag{3.80}$$

The PC coefficients a_j may be estimated by quadrature as shown in the previous section:

$$a_j \approx \widehat{a}_j \equiv \sum_{i=1}^{n} w^{(i)} \, \mathcal{M}(x^{(i)}) \, \psi_j(x^{(i)}) \qquad , \qquad j = 0, \ldots, n-1 \tag{3.81}$$

where the $w^{(i)}$'s and the $x^{(i)}$'s are the integration weights and points, respectively. Due to the interpolation property $\mathcal{M}(x^{(i)}) = \widetilde{\mathcal{M}}_n(x^{(i)})$, $i = 1, \ldots, N$, the previous equation rewrites:

$$\widehat{a}_j = \sum_{i=1}^{n} w^{(i)} \, \widetilde{\mathcal{M}}_n(x^{(i)}) \, \psi_j(x^{(i)}) \tag{3.82}$$

that is, according to Eq.(3.79):

$$\widehat{a}_j = \sum_{i=1}^{n} w^{(i)} \left(\sum_{k=0}^{n-1} b_k \, \psi_k(x^{(i)}) \right) \psi_i(x^{(i)}) \tag{3.83}$$

Upon permuting the two above summations, one gets:

$$\widehat{a}_j \;=\; \sum_{k=0}^{n-1} b_k \;\left(\sum_{i=1}^{n} w^{(i)}\; \psi_k(x^{(i)})\; \psi_j(x^{(i)}) \right) \tag{3.84}$$

The summands between brackets are polynomials with degree not greater than $2n-2$. If a n-point Gauss quadrature rule is used, then any polynomial of degree not greater than $2n-1$ can be exactly integrated. Then the second sum is equal to the following integral:

$$\int_{\mathcal{D}_X} \psi_k(x)\; \psi_j(x)\; f_X(x)\; dx \;\equiv\; \mathbb{E}\left[\psi_k(X)\; \psi_j(X)\right] \;=\; \delta_{j,k} \tag{3.85}$$

Hence Eq.(3.84) reduces to:

$$\widehat{a}_j \;=\; \sum_{k=0}^{n-1} b_k\; \delta_{j,k} \tag{3.86}$$

that is:

$$\widehat{a}_j \;=\; b_j \qquad , \qquad j = 0, \ldots, n-1 \tag{3.87}$$

Thus stochastic collocation in one-dimension in conjonction with a Gauss point set of size n yields the *same* polynomial metamodel than the quadrature scheme applied to a truncated PC expansion of degree $n-1$.

4.4.2 Multivariate case

a) Tensor-product interpolation

The previous derivations may be straightforwardly extended to the multivariate case using a tensor-product formulation. Let us consider an experimental design $\mathcal{X} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}^\mathsf{T}$ based on a Gauss tensor-product quadrature point set. One denotes by n the number of points in each dimension, so $N = n^M$. In this section we denote by $\widetilde{\mathcal{M}}_N(\boldsymbol{X})$ the corresponding interpolating polynomial:

$$\widetilde{\mathcal{M}}_N(\boldsymbol{X}) \;\equiv\; \sum_{i=1}^{N} \mathcal{M}(\boldsymbol{x}^{(i)})\; \mathcal{L}_i(\boldsymbol{X}) \tag{3.88}$$

where the $\mathcal{L}_i$'s are the multivariate tensor-product Lagrange polynomials introduced in Section 4.2. As $\widetilde{\mathcal{M}}_N$ is a polynomial of partial degree equal to $n-1$, it may be recast in a *full tensor-product* PC basis as follows:

$$\widetilde{\mathcal{M}}_N(\boldsymbol{X}) \;=\; \sum_{0 \leq \|\boldsymbol{\alpha}\|_\infty \leq n-1} b_{\boldsymbol{\alpha}}\; \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{3.89}$$

where $\|\boldsymbol{\alpha}\|_\infty \equiv \max_i \alpha_i$. Note that this index set differs from the traditional choice $\{0 \leq |\boldsymbol{\alpha}| \leq n\}$.

On the other hand, let us consider the following truncated full tensor-product PC decomposition of the model response $\mathcal{M}(\boldsymbol{X})$:

$$\mathcal{M}_{n-1}(\boldsymbol{X}) \equiv \sum_{0 \leq \|\boldsymbol{\alpha}\|_\infty \leq n-1} a_{\boldsymbol{\alpha}} \, \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{3.90}$$

The PC coefficients $a_{\boldsymbol{\alpha}}$ may be estimated by quadrature as shown in the previous section:

$$a_{\boldsymbol{\alpha}} \approx \widehat{a}_{\boldsymbol{\alpha}} \equiv \sum_{i=1}^{N} w^{(i)} \, \mathcal{M}(\boldsymbol{x}^{(i)}) \, \psi_{\boldsymbol{\alpha}}(\boldsymbol{x}^{(i)}) \tag{3.91}$$

where the $w^{(i)}$'s and the $\boldsymbol{x}^{(i)}$'s are the integration weights and points, respectively.

Similarly to the univariate case treated in the previous section, one gets the identity:

$$\widehat{a}_{\boldsymbol{\alpha}} = b_{\boldsymbol{\alpha}} \quad , \quad 0 \leq \|\boldsymbol{\alpha}\|_\infty \leq n-1 \tag{3.92}$$

provided that Gauss point sets are used.

It has to be noted that PC truncations over the whole hypercube $\{\|\boldsymbol{\alpha}\|_\infty \leq n-1\}$ (rather than the simplex $\{|\boldsymbol{\alpha}| \leq n-1\}$) should be used in order to fully take benefit of tensor-product quadrature. However, tensor-product quadrature is rarely used in practice due its prohibitive computational cost.

b) Interpolation on sparse grids

In order to avoid the curse of the dimensionality, it is recommanded to interpolate multivariate functions on a *sparse grid* (Barthelmann et al., 2000). In this setup, the model function may be interpolated using the following Smolyak formula of level l:

$$\widetilde{\mathcal{M}}_l^{Smolyak}(\boldsymbol{x}) \equiv \sum_{l \leq |\boldsymbol{k}| \leq l+M-1} (-1)^{l+M-|\boldsymbol{k}|-1} \binom{M-1}{|\boldsymbol{k}|-l} \widetilde{\mathcal{M}}_{n(k_1),\ldots,n(k_M)}(\boldsymbol{x}) \tag{3.93}$$

where $\widetilde{\mathcal{M}}_{n(k_1),\ldots,n(k_M)}(\boldsymbol{x})$ denotes the interpolating polynomial over the grid formed by the tensor-product of univariate integration points of levels $\{k_1, \ldots, k_M\}$.

Let $\mathcal{P}_n^1$ be the space of unidimensional polynomials of degree not greater than n. Suppose that each univariate grid of level l has n_l points. Let us define the following "non classical" polynomial space:

$$F(l, M) \equiv \sum_{\substack{|\boldsymbol{\alpha}|=l+M-1 \\ \alpha_1,\ldots,\alpha_M>0}} \left(\mathcal{P}^1_{n_{\alpha_1}-1} \otimes \cdots \otimes \mathcal{P}^1_{n_{\alpha_M}-1} \right) \tag{3.94}$$

As pointed out in Barthelmann et al. (2000, Remark 3), the interpolating polynomial $\widetilde{\mathcal{M}}_l^{Smolyak}(\boldsymbol{x})$ belongs to $F(l, M)$. Therefore one should use a PC decomposition of the form:

$$\widehat{\mathcal{M}}_{n-1}(\boldsymbol{X}) \equiv \sum_{\substack{|\boldsymbol{\alpha}| \leq n+M-1 \\ \alpha_1,\ldots,\alpha_M>0}} \left(\sum_{k_1=0}^{n_{\alpha_1}-1} \cdots \sum_{k_M=0}^{n_{\alpha_M}-1} a_{k_1 \cdots k_M} \; \psi_{k_1 \cdots k_M}(X_1, \ldots, X_M) \right) \tag{3.95}$$

in order to get the same metamodel as the one based on stochastic collocation. Notice that the interpolation property requires the use of *nested* point sets (see Section 4.2.2), which is *not* the case of Gauss points. Alternative nested rules may be used instead, such as Gauss-Chebyshev-Lobatto or Kronrod-Patterson, when available.

This link between stochastic collation and projection-based PC expansion shows tracks to a fruitful combination of advanced techniques for dimension-adaptive interpolation (Klimke, 2006) and quadrature (Gerstner and Griebel, 2003) with adaptive PC representations.

4.5 Regression method

4.5.1 Theoretical expression of the regression-based PC coefficients

The regression approach aims at computing the PC coefficients that minimize the mean-square error of approximation of the model response $Y = \mathcal{M}(\boldsymbol{X})$ by the PC metamodel (Berveiller et al., 2006; Choi et al., 2004). In the following we use the vector notation:

$$\boldsymbol{a} \;=\; \{a_{\boldsymbol{\alpha}_0}, \ldots, a_{\boldsymbol{\alpha}_{P-1}}\}^\mathsf{T} \tag{3.96}$$

$$\boldsymbol{\psi}(\boldsymbol{X}) \;=\; \{\psi_{\boldsymbol{\alpha}_0}(\boldsymbol{X}), \ldots, \psi_{\boldsymbol{\alpha}_{P-1}}(\boldsymbol{X})\}^\mathsf{T} \tag{3.97}$$

The regression problem may be cast as follows[2]:

$$\text{Find} \quad \widehat{\boldsymbol{a}} \quad \text{that minimizes} \quad \mathcal{J}(\boldsymbol{a}) \;\equiv\; \mathbb{E}\left[\left(\boldsymbol{a}^\mathsf{T}\boldsymbol{\psi}(\boldsymbol{X}) \;-\; \mathcal{M}(\boldsymbol{X})\right)^2\right] \tag{3.98}$$

The minimality condition $\frac{d\mathcal{J}}{d\boldsymbol{a}}(\widehat{\boldsymbol{a}}) = 0$ leads to:

$$\mathbb{E}\left[\boldsymbol{\psi}(\boldsymbol{X})\, \boldsymbol{\psi}^\mathsf{T}(\boldsymbol{X})\right]\, \widehat{\boldsymbol{a}} \;=\; \mathbb{E}\left[\boldsymbol{\psi}(\boldsymbol{X})\, \mathcal{M}(\boldsymbol{X})\right] \tag{3.99}$$

The mathematical expectation in the left hand side reduces to the identity matrix $\mathbf{1}$ since it is the correlation matrix of the random vector $\boldsymbol{\psi}(\boldsymbol{X})$ whose components are uncorrelated by definition. It follows that:

$$\widehat{\boldsymbol{a}} \;=\; \mathbb{E}\left[\boldsymbol{\psi}(\boldsymbol{X})\, \mathcal{M}(\boldsymbol{X})\right] \tag{3.100}$$

hence a formal equivalence with the projection-based coefficients $\widetilde{\boldsymbol{a}}$ in Eq.(3.62). In other words the theoretical projection coefficients minimize the mean-square error of approximation.

4.5.2 Estimates of the PC coefficients based on regression

The present section is aimed at proposing an alternative to projection estimates (based either on simulation or quadrature). In this purpose, let us consider a set of realizations $\mathcal{X} \equiv$

[2]Mathematically speaking, this is clearly the definition of the $\mathcal{L}^2$-projection. However, the current approach is referred to as *regression* to avoid confusion with the *spectral projection* method detailed in Section 4.3.

$\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}^\mathsf{T}$ of the input random vector $\boldsymbol{X}$. The empirical analogue of Eq.(3.99) reads:

$$\boldsymbol{\Psi}^\mathsf{T} \boldsymbol{\Psi}\, \widehat{\boldsymbol{a}} \;=\; \boldsymbol{\Psi}^\mathsf{T}\, \mathcal{Y} \tag{3.101}$$

with

$$\boldsymbol{\Psi} \;\equiv\; \begin{pmatrix} \psi_{\boldsymbol{\alpha}_0}(\boldsymbol{x}^{(1)}) & \cdots & \psi_{\boldsymbol{\alpha}_{P-1}}(\boldsymbol{x}^{(1)}) \\ \vdots & \ddots & \vdots \\ \psi_{\boldsymbol{\alpha}_0}(\boldsymbol{x}^{(N)}) & \cdots & \psi_{\boldsymbol{\alpha}_{P-1}}(\boldsymbol{x}^{(N)}) \end{pmatrix} \tag{3.102}$$

The matrix $\boldsymbol{\Psi}^\mathsf{T}\boldsymbol{\Psi}$ is called *information matrix*. We obtain the following *regression estimates* of the PC coefficients:

$$\widehat{\boldsymbol{a}} \;=\; \left(\boldsymbol{\Psi}^\mathsf{T}\boldsymbol{\Psi}\right)^{-1}\, \boldsymbol{\Psi}^\mathsf{T}\, \mathcal{Y} \tag{3.103}$$

Note that the projection estimates $\widetilde{\boldsymbol{a}}$ are sometimes called *quasi-regression estimates* (An and Owen, 2001) because they "ignore" the information matrix. It is clear that Eq.(3.103) is only valid for a full rank information matrix. A necessary condition is that the size N of the experimental design is not less than the number P of PC coefficients to estimate. In practice, it is not recommended to directly invert $\boldsymbol{\Psi}^\mathsf{T}\boldsymbol{\Psi}$ as in Eq.(3.103) since the solution may be particularly sensitive to an ill-conditioning of the matrix. The problem in Eq.(3.101) is rather solved using more robust numerical methods such as *singular value decomposition* (SVD) (Bjorck, 1996).

The experimental design $\mathcal{X}$ may be built using the sampling techniques introduced in Section 4.3.1, namely MC, LHS and QMC. Note that deterministic designs made of roots of Gauss quadrature points (Section 4.3) may also be employed (Isukapalli, 1999; Berveiller, 2005). A detailed statistical study of the regression coefficients estimates $\widehat{\boldsymbol{a}}^{MC}$ based on MC simulation may be found in Owen (1998). It appears that the $\widehat{a}^{MC}_{\boldsymbol{\alpha}}$'s are asymptotically unbiased:

$$\mathbb{E}\left[\widehat{a}^{MC}_{\boldsymbol{\alpha}}\right] \;=\; a_{\boldsymbol{\alpha}} \;-\; \frac{1}{N}\, \mathbb{E}\left[\psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\, S^2(\boldsymbol{X})\, \varepsilon(\boldsymbol{X})\right] \tag{3.104}$$

and have the following asymptotic variance:

$$\mathbb{V}\left[\widehat{a}^{MC}_{\boldsymbol{\alpha}}\right] \;\sim\; \frac{1}{N}\, \mathbb{E}\left[\psi^2_{\boldsymbol{\alpha}}(\boldsymbol{X})\, \varepsilon^2(\boldsymbol{X})\right] \quad , \quad N \to +\infty \tag{3.105}$$

where

$$S(\boldsymbol{X}) \;\equiv\; \boldsymbol{\psi}(\boldsymbol{X})^\mathsf{T}\, \boldsymbol{\psi}(\boldsymbol{X}) \tag{3.106}$$

and

$$\varepsilon(\boldsymbol{X}) \;\equiv\; \mathcal{M}(\boldsymbol{X}) \;-\; \boldsymbol{a}^\mathsf{T}\, \boldsymbol{\psi}(\boldsymbol{X}) \qquad \text{(remainder of the PC series)} \tag{3.107}$$

4.6 Discussion

Various non intrusive schemes have been described for estimating the PC coefficients. Stochastic collocation and quadrature are attractive methods since they rely upon well-established mathematical results in order to select optimal points in the experimental design. As noted in Xiu (2009), quadrature should be preferred to stochastic collocation in practice since:

- the computational manipulation of multivariate Lagrange polynomials is cumbersome;
- quadrature is based on an explicit representation in a PC basis, which allows straightforward post-processing (*e.g.* second moments and sensitivity indices).

The original tensor-product quadrature approach generally suffers the curse of dimensionality since the required number of model evaluations is given by $N = n^M$. To bypass this issue, one rather uses Smolyak quadrature which leads to the following computational cost in high dimensions:

$$N \sim \frac{2^p}{p!} M^p \quad , \quad M \to \infty \tag{3.108}$$

On the other hand, an asymptotic equivalent of the number of terms in a PC expansion of degree p is obtained by:

$$P = \binom{M+p}{p} \sim \frac{1}{p!} M^p \quad , \quad M \to \infty \tag{3.109}$$

Hence the ratio N/P tends to the factor 2^p for large M. The Smolyak construction has been labelled *optimal* in Novak and Ritter (1999) insofar as this quantity does not depend on M. However the computational cost may be important if a great accuracy of the PC expansion (*i.e.* a large p) is required.

Regression appears to be a relevant approach in order to reduce the number of model evaluations. Indeed, many studies show that a number of model evaluations given by $N = kP$ with $k = 2, 3$ often provides satisfactory results. In particular, good empirical results have been obtained in Berveiller et al. (2006); Berveiller (2005) in the context of non intrusive stochastic finite elements. A limitation of the method lies in the problem of selecting the points in the experimental design though. It is worth mentioning that an algorithm has been devised in Sudret (2008) to select a minimum number of roots of orthogonal polynomials in the design. Random designs may be also employed, which allows the derivation of statistical properties of the PC coefficients estimators (Owen, 1998). Thus it is shown that *regression should outperform simulation* provided that a sufficiently accurate PC expansion (*i.e.* a large enough degree p) has been chosen.

Indeed, even if both the regression and simulation estimates converge at the (slow) rate $N^{-1/2}$, their associated constants are respectively given by $\mathbb{E}\left[\psi^2_{\boldsymbol{\alpha}}(\boldsymbol{X})\varepsilon^2(\boldsymbol{X})\right]$ and $\mathbb{E}\left[\psi^2_{\boldsymbol{\alpha}}(\boldsymbol{X})\,\mathcal{M}^2(\boldsymbol{X})\right]$, where $\varepsilon(\boldsymbol{X})$ denotes the remainder of the PC series. Thus the regression estimates have typically a much smaller error than their simulation counterparts provided that the remainder $\varepsilon(\boldsymbol{X})$ is "small". Moreover, just as for simulation, the convergence rate of the regression estimates should be noticeably improved by using a more efficient sampling scheme than Monte Carlo. As a consequence, a special focus will be given to the regression technique combined to specific sampling methods such as LHS and QMC in the sequel.

5 Post-processing of the PC coefficients

Let us consider a truncated PC expansion of degree p of the model response $Y \equiv \mathcal{M}(\boldsymbol{X})$:

$$\mathcal{M}_p(\boldsymbol{X}) \equiv \sum_{|\boldsymbol{\alpha}|\leq p} a_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{3.110}$$

Assume that the coefficients $a_{\boldsymbol{\alpha}}$ have been estimated using one of the non intrusive methods presented in the previous section. Denoting by $\widehat{a}_{\boldsymbol{\alpha}}$ the estimates of the coefficients, one gets the following PC approximation:

$$\widehat{\mathcal{M}}_p(\boldsymbol{X}) \equiv \sum_{|\boldsymbol{\alpha}|\leq p} \widehat{a}_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{3.111}$$

5.1 Statistical moment analysis

The statistical moments of the response PC expansion can be analytically derived from its coefficients. In particular, the mean and the variance respectively read:

$$\widehat{\mu}_{Y,p} \equiv \widehat{a}_{\boldsymbol{0}} \tag{3.112}$$

$$\widehat{\sigma}^2_{Y,p} \equiv \sum_{0<|\boldsymbol{\alpha}|\leq p} \widehat{a}^2_{\boldsymbol{\alpha}} \tag{3.113}$$

where $(\cdot)_{,p}$ recalls that a PC expansion of order p is used. The *skewness coefficient* of the response is defined by:

$$\delta_Y \equiv \frac{1}{\sigma_Y^3} \mathbb{E}\left[\left(Y-\mu_Y\right)^3\right] \tag{3.114}$$

Its PC-based approximation reads:

$$\widehat{\delta}_{Y,p} \equiv \frac{1}{\sigma^3_{Y,P}} \sum_{0<|\boldsymbol{\alpha}|,|\boldsymbol{\beta}|,|\boldsymbol{\delta}|\leq p} \widehat{a}_{\boldsymbol{\alpha}}\, \widehat{a}_{\boldsymbol{\beta}}\, \widehat{a}_{\boldsymbol{\delta}}\, \mathrm{E}\left[\psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\psi_{\boldsymbol{\beta}}(\boldsymbol{X})\psi_{\boldsymbol{\delta}}(\boldsymbol{X})\right] \tag{3.115}$$

Note that the expectations in the above equation are zero for many sets of multi-indices $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\delta})$ and can thus be efficiently stored in a sparse structure. If the $\psi_{\boldsymbol{\alpha}}$'s are exclusively products of Hermite polynomials, these expectations can be computed analytically (Malliavin, 1997). Otherwise a quadrature scheme can be used. Similarly the *kurtosis coefficient* of the response is given by:

$$\kappa_Y \equiv \frac{1}{\sigma_Y^4} \mathbb{E}\left[\left(Y-\mu_Y\right)^4\right] \tag{3.116}$$

and is approximated as follows:

$$\widehat{\kappa}_{Y,p} \equiv \frac{1}{\sigma^4_{Y,P}} \sum_{0<|\boldsymbol{\alpha}|,|\boldsymbol{\beta}|,|\boldsymbol{\delta}|,|\boldsymbol{\gamma}|\leq p} \widehat{a}_{\boldsymbol{\alpha}}\, \widehat{a}_{\boldsymbol{\beta}}\, \widehat{a}_{\boldsymbol{\gamma}} \widehat{a}_{\boldsymbol{\delta}} \quad \mathrm{E}\left[\psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\psi_{\boldsymbol{\beta}}(\boldsymbol{X})\psi_{\boldsymbol{\gamma}}(\boldsymbol{X})\psi_{\boldsymbol{\delta}}(\boldsymbol{X})\right] \tag{3.117}$$

Note that Eqs.(3.115),(3.117) may lead to computationally expensive calculations though if the number of terms P is high. As an alternative, it is possible to recast the quantities $\delta_{Y,p}$ and $\kappa_{Y,p}$ by substituting the model $\mathcal{M}$ by its PC approximation $\mathcal{M}_p$ into Eqs.(3.114),(3.116) as follows:

$$\widehat{\delta}_{Y,p} \;=\; \frac{1}{\sigma_Y^3}\,\mathbb{E}\left[\left(\mathcal{M}_p(\boldsymbol{X})-\mu_{Y,p}\right)^3\right] \tag{3.118}$$

$$\widehat{\kappa}_{Y,p} \;=\; \frac{1}{\sigma_Y^4}\,\mathbb{E}\left[\left(\mathcal{M}_p(\boldsymbol{X})-\mu_{Y,p}\right)^4\right] \tag{3.119}$$

These quantities may be evaluated by numerical integration techniques, such as a *quadrature* scheme.

5.2 Global sensitivity analysis

Let us define by $\mathcal{I}_{i_1,\ldots,i_s}$ the set of indices in $\{\boldsymbol{\alpha}\in\mathbb{N}^M : 0\leq|\boldsymbol{\alpha}|\leq p\}$ such that only the indices $\{i_1,\ldots,i_s\}$ are nonzero:

$$\mathcal{I}_{i_1,\ldots,i_s} \;=\; \left\{\; \boldsymbol{\alpha}\in\mathbb{N}^M : 0\leq|\boldsymbol{\alpha}|\leq p \;\;,\;\; \forall k\in\{1,\ldots,M\}\setminus\{i_1,\ldots,i_s\}\;,\; \alpha_k \;=\; 0 \;\right\} \tag{3.120}$$

The set $\mathcal{I}_{i_1,\ldots,i_s}$ corresponds to the polynomials $\psi_{\boldsymbol{\alpha}}$ depending on *all* the input parameters $\{X_{i_1},\ldots,X_{i_s}\}$ and *only* on them. Using this notation, it is possible to reorder the PC terms according to the variables they depend on:

$$\begin{aligned}\mathcal{M}_p(\boldsymbol{X}) \;=\; a_{\mathbf{0}} \;+\; \sum_{i=1}^{M}\sum_{\boldsymbol{\alpha}\in\mathcal{I}_i} a_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(X_i) \;+\; \sum_{1\leq i_1<i_2\leq M}\;\sum_{\boldsymbol{\alpha}\in\mathcal{I}_{i_1,i_2}} a_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(X_{i_1},X_{i_2}) \\ +\;\cdots\;+\; \sum_{1\leq i_1<\cdots<i_s\leq M}\;\sum_{\boldsymbol{\alpha}\in\mathcal{I}_{i_1,\ldots,i_s}} a_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(X_{i_1},\ldots,X_{i_s}) \;+\cdots+\; \sum_{\boldsymbol{\alpha}\in\mathcal{I}_{1,\ldots,M}} a_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\end{aligned} \tag{3.121}$$

In the above equation, the true dependence of each multivariate polynomial to each subset of input parameters has been indicated for the sake of clarity. A PC-based analogue of the ANOVA decomposition has been obtained.

Due to the orthonormality of the PC basis, the random summands on the right hand side of (3.121) satisfy the properties (2.29),(2.30) in Chapter 2, Section 2.4.2. It is therefore possible to identify each summand in (3.121) as follows:

$$\mathcal{M}_{i_1,\ldots,i_s}(X_{i_1},\ldots,X_{i_s}) \;=\; \sum_{\boldsymbol{\alpha}\in\mathcal{I}_{i_1,\ldots,i_s}} a_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(X_{i_1},\ldots,X_{i_s}) \tag{3.122}$$

It is now easy to derive sensitivity indices from the above representation (Sudret, 2008). These indices, called *PC-based sensitivity indices* are denoted by $S^p_{i_1,\ldots,i_s}$ and given by:

$$S^p_{i_1,\ldots,i_s} \;=\; \frac{1}{D_p}\sum_{\boldsymbol{\alpha}\in\mathcal{I}_{i_1,\ldots,i_s}} a_{\boldsymbol{\alpha}}^2 \tag{3.123}$$

Moreover, the *PC-based total sensitivity indices* are given by:

$$S_i^{T,p} \;=\; \frac{1}{D_p} \sum_{\boldsymbol{\alpha} \in \mathcal{I}_i^+} a_{\boldsymbol{\alpha}}^2 \tag{3.124}$$

where $\mathcal{I}_i^+$ denotes the set of all indices with a non zero i-th component, that is:

$$\mathcal{I}_i^+ \;\equiv\; \Big\{ \boldsymbol{\alpha} \in \mathbb{N}^M \;:\; 0 \leq |\boldsymbol{\alpha}| \leq p \;\;,\; \alpha_i \neq 0 \Big\} \tag{3.125}$$

5.3 Probability density function of response quantities and reliability analysis

Post-processing based on an intensive simulation of the metamodel $\widehat{\mathcal{M}}_p$ is now considered. First, in order to obtain a graphical representation of the response PDF, the truncated PC expansion may be simulated using Monte Carlo sampling at a negligible cost. This yields a sample set of response quantities, say $\{y^{(i)} \equiv \widehat{\mathcal{M}}_p(\boldsymbol{x}^{(i)}), i = 1, \ldots, N_K\}$. From this set, a kernel representation may be built as follows (Chapter 2, Section 2.2):

$$\hat{f}_Y(y) = \frac{1}{N_K\, h_K} \sum_{i=1}^{N_K} K\left(\frac{y - \widehat{\mathcal{M}}_p(\boldsymbol{x}^{(i)})}{h_K} \right) \tag{3.126}$$

where it is recalled that $K(x)$ is a suitable positive function called kernel, and h_K is the bandwith parameter.

On the other hand, reliability analysis may be carried out at a low computational cost. Let us consider a *limit state function* $g(\boldsymbol{X}, \boldsymbol{X}') \equiv g(\mathcal{M}(\boldsymbol{X}), \boldsymbol{X}')$ which depends on the input random variables $\boldsymbol{X}$ through the model $\mathcal{M}$ and on other random variables $\boldsymbol{X'}$ (see Chapter 2, Section 2.3). The physical system under consideration is assumed to be safe (resp. to fail) if $g(\boldsymbol{x}, \boldsymbol{x}') > 0$ (resp. $g(\boldsymbol{x}, \boldsymbol{x}') \leq 0$). The associated probability of failure reads:

$$P_f \;=\; \int_{\mathcal{D}_{\boldsymbol{X},\boldsymbol{X}'}} \mathbf{1}_{g(\boldsymbol{x},\boldsymbol{x}')\leq 0}(\boldsymbol{x}, \boldsymbol{x}')\; f_{\boldsymbol{X},\boldsymbol{X}'}(\boldsymbol{x}, \boldsymbol{x}')\; d\boldsymbol{x} d\boldsymbol{x}' \tag{3.127}$$

Upon substituting the model response $\mathcal{M}(\boldsymbol{X})$ for its PC representation $\widehat{\mathcal{M}}(\boldsymbol{X})$ into $g(\boldsymbol{X}, \boldsymbol{X}')$, one gets the following *analytical* approximation:

$$g(\boldsymbol{X}, \boldsymbol{X}') \quad \simeq \quad \widehat{g}(\boldsymbol{X}, \boldsymbol{X}') \quad = \quad g(\widehat{\mathcal{M}}(\boldsymbol{X}), \boldsymbol{X}') \tag{3.128}$$

Thus the probability of failure (3.127) may be inexpensively estimated by applying the classical reliability methods (*e.g.* crude Monte Carlo, FORM and importance sampling (Ditlevsen and Madsen, 1996)) to the response surface (3.128).

6 Conclusion

Spectral methods rely upon the expansion of the unknown random response of the model onto a specified basis. The scope of the present work has been restricted to bases made of multivariate

orthonormal polynomials, which are referred to as *polynomial chaos* (PC) representations. The mathematical formalism of the PC expansions has been introduced in the cases of independent and mutually dependent input random variables, as well as input random fields. Then the use of the PC representations in the so-called *Spectral Stochastic Finite Element Method* (SSFEM) has been outlined for a linear elliptic boundary value problem featuring random parameters. It has been shown that the method leads to a large system of coupled equations, hence a possible high computational cost. A recently developed strategy, namely the *generalized spectral decomposition* (GSD) method, has been developed in order to circumvent this difficulty. GSD is aimed at yielding an optimal approximation of the solution in a reduced basis. In this context, one has to solve a series of low-dimensional problems, which results in a significant diminution of the computational work compared to classical solving schemes. However both SSFEM and GSD require a specific modification of the existing deterministic computer code.

The so-called *non intrusive* methods are proposed in order to solve a wide class of stochastic problems by means of a set of calls to the deterministic model. Two approaches are investigated, namely an *interpolating* approach and a *non interpolating* approach. The former, known as *stochastic collocation*, is based on the polynomial interpolation of the model response. The latter focuses on the approximation of the PC coefficients. Two estimates are proposed:

- the *projection* estimates, which can be computed either by simulation or quadrature;
- the *regression* estimates, which are computed by ordinary least-square regression.

After a careful review of the convergence estimates and computational cost associated with each method, it appears that the regression-based estimates should be the most accurate for a given number of performed model evaluations. The *Smolyak sparse quadrature* may be a relevant alternative provided that the number of input random variables is not too large (say $M \leq 8$). Also, it has been shown that the quadrature schemes are equivalent to stochastic collocation when the interpolation points are selected as Gauss quadrature nodes. The "'non interpolating" point of view is preferred though since it provides a quite interpretable metamodel, allowing straightforward post-processing such as the computation of statistical moments and sensitivity indices.

Chapter 4

Adaptive sparse polynomial chaos approximations using stepwise regression

1 The curse of dimensionality

Various methods have been reviewed in the previous chapter for building up an approximation of the model reponse onto a polynomial chaos basis. Whatever the approach, may it be intrusive or non intrusive, the number of basis functions P may be prohibitively large when the number of input random variables M increases. Indeed, when truncating the PC expansion such that one only retains the basis polynomials of total degree not greater than p, the number of terms grows polynomially both with p and M:

$$P = \binom{M+p}{p} \tag{4.1}$$

In the classical SSFEM approach (Section 3.2), such an increase is particularly embarrassing since the methods leads to a coupled system of size $\mathcal{N} \times P$, where $\mathcal{N}$ denotes the number of degrees of freedom of the spatial discretization. The GSD scheme (Section 3.4) allows one to dramatically reduce this impact insofar as one has to solve several low-dimensional problems of size $m \times P$ (*updating* stage) where m is typically small, say $m \leq 10$ (Nouy, 2007a; Nouy and Le Maître, 2009). On the other hand, non intrusive dimension-adaptive interpolation schemes known as *stochastic collocation* have been proposed recently in Nobile et al. (2006); Klimke (2006); Ganapathysubramanian and Zabaras (2007) to build up a metamodel of the model response at a reduced computational cost.

In the present work, the regression-based PC approach has been retained among the various non intrusive methods since it is believed to be particularly efficient. Indeed, as shown in the

previous chapter, the number of model evaluations required by Smolyak quadrature for large M is:

$$N \approx 2^p P \quad , \quad P \approx \frac{M^p}{p!} \tag{4.2}$$

instead of only:

$$N \approx kP \quad , \quad k \in \{2, 3\} \tag{4.3}$$

for regression. As the present work is aimed at minimizing the computational cost N, a reduction of the dimensionality P of the PC basis is necessary.

To this end, alternative strategies for truncating the PC expansion are considered in Section 2. They are inspired by the so-called *sparsity-of-effects principle* (Montgomery, 2004), which states that most models are principally governed by main effects and low-order interactions. *Low-rank* truncation schemes that have been defined in An and Owen (2001); Todor and Schwab (2007) are first described. In this setting, one builds truncated PC representations by constraining not only its degree but also the maximum interaction order of the basis polynomials. Moreover, a new truncation scheme is proposed which retains in priority the basis terms associated with low-order interactions. The resulting truncated PC expansions are referred to as *hyperbolic*.

In order to further decrease the size P of the PC basis, one considers *sparse* PC expansions, *i.e.* which contain a low number of nonzero coefficients compared to the "full" metamodels. In the lack of knowledge of the model function, it is not possible to detect *a priori* the significant and the negligible terms in the PC expansion though. A solution consists in performing an incremental search of the significant terms. Such an algorithm requires reliable estimates for assessing the metamodels. Several error estimates are investigated in Section 3. Then an iterative procedure based on stepwise regression is developed in Section 4 in order to build up a sparse PC approximation.

It should be noted that a similar stepwise scheme was recently proposed in Choi et al. (2004), in which classical statistical tests are adopted as criteria for accepting or rejecting the candidate basis functions. It was assumed implicitly that the error of approximation of the response by the PC expansion is random conditionally to the input variables. As only deterministic models are of interest herein, selection criteria based on estimates of the deterministic model deviation have been preferred in the present work.

The analytical *Sobol' function* is used as an example through this chapter in order to illustrate the efficiency of the proposed polynomial chaos strategies. However the iterative algorithms and the truncation schemes have been devised and validated progressively on various application examples, see Blatman and Sudret (2008a,c, 2009d, 2008d, 2009b,a,c).

2 Strategies for truncating the polynomial chaos expansions

Let us consider the PC expansion of the model response:

$$Y \equiv \mathcal{M}(\boldsymbol{X}) = \sum_{\boldsymbol{\alpha}\in\mathbb{N}^M} a_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{4.4}$$

Any truncation strategy of the PC representation corresponds to a specific choice of a non empty finite set of multi-indices $\boldsymbol{\alpha} \equiv \{\alpha_1, \ldots, \alpha_M\}$ in $\mathbb{N}^M$. To this end one first introduces the following notation. The *length* of a multi-index $\boldsymbol{\alpha}$ is defined by:

$$|\boldsymbol{\alpha}| \equiv \|\boldsymbol{\alpha}\|_1 \equiv \alpha_1 + \cdots + \alpha_M \tag{4.5}$$

Moreover, the *rank* of $\boldsymbol{\alpha}$ is defined by:

$$\|\boldsymbol{\alpha}\|_0 \equiv \sum_{i=1}^{M} \mathbf{1}_{\{\alpha_i>0\}} \tag{4.6}$$

The PC expansion is commonly truncated by retaining the polynomials of total degree not greater than p, that is:

$$Y \approx \mathcal{M}_p(\boldsymbol{X}) \equiv \sum_{0\leq\|\boldsymbol{\alpha}\|_1\leq p} a_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{4.7}$$

The corresponding *index set* is defined by:

$$\mathcal{A}^{M,p} \equiv \left\{\boldsymbol{\alpha} \in \mathbb{N}^M : \|\boldsymbol{\alpha}\|_1 \leq p\right\} \tag{4.8}$$

and its cardinality is given by:

$$\text{card}\left(\mathcal{A}^{M,p}\right) = \binom{M+p}{p} \tag{4.9}$$

As a result the number of terms strongly increases with both the number of input random variables M and the PC degree p. As the number of model evaluations N has to be greater than P in a regression context, the usual truncation scheme may lead to intractable calculations in high dimensions. This motivates the investigation of other truncation strategies that would provide a more "optimal" PC decomposition, *i.e.* that would properly represent the model response by means of a lower number of terms. In this purpose, alternative types of index sets based on the so-called *sparsity-of-effects principle* are proposed in the sequel.

2.1 Low-rank index sets

2.1.1 Definition

Of interest are index sets inspired by the *sparsity-of-effects principle* (Montgomery, 2004), which states that most models are principally governed by main effects and low-order interactions.

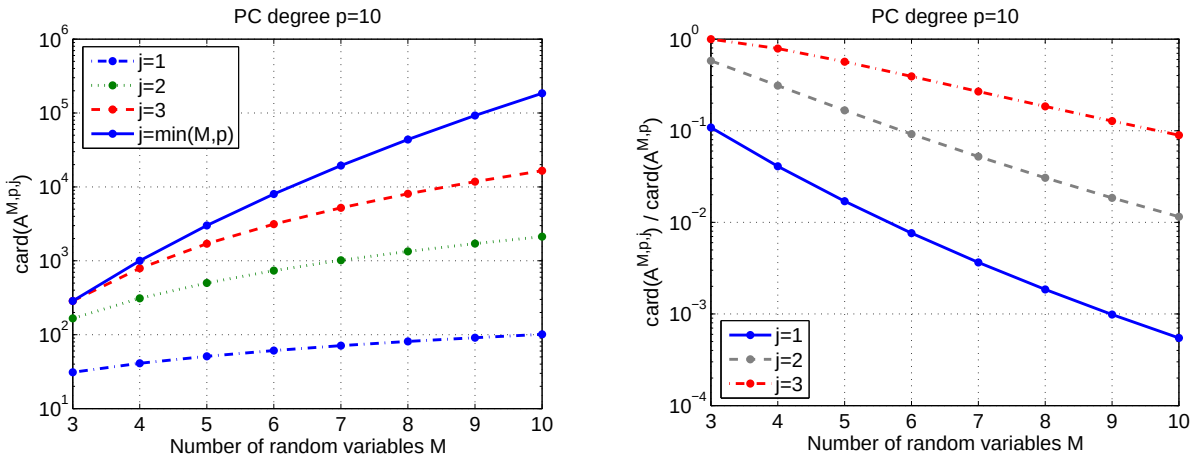


Figure 4.1: Cardinality of the index sets $\mathcal{A}^{M,p,j}$ with respect to the number of input random variables M. (Left) Cardinality of $\mathcal{A}^{M,p,j}$ compared to the cardinality of $\mathcal{A}^{M,p}$. (Right) Relative cardinality of $\mathcal{A}^{M,p,j}$.

Following An and Owen (2001); Todor and Schwab (2007), one proposes index sets with a prescribed maximum rank $j \leq p$:

$$\mathcal{A}^{M,p,j} \equiv \left\{\boldsymbol{\alpha} \in \mathbb{N}^M : \|\boldsymbol{\alpha}\|_1 \leq p\,,\ \|\boldsymbol{\alpha}\|_0 \leq j\right\} \tag{4.10}$$

As low values will be typically chosen for j (say $j = 2, 3$), such index sets will be referred to as *low-rank index sets*. Accordingly one defines the following *low-rank PC expansions*:

$$\mathcal{M}_{\mathcal{A}^{M,p,j}}(\boldsymbol{X}) \equiv \sum_{\boldsymbol{\alpha}\in\mathcal{A}^{M,p,j}} a_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{4.11}$$

The evolution of the cardinality of $\mathcal{A}^{M,p,j}$ with respect to the number of input random variables M for various values of j and $p = 10$ is depicted in Figure 4.1. It is shown that incrementing j results in an increase up to one order of magnitude of the number of terms when $M = 10$. Also, it appears that decreasing j allows a relative decrease factor of about 10 with respect to the usual index set $\mathcal{A}^{M,p}$. Exactly the same conclusions could be drawn when varying the PC degree p for fixed j. Indeed, the cardinality of the index sets satisfies the following symmetry property:

$$\text{card}\left(\mathcal{A}^{M,p,j}\right) = \text{card}\left(\mathcal{A}^{p,M,j}\right) \tag{4.12}$$

In other words, the sensitivity of the cardinality of the index sets with respect to M and p is exactly the same.

2.1.2 Numerical example

Let us consider now the so-called *Sobol' function* (Sobol', 2003):

$$Y \equiv \mathcal{M}(\boldsymbol{X}) = \prod_{i=1}^{M} \frac{|4X_i - 2| + c_i}{1 + c_i} \tag{4.13}$$

where the input variables $X_i, i = 1, \ldots, M$ are uniformly distributed over $[0, 1]$ and c_i are non negative constants. The sensitivity indices of Y can be derived analytically. For numerical application, the number of input variables M is set equal to 8 and one selects $\boldsymbol{c} = \{1, 2, 5, 10, 20, 50, 100, 500\}^{\mathsf{T}}$. Several projections of the Sobol' function are depicted in Figure 4.2. It appears from visual inspection that the sensitivity of the function to its input parameters decrease as the index i of variable x_i increases. In addition, the probability density function of Y is obtained using a kernel method (Chapter 2, Section 2.2) and is plotted in Figure 4.3.

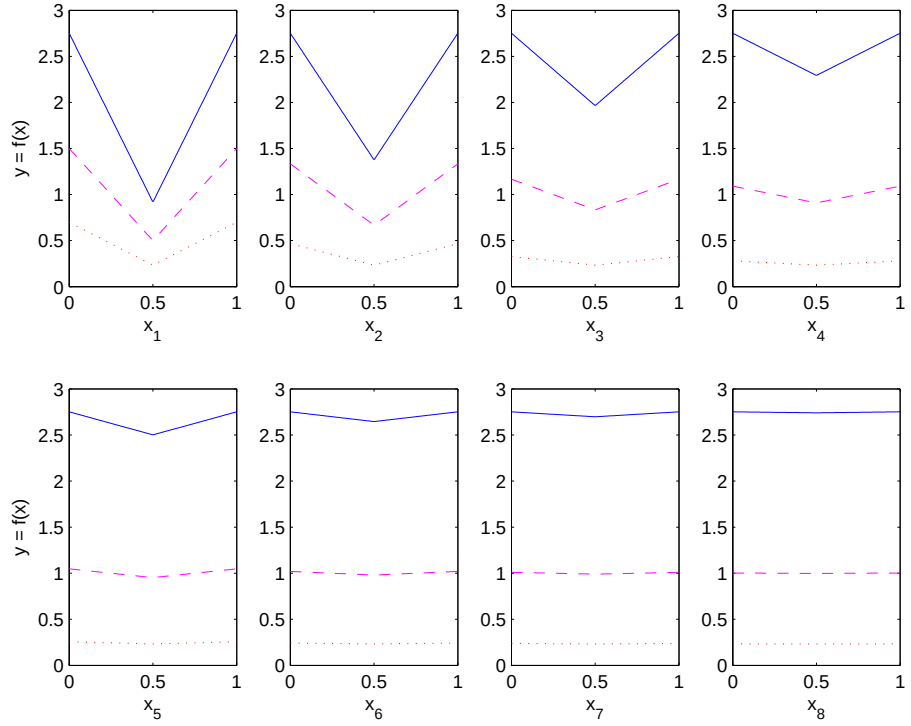


Figure 4.2: *Sobol' function - Several projections of the function. Solid (resp. dotted and dashed) lines correspond to fixed variables that are set equal to 0 (resp.* 0.5 *and* 0.75*).*

The model response Y is approximated by low-rank Legendre PC expansions $\mathcal{M}_{\mathcal{A}^{M,p,j}}$ with $j = 1, \ldots, 4$ and $p = 4, 5, \ldots$. The coefficients of the PC expansions are computed by least-square regression (Eq.(3.103)) using an experimental design based on a Sobol' quasi-random sequence. The size of the design N is chosen such that $N = 2\ \mathrm{card}(\mathcal{A}^{M,p,j})$. The accuracy of the approximations is assessed by the following empirical relative $\mathcal{L}^2$-error:

$$\hat{\varepsilon} \equiv \frac{\sum_{i=1}^{\mathcal{N}} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}_{\mathcal{A}^{M,p,j}}(\boldsymbol{x}^{(i)}) \right)^2}{\sum_{i=1}^{\mathcal{N}} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \bar{y} \right)^2} \quad , \qquad \mathcal{N} = 50,000 \tag{4.14}$$

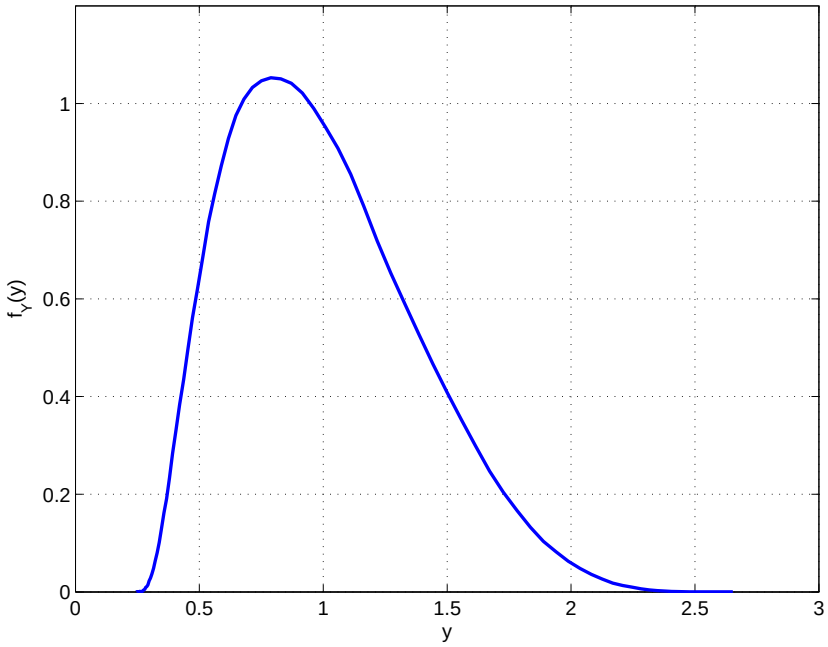


Figure 4.3: *Sobol' function - Probability density function obtained by a kernel method*

where the $\boldsymbol{x}^{(i)}$'s are random realizations of the input random vector $\boldsymbol{X}$, and $\bar{y}$ is defined by:

$$\bar{y} \equiv \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \mathcal{M}(\boldsymbol{x}^{(i)}) \tag{4.15}$$

The convergence of the various PC approximations is shown in Figure 4.4. It is observed that for $j = 1$ the relative error of approximation cannot decrease below 0.06. This is due to a too severe truncation of the PC representation (no interaction effect is considered). For $j \geq 2$, it appears that the convergence is all the faster since the rank of the index sets j is low. For instance, the metamodel related to $j = 2$ provides a relative error less than 10^{-3} using about $1,700$ model evaluations (PC degree $p = 6$), whereas such an accuracy is not reached using more than $5,000$ simulations when $j \geq 3$.

In this example, a low-rank index set of the form $\mathcal{A}^{M,p,2}$ (*i.e.* main effects and interactions of order 2) is sufficient to properly represent the model response, allowing a computation of a small number of coefficients at a low computational cost compared to the usual PC associated with $\mathcal{A}^{M,p}$. In the lack of further information, the maximum rank j will be set equal to 2 as a default value.

2.1.3 Limitation

A limitation of the low-rank index sets lies in the fact that for a fixed rank $j < \min(M, p)$, the associated PC metamodel $\mathcal{M}_{\mathcal{A}^{M,p,j}}$ will *not* converge to the true model response since the iteractions of order strictly greater than j will be missing. Mathematically speaking, for $j <$

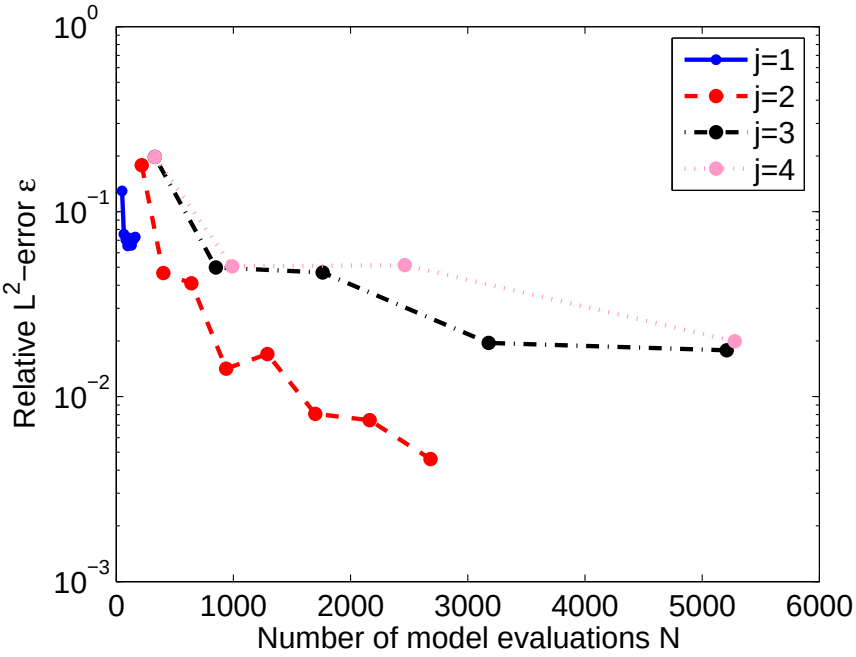


Figure 4.4: *Sobol' function - Convergence rates of the PC approximations based on low-order index sets* $\mathcal{A}^{M,p,j}$ *with respect to the degree* p *and for various values of* j *(for each* p*, the number of model evaluations is given by* $N = 2\ card(\mathcal{A}^{M,p,j})$*)*

$\min(M,p)$, the sequence of nested sets $(\mathcal{A}^{M,p,j})_{p\in\mathbb{N}}$ does not converge to $\mathbb{N}^M$ but to $\mathcal{A}^{M,\infty,j}$, where:

$$\mathcal{A}^{M,\infty,j} \equiv \{\boldsymbol{\alpha} \in \mathbb{N}^M : \|\boldsymbol{\alpha}\|_0 \leq j\} \tag{4.16}$$

This may be a problem in case of the choice of a too low rank j, as shown in the previous numerical example when selecting $j = 1$. This problem has been alleviated in Bieri and Schwab (2009) in the "intrusive" context of a stochastic elliptic boundary value problem such as in Section 3.2.1. In that work, two strategies based on regularity statements are designed in order to automatically tune the parameters p and j, leading to an optimal index set.

It is not possible to develop such a strategy in our non intrusive context though since the model function is unknown and is not supposed to have an *a priori* prescribed regularity. Instead one proposes low cardinality new index sets which now tend to $\mathbb{N}^M$ when $p \to \infty$.

2.2 Hyperbolic index sets (Blatman and Sudret, 2009a)

2.2.1 Isotropic hyperbolic index sets

One proposes the use of the following index sets based on q-norms[1], $0 < q < 1$:

$$\mathcal{A}_q^{M,p} \equiv \{\boldsymbol{\alpha} \in \mathbb{N}^M : \|\boldsymbol{\alpha}\|_q \leq p\} \tag{4.17}$$

where:

$$\|\boldsymbol{\alpha}\|_q \equiv \left(\sum_{i=1}^{M} \alpha_i^q\right)^{1/q} \tag{4.18}$$

Such norms penalize the high-rank indices all the more since q is low, as shown in Figure 4.5 for a two-dimensional case. Note that setting q equal to 1 corresponds to the usual truncation scheme, *i.e.* $\mathcal{A}_1^{M,p} = \mathcal{A}^{M,p}$. When using $q < 1$, the retained basis polynomials are located under an hyperbola, hence the name *hyperbolic index sets*. For instance, the retained basis terms in PC expansions of varying degree p are plotted in Figure 4.6 for $q = 1$, $q = 0.75$ and $q = 0.5$. Note that whatever the choice of the value of q, the sequence of nested sets $(\mathcal{A}_q^{M,p})_{p\in\mathbb{N}}$ always converges to the set $\mathbb{N}^M$. Thus the associated PC approximations will necessarily converge to the model response in the $\mathcal{L}^2$-norm.

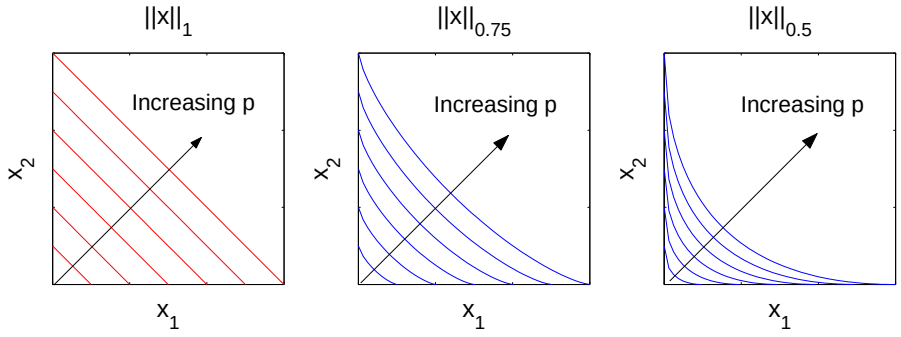


Figure 4.5: *Principle of the truncation strategy based on q-norms* ($0 < q \leq 1$)

The evolution of the cardinality of $\mathcal{A}_q^{M,p}$ with respect to the number of input random variables M for various values of q and $p = 10$ is depicted in Figure 4.7. It is shown that decreasing q results in a significant reduction of the number of terms compared to the usual truncated PC expansion (*i.e.* $q = 1$). On the other hand, the evolution of $\text{card}(\mathcal{A}_q^{M,p})$ with respect to the PC degree p for various values of q and $M = 10$ is depicted in Figure 4.8.

[1]Note that Eq.(4.18) actually defines a *quasi-norm* rather than a norm, since the triangular inequality is violated for $q < 1$. However, as this has no incidence on the following derivations, the label "norm" will be used throughout this report.

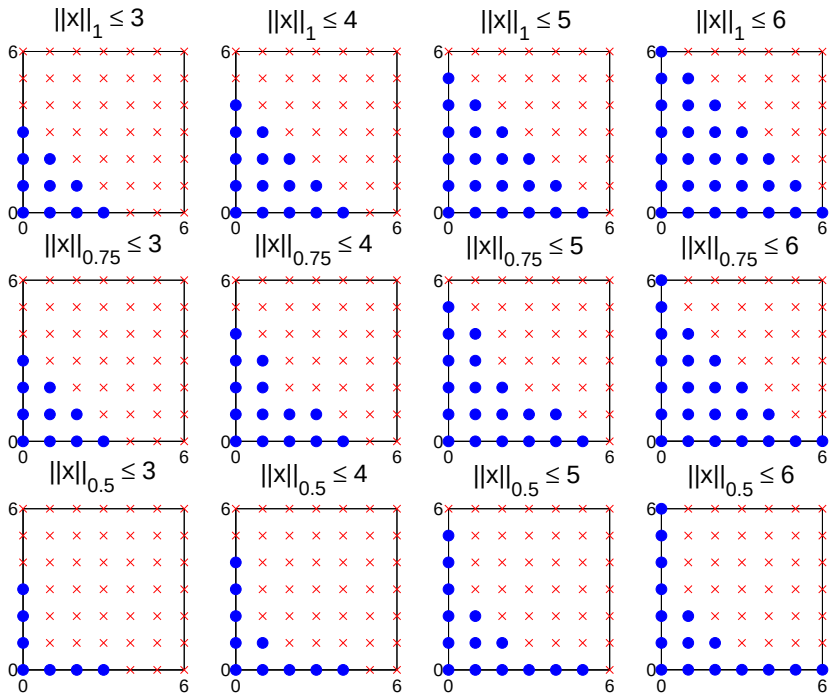


Figure 4.6: *Retained basis terms in the polynomial chaos expansion when varying the parameter q of the index sets $\mathcal{A}_q^{M,p}$ and the total degree $p = 3, 4, 5, 6$, for $M = 2$. The x-axes (resp. y-axes) correspond to the partial degree of the polynomials in X_1 (resp. X_2).*

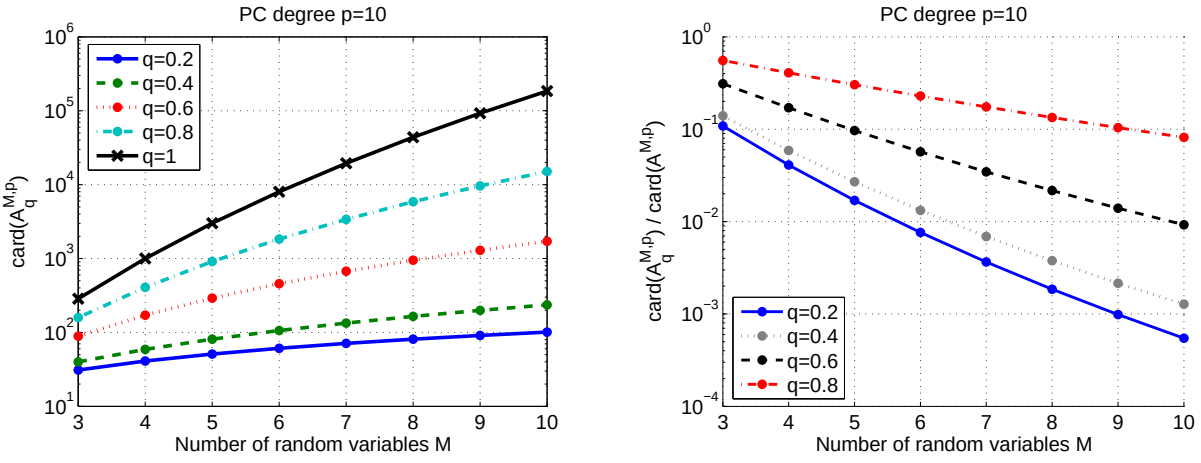


Figure 4.7: *Cardinality of the index sets $\mathcal{A}_q^{M,p}$ with respect to the number of input random variables M. (Left) Cardinality of $\mathcal{A}_q^{M,p}$ compared to the cardinality of $\mathcal{A}^{M,p}$. (Right) Relative cardinality of $\mathcal{A}_q^{M,p}$.*

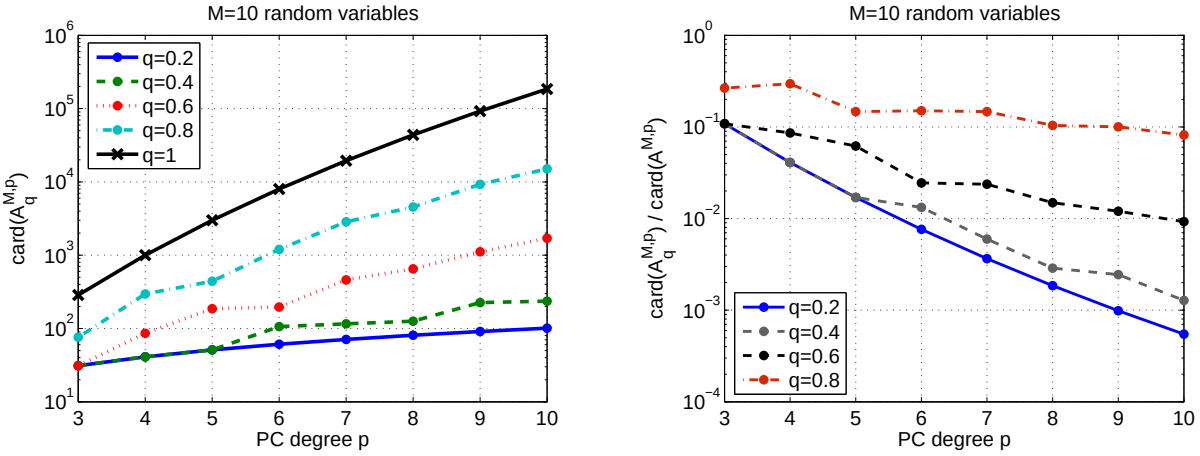


Figure 4.8: *Cardinality of the index sets $\mathcal{A}_q^{M,p}$ with respect to the PC degree p. (Left) Cardinality of $\mathcal{A}_q^{M,p}$ compared to the cardinality of $\mathcal{A}^{M,p}$. (Right) Relative cardinality of $\mathcal{A}_q^{M,p}$.*

Numerical example

Let us consider the Sobol' function already studied in Section 2.1.2. The model response is approximated by truncated Legendre PC expansions associated with hyperbolic index sets $\mathcal{A}_q^{M,p}$ for various values of p and q. It is recalled that Sobol' quasi-random experimental designs of size $N = 2\ \mathrm{card}(\mathcal{A}_q^{M,p})$ are used to estimate the PC coefficients. The convergence of the various approximations with respect to the PC degree p are compared in Figure 4.9. It is shown that setting q equal to 0.2 does not converge when increasing the PC degree p, with a relative error greater than 5%. The best convergence rate is obtained for $q = 0.4$, with a relative error of 1% (resp. 0.7%) using about 900 (resp. $1,200$) simulations. The convergence degradates when increasing q. The usual PC approximation (*i.e.* $q = 1$) appears to be the least efficient.

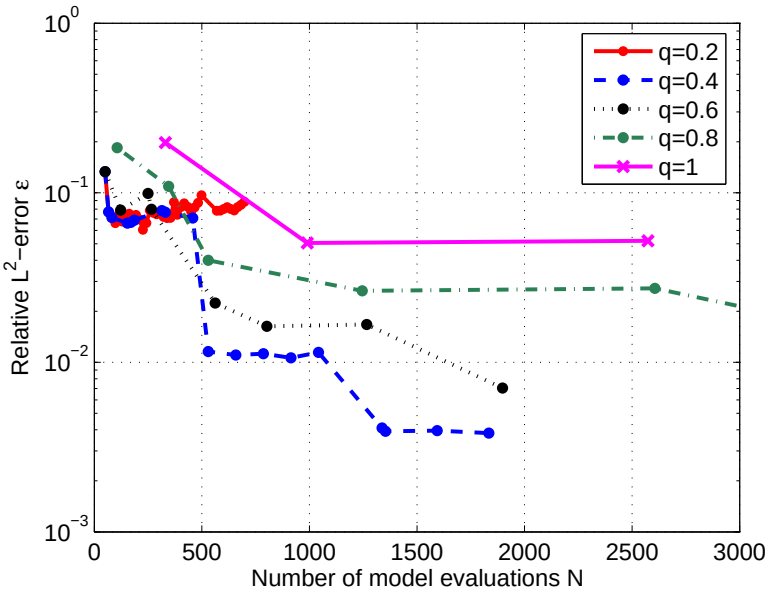


Figure 4.9: *Sobol' function - Convergence rates of the PC approximations based on hyperbolic index sets* $\mathcal{A}_q^{M,p}$ *with respect to the maximum length p and for various values of q (for each p, the number of model evaluations is given by* $N = 2\ card(\mathcal{A}_q^{M,p})$*)*

This example shows that the convergence rate of the PC approximations can be dramatically improved by a suitable choice of the parameter q of the hyperbolic truncation set. The computational cost may be further reduced by taking into account the fact that the input variables might have a different impact on the model response, as shown from global sensitivity analysis (Sudret, 2008) (see Chapter 2, Section 2.4). This is the scope of the next section.

2.2.2 Anisotropic hyperbolic index sets

The present section is focused on a truncation strategy which favors those input random variables X_i's with large *total sensitivity indices* S_i^T's. To this end one considers a strategy for truncating

the PC expansions that is based on the following *anisotropic* norm:

$$\|\boldsymbol{z}\|_{q,\boldsymbol{w}} = \left(\sum_{i=1}^{M} |w_i z_i|^q\right)^{1/q} \quad , \forall\, \boldsymbol{z} \equiv \{z_1, \ldots, z_M\}^\mathsf{T} \in \mathbb{R}^M \quad , \; w_i \geq 1 \tag{4.19}$$

This allows one to define *anisotropic index sets* by:

$$\mathcal{A}_{q,\boldsymbol{w}}^{M,p} \equiv \{\boldsymbol{\alpha} \in \mathbb{N}^M \, : \, \|\boldsymbol{\alpha}\|_{q,\boldsymbol{w}} \leq p\} \tag{4.20}$$

The retained basis terms in the PC expansion for several sets of weights $\boldsymbol{w}$ and total degrees p are depicted in Figure 4.7 in a two-dimensional example.

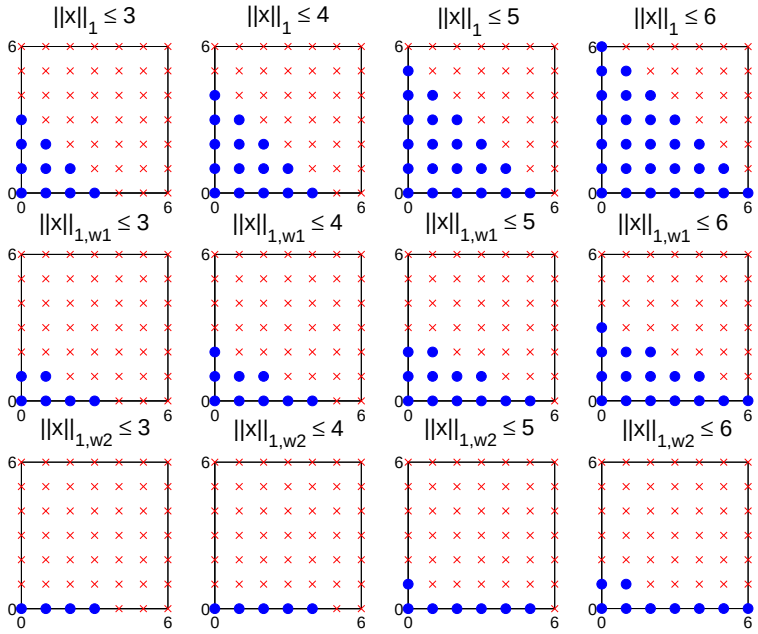


Figure 4.10: *Anisotropic index sets - Retained basis terms in the polynomial chaos expansion of the model response when setting the weights* $\boldsymbol{w}$ *equal to* $\boldsymbol{w}_0 = \{1,1\}, \boldsymbol{w}_1 = \{1,2\}$ *and* $\boldsymbol{w}_2 = \{1,5\}$*, and varying the total degree* $p = 3,4,5,6$ *(the* $q = 1$*-norm is considered)*

In this work, an heuristic definition of the weights based on the total sensitivity indices is considered, such that:

- the weight w_k increases when the total sensitivity index S_k^T of the input random variable X_k is small;
- denoting $j \equiv \arg\max_k(S_k^T)$, one gets $w_j = 1$. This ensures that the index set $\mathcal{A}_{q,\boldsymbol{w}}^{M,p}$ contains at least one p-degree term, namely $\{0, \ldots, 0, p, 0, \ldots, 0\}$. Thus p in the notation $\mathcal{A}_{q,\boldsymbol{w}}^{M,p}$ corresponds to the maximum degree of the polynomial in the most important input parameter.

According to these requirements, the following definition is proposed:

$$w_i \equiv 1 + K \frac{S_{max}^T - S_i^T}{\sum_{k=1}^M S_k^T} \quad , \quad i = 1, \ldots, M \tag{4.21}$$

where $S^T_{max} \equiv \max_k S^T_k$ and K is a non-negative constant. A high value of K leads to a great anisotropy of the PC index set, whereas $K = 0$ corresponds to the isotropic case. In the lack of further investigation, the value of K will be set equal to 1 in the following.

The proposed anisotropic strategy may lead to a noticeable reduction of the number of terms in the PC expansion of the model response and then to a strong decrease of the required number of model evaluations. Nonetheless it is clear that the total sensitivity indices have to be computed *a priori* in order to define a suitable set of weights such as in Eq.(4.21). To this end one may use approximations of the sensitivity indices based on low-order PC expansions. Such an approach will be investigated in Section 4.

3 Error estimates of the polynomial chaos approximations

3.1 Generalization error and empirical error

Let us consider an experimental design $\mathcal{X} = \{\boldsymbol{x}^{(1)}, ..., \boldsymbol{x}^{(N)}\}^\mathsf{T}$. Let $\mathcal{Y} = \{y^{(i)} \equiv \mathcal{M}(\boldsymbol{x}^{(i)}), i = 1, \ldots, N\}^\mathsf{T}$ be the vector of the corresponding model evaluations. As shown in Chapter 3, Section 4.5, one may use this data in order to compute the following PC approximation by regression:

$$\widehat{\mathcal{M}}_{\mathcal{A}}(\boldsymbol{X}) \equiv \sum_{\boldsymbol{\alpha} \in \mathcal{A}} \hat{a}_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{4.22}$$

where $\mathcal{A}$ is a finite non empty subset of $\mathbb{N}^M$. In this work, we focus on the following approximation error in the $\mathcal{L}^2$-norm:

$$Err \equiv \mathbb{E}\left[\left(\mathcal{M}(\boldsymbol{X}) - \widehat{\mathcal{M}}_{\mathcal{A}}(\boldsymbol{X})\right)^2\right] \tag{4.23}$$

The quantity Err is sometimes referred to as the *generalization error* in statistical learning (Vapnik, 1995). In practice, Err may be estimated by the *empirical error* (or *training error*) defined by:

$$Err_{emp} \equiv \frac{1}{N} \sum_{i=1}^{N} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \widehat{\mathcal{M}}_{\mathcal{A}}(\boldsymbol{x}^{(i)})\right)^2 \tag{4.24}$$

where the $\boldsymbol{x}^{(i)}$'s are the points of the experimental design $\mathcal{X}$. The *relative* training error is defined by:

$$\varepsilon_{emp} \equiv \frac{Err_{emp}}{\hat{\mathbb{V}}[\mathcal{Y}]} \tag{4.25}$$

where $\hat{\mathbb{V}}[\mathcal{Y}]$ denotes the empirical variance of the response sample $\mathcal{Y}$:

$$\hat{\mathbb{V}}[\mathcal{Y}] \equiv \frac{1}{N-1} \sum_{i=1}^{N} \left(y^{(i)} - \bar{\mathcal{Y}}\right)^2 \quad , \quad \bar{\mathcal{Y}} \equiv \frac{1}{N} \sum_{i=1}^{N} y^{(i)} \tag{4.26}$$

Of common use is the corresponding *coefficient of determination* R^2 that reads:

$$R^2 \equiv 1 - \varepsilon_{emp} \tag{4.27}$$

However it is well-known that Err_{emp} underpredicts the generalization error. The quantity Err_{emp} is even systematically reduced by increasing the complexity of the PC approximation (*i.e.* the cardinality of $\mathcal{A}$), whereas Err may increase. This is the so-called *overfitting* phenomenon. Note that the so-called *Runge effect* described in Chapter 3, Section 4.2.1 may be regarded as an extreme case of overfitting. Let us recall the definition of the Runge function:

$$f(x) \equiv \frac{1}{1+25\,x^2} \quad , \quad \forall x \in [-1,1] \tag{4.28}$$

One selects an experimental design $\mathcal{X}$ made of equally spaced points in the interval $[-1,1]$. One tries to approximate the Runge function by a polynomial of degree $p = N-1$, which is a classical polynomial interpolation problem. The Runge function is plotted in Figure 4.11 together with the interpolating polynomial.

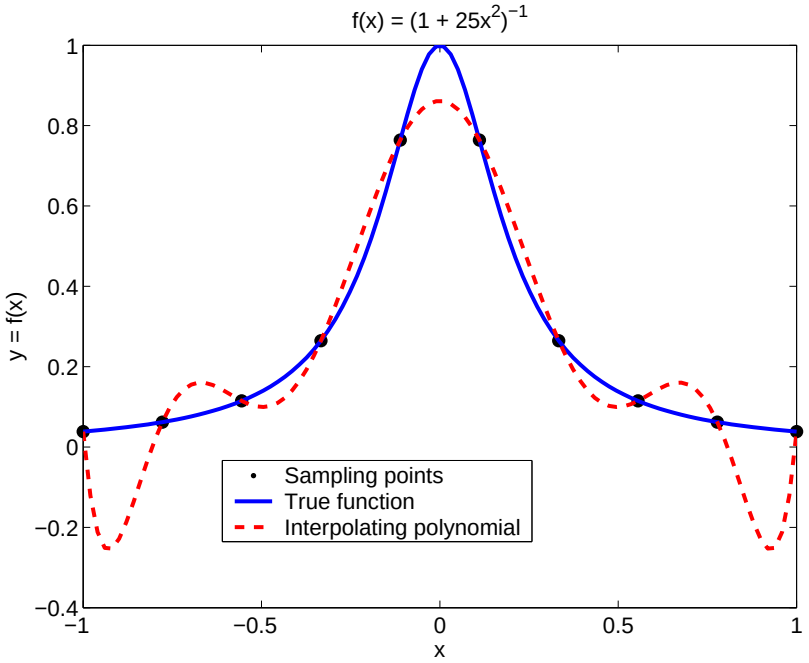


Figure 4.11: *Runge function - Overfitting phenomenon:* $Err_{emp} = 0$ *(i.e.* $R^2 = 1$*) in spite of a large generalization error*

It appears that the polynomial strongly oscillates toward the ends of the variation domain. In this example the training error Err_{emp} would be zero in spite of a poor approximation. As a consequence, an error estimate which is known to be much less sensitive to overfitting than Err_{emp} is investigated in the next section.

3.2 Leave-one-out error

The cross-validation technique (Stone, 1974; Geisser, 1975) consists in dividing the data sample into two subsamples. A metamodel is built from one subsample, *i.e.* the *training set*, and its performance is assessed by comparing its predictions to the other subset, *i.e.* the *test set*. A refinement of the method is the *ν-fold cross-validation*, in which the observations are randomly assigned to one of ν partitions of nearly equal size. The learning set contains in turn all but one of the partitions which is considered as the test set. The generalization error is estimated for each of the ν sets and then averaged over ν.

The *leave-one-out* cross-validation corresponds to the special case of ν-fold cross-validation with $\nu = N$. A comparative study reported in Molinaro et al. (2005) indicates that this technique generally performs very well in terms of estimation bias and mean-square error. Let us denote by $\widehat{\mathcal{M}}_{\mathcal{A}}^{(-i)}$ the metamodel that has been built from the experimental design $\mathcal{X} \setminus \{\boldsymbol{x}^{(i)}\}$, *i.e.* when removing the i-th observation. The *predicted residual* is defined as the difference between the model evaluation at $\boldsymbol{x}^{(i)}$ and its prediction based on $\widehat{\mathcal{M}}_{\mathcal{A}}^{(-i)}$:

$$\Delta^{(i)} \equiv \mathcal{M}(\boldsymbol{x}^{(i)}) - \widehat{\mathcal{M}}_{\mathcal{A}}^{(-i)}(\boldsymbol{x}^{(i)}) \tag{4.29}$$

The expected risk is then estimated by the following *leave-one-out error*:

$$Err_{LOO} \equiv \frac{1}{N} \sum_{i=1}^{N} {\Delta^{(i)}}^2 \tag{4.30}$$

The quantity Err_{LOO} is sometimes referred to as *PRESS* (*Predicted Residual Sum of Squares*) (Allen, 1971) or *jacknife error* (Miller, 1974). In our context of linearly parametrized regression, it is possible to calculate analytically each predicted residual as follows (Saporta, 2006, Chapter 17):

$$\Delta^{(i)} = \frac{\mathcal{M}(\boldsymbol{x}^{(i)}) - \widehat{\mathcal{M}}_{\mathcal{A}}(\boldsymbol{x}^{(i)})}{1 - h_i} \tag{4.31}$$

where h_i is the i-th diagonal term of the matrix $\boldsymbol{\Psi}(\boldsymbol{\Psi}^{\mathsf{T}}\boldsymbol{\Psi})^{-1}\boldsymbol{\Psi}^{\mathsf{T}}$, using the notation:

$$\boldsymbol{\Psi}_{ij} \equiv \left(\psi_{\boldsymbol{\alpha}_j}(\boldsymbol{x}^{(i)})\right)_{\substack{i=1,\ldots,N \\ j=0,\ldots,\mathrm{card}(\mathcal{A})-1}} \tag{4.32}$$

A proof for the equality in Eq.(4.31) is given in Appendix D. The leave-one-out error rewrites:

$$Err_{LOO} = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{\mathcal{M}(\boldsymbol{x}^{(i)}) - \widehat{\mathcal{M}}_{\mathcal{A}}(\boldsymbol{x}^{(i)})}{1 - h_i} \right)^2 \tag{4.33}$$

As for the training error one considers the following relative leave-one-out error:

$$\varepsilon_{LOO} \equiv \frac{Err_{LOO}}{\hat{\mathbb{V}}[\mathcal{Y}]} \tag{4.34}$$

as well as the following counterpart of R^2 denoted by Q^2:

$$Q^2 \equiv 1 - \varepsilon_{LOO} \tag{4.35}$$

3.3 Corrected error estimates

Various penalty-based methods have been proposed in the literature in order to reduce the sensitivity of error estimates to overfitting. Considering for instance the empirical error Err_{emp}, one gets estimates under the form:

$$Err^*_{emp} \;=\; Err_{emp}\, T(P,N) \tag{4.36}$$

where P denotes the number of terms in the PC approximation and $T(P,N)$ is a correcting factor. In particular the so-called *adjusted empirical error* corresponds to:

$$T(P,N) \;\equiv\; \frac{N-1}{N-P-1} \tag{4.37}$$

The quantity Err^*_{emp} is all the larger since the complexity P increases, *i.e.* as extra terms are included in the metamodel. In this section we focus on a correcting factor which has been derived in Chapelle et al. (2002) for regression using a small experimental design. The factor is defined by:

$$T(P,N) \;\equiv\; \frac{N}{N-P}\left(1 \;+\; \frac{\mathrm{tr}\left(\boldsymbol{C}^{-1}_{emp}\right)}{N}\right) \tag{4.38}$$

where:

$$\boldsymbol{C}_{emp} \;\equiv\; \frac{1}{N}\boldsymbol{\Psi}^{\mathsf{T}}\boldsymbol{\Psi} \tag{4.39}$$

In the following one not only considers a corrected empirical error Err^*_{emp} as in Eq.(4.36), but also suggests the use of an heuristic *corrected leave-one-out error* Err^*_{LOO}. The scaled counterparts of Err^*_{emp} and Err^*_{LOO} are denoted by ε^*_{emp} and ε^*_{LOO}, respectively.

4 Adaptive sparse polynomial chaos approximations

4.1 Sparse polynomial chaos expansions

Let $\mathcal{A}$ be any finite subset of $\mathbb{N}^M$, and let us consider the associated truncated PC expansion:

$$\mathcal{M}_{\mathcal{A}}(\boldsymbol{X}) \;\equiv\; \sum_{\boldsymbol{\alpha}\in\mathcal{A}} a_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{4.40}$$

The maximum length of the indices in $\mathcal{A}$ is denoted by p as shown below:

$$p \;\equiv\; \max_{\boldsymbol{\alpha}\in\mathcal{A}} \|\boldsymbol{\alpha}\|_1 \tag{4.41}$$

using the notation in Section 2. p is referred to as the *degree* of the truncated PC representation. This allows one to define the *index of sparsity* of $\mathcal{A}$ by:

$$IS(\mathcal{A}) \;\equiv\; \frac{\mathrm{card}(\mathcal{A})}{\mathrm{card}(\mathcal{A}^{M,p})} \tag{4.42}$$

The index set $\mathcal{A}$ and the PC expansion $\mathcal{M}_{\mathcal{A}}(\boldsymbol{X})$ are said to be *sparse* if the index of sparsity is small compared to 1.

Using the concept of sparse PC expansions and the error estimates presented in Section 3, it is now possible to devise an algorithm that builds iteratively a sparse PC expansion while mastering the approximation error.

4.2 Algorithm for a step-by-step building of a sparse polynomial chaos approximation

An iterative procedure is now presented for building a PC approximation of the system response using a fixed ED. Similarly to usual stagewise regression, our algorithm accepts (resp. discards) terms in the PC representation according to the induced drop (resp. increase) in the empirical error ε_{emp}. Nevertheless the accuracy of the obtained sparse metamodel is assessed by a more robust error estimate denoted by $\hat{\varepsilon}$, which may be selected among the statistics proposed in Section 3. Such a choice will be discussed in Section 5.1.1.

The procedure is first outlined in the case of low-rank index sets $\mathcal{A}^{M,p,j}$ such as in Blatman and Sudret (2008d, 2009d,b):

1. Choose an ED $\mathcal{X}$ and perform the model evaluations $\mathcal{Y}$ once and for all.
2. Select the values of the algorithm parameters, *i.e.* the target error ε_{tgt}, the maximal PC degree p_{max} and interaction order j_{max} and cut-off values θ_1, θ_2.
3. Initialize the PC degree to $p = 0$, the index set to $\mathcal{A}^{(0)} = \{\mathbf{0}\}$ (where $\mathbf{0}$ is the null element of $\mathbb{N}^M$).
4. For any index length $p \in \{1, \ldots, p_{max}\}$:
   - **Forward step**: for any index rank $j \in \{1, \ldots, \min(j_{max}, p)\}$: the set of candidate indices is defined by $\mathcal{C} \equiv \mathcal{A}^{M,p,j} \setminus \mathcal{A}^{M,p,j-1}$. Compute the empirical errors ε_{emp} corresponding to all the sets $\mathcal{A}^{(p-1)} \cup \{c\}$, $c \in \mathcal{C}$. Add eventually to $\mathcal{A}^{(p-1)}$ those terms in $\mathcal{C}$ that have led to a significant decrease in ε_{emp}, say greater than θ_1, and discard the other candidate terms. Let $\mathcal{A}^{(p,+)}$ be the final truncation set at this stage.
   - **Backward step**: remove in turn each term in $\mathcal{A}^{(p,+)}$ of degree strictly less than p. In each case, compute the PC expansion coefficients and the associated empirical error ε_{emp}. Eventually discard from $\mathcal{A}^{(p,+)}$ those terms that lead to an insignificant increase in ε_{emp}, say less than θ_2. Let $\mathcal{A}^{(p)}$ be the final truncation set. Store the associated error estimate $\hat{\varepsilon}$ coefficient in $\hat{\varepsilon}^{(p)}$.
   - If $\hat{\varepsilon}^{(p)} \leq \varepsilon_{tgt}$, stop.

The procedure has also been adapted to hyperbolic index sets $\mathcal{A}_q^{M,p}$ in Blatman and Sudret (2009a). In this setting, the forward steps no longer include a loop through the rank j and the candidate sets are set equal to $\mathcal{A}_q^{M,p} \setminus \mathcal{A}_q^{M,p-1}$.

Note that the regression calculations only involve analytical derivations, so their computational cost is small or negligible with respect to the model evaluations on the ED. Moreover, it is important to notice that the resulting sparse PC approximation does not depend on the (arbitrary) ordering of the PC basis.

Step-by-step run of the algorithm using a polynomial model

The iterative procedure detailed above is illustrated by the following simple polynomial model in the case of low-order index sets:

$$Y \;=\; \mathcal{M}(\xi_1,\xi_2) \;= 1 + H_1(\xi_1)H_1(\xi_2) + H_3(\xi_1) \tag{4.43}$$

where H_j represents the Hermite polynomial of degree j ($j = 1,\ldots,3$) and ξ_1, ξ_2 are independent standard Gaussian random variables. A random design made of $N = 100$ *Latin Hypercube samples* is used. The various steps of the PC construction are illustrated in Figure 4.12. In this example the polynomials $H_1(\xi_2)$ and $H_3(\xi_2)$ are correctly neglected in the forward steps associated with $p = 1$ and $p = 3$ respectively. All the remaining useless polynomials are discarded in the backward step of iteration $p = 3$, *i.e.* once the polynomial model is fully represented.

4.3 Adaptive sparse polynomial chaos approximation using a sequential experimental design

4.3.1 Modified algorithm

The algorithm proposed in the previous subsection allows one to detect automatically the significant terms in the PC expansion. However, when enriching the PC basis (*i.e.* the index set $\mathcal{A}$), the number of retained terms may get close to the size of the experimental design $\mathcal{X}$, hence a poor conditioning of the regression information matrix. To circumvent this problem, additional points are added to $\mathcal{X}$ until its size N satisfies $N \geq k\ \mathrm{card}(\mathcal{A})$. Although $k = 2$ is commonly used when performing ordinary least-square regression, one rather sets $k = 3$ for building up a sparse PC representation. Indeed, such a rule of thumb provides good results from the author's experience. When the information matrix is well-conditioned, $\mathcal{A}$ is reset to $\{\mathbf{0}\}$ and the basis enrichment procedure is restarted, so that the sparse structure of the PC expansion can be built more accurately than in the previous iteration due to the additional data. A simplified flowchart of the procedure is sketched in Figure 4.13.

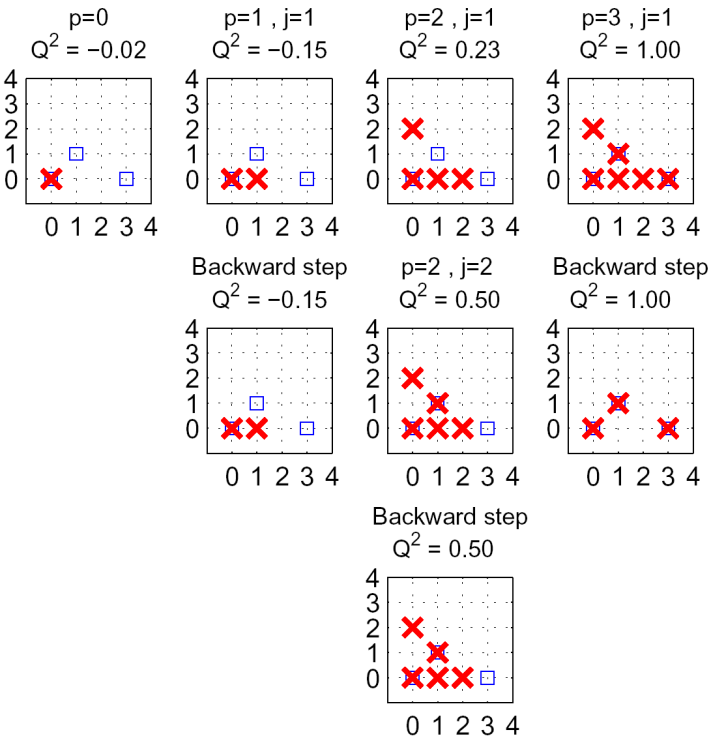


Figure 4.12: *Polynomial model - Step-by-step construction of the sparse polynomial chaos approximation. Squares represent the original polynomial to be recovered. Crosses represent the current polynomial chaos expansion. The x-axis (resp. y-axis) is associated with the degree of the polynomials in ξ_1 (resp. ξ_2). The iterations on the length p and index rank j are respectively displayed from left to right and from top to bottom.*

4.3.2 Sequential experimental designs

The procedure relies upon an adaptation of the experimental design. As model evaluations may be time consuming, it is of major importance to develop a *sequential* strategy, *i.e.* to create a new design by recycling all the already performed computer experiments. The use of quasi-random numbers (Niederreiter, 1992) (*e.g.* Sobol' sequences) is an efficient way to build adaptive space-filling designs. As shown in Blatman and Sudret (2009d), one may alternatively use *nested* Latin Hypercube (NLHS) designs, which are inspired from a sampling scheme described in Wang (2003).

The NLHS technique is explained in the simple case of two independent uniform random variables over the unit square. Assume an initial LHS design of size $N = 3$. The design is to be complemented in such a way that the resulting design is still "LHS-like". As shown in Figure 4.14, the support of each random variable (*i.e.* the interval $[0, 1]$) is first split into 4 equiprobable stratas. Those intervals represented by the current sample set are shaded. If shaded areas are removed from the figure, there is a single cell remaining, in which the additional point

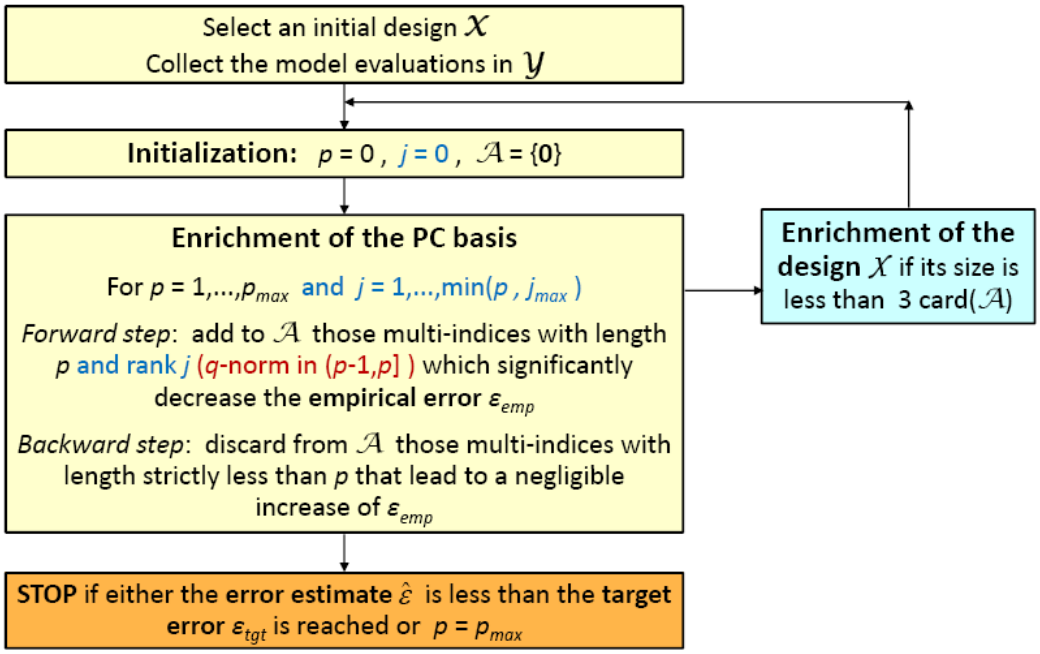


Figure 4.13: *Computational flowchart of the procedure for building up an adaptive sparse polynomial chaos expansion. The statements in blue (resp. red) correspond to low-rank (resp. hyperbolic) index sets.*

is randomly sampled. However it is not always possible to adapt the LHS design this way. Consider for instance the situation depicted in Figure 4.15. After having built the new 4×4 grid, there are two points falling in the same variable interval. If shaded areas are removed, the underrepresented intervals form a 2×2 grid: a 2-point (random) LHS sample is then generated in these cells. Note that the 2 samples in cell (3,3) are kept in the analysis, although the resulting scheme is a quasi-LHS. Indeed, the model evaluation in those points has been carried out already, and it is desirable to exploit this information in the sequel.

4.4 Anisotropic sparse polynomial chaos approximation

As pointed out in Section 2.2, the number of terms in the PC approximations may be further reduced by taking into account the fact that the input variables might have a different impact on the model response. The present section focuses on an iterative procedure based on the anisotropic hyperbolic index sets defined in Eq.(4.20), which has been designed in Blatman and Sudret (2009a). The strategy relies upon a computation of approximate total sensitivity indices of the model response at each iteration, which allows an updating of the set of weights for defining the anisotropic norm (Eq.(4.19)). The computational flowchart of the algorithm is depicted in Figure 4.16.

The weights are initialized to $\boldsymbol{w} = \{1, \ldots, 1\}$ (isotropic norm). At each iteration, the total sensi-

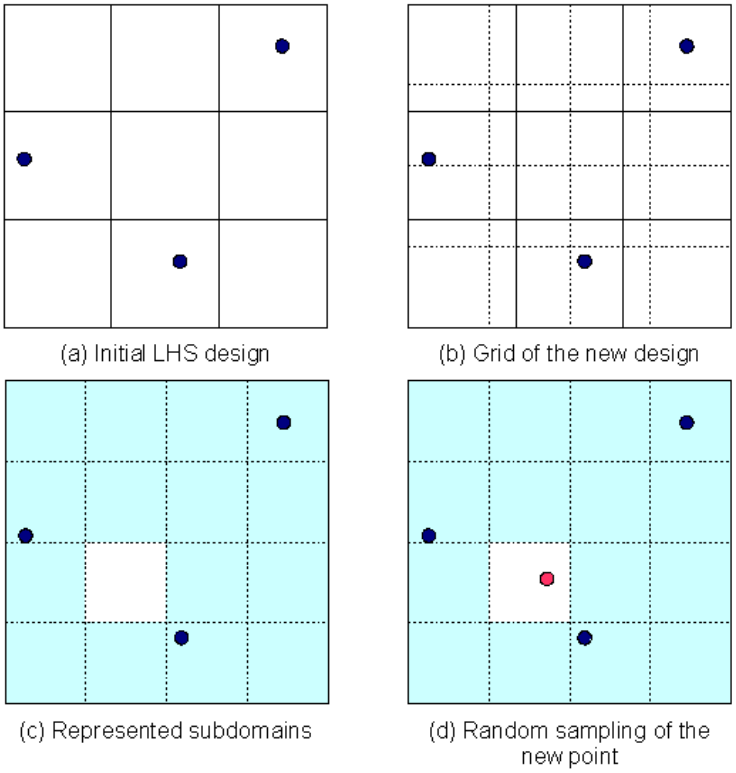


Figure 4.14: *Nested Latin Hypercube - first situation*

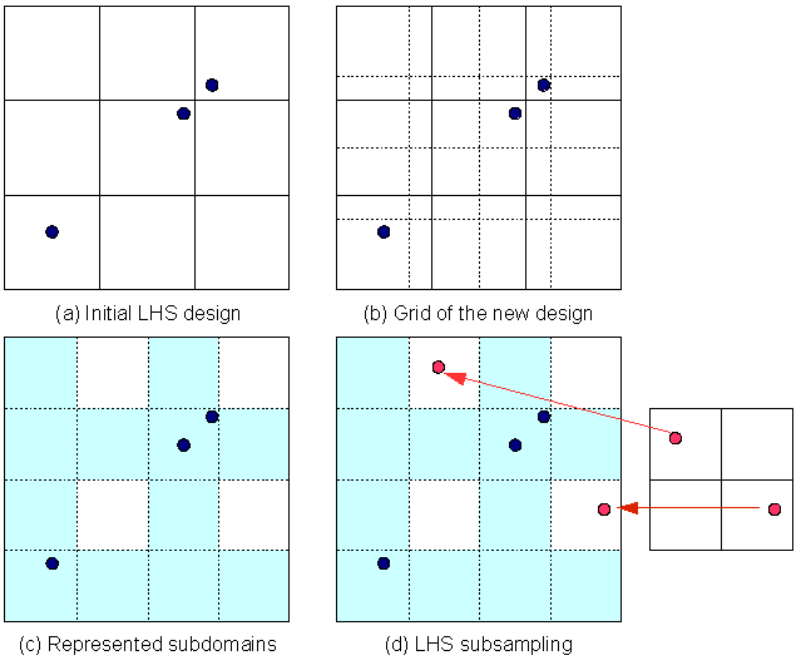


Figure 4.15: *Nested Latin Hypercube - second situation*

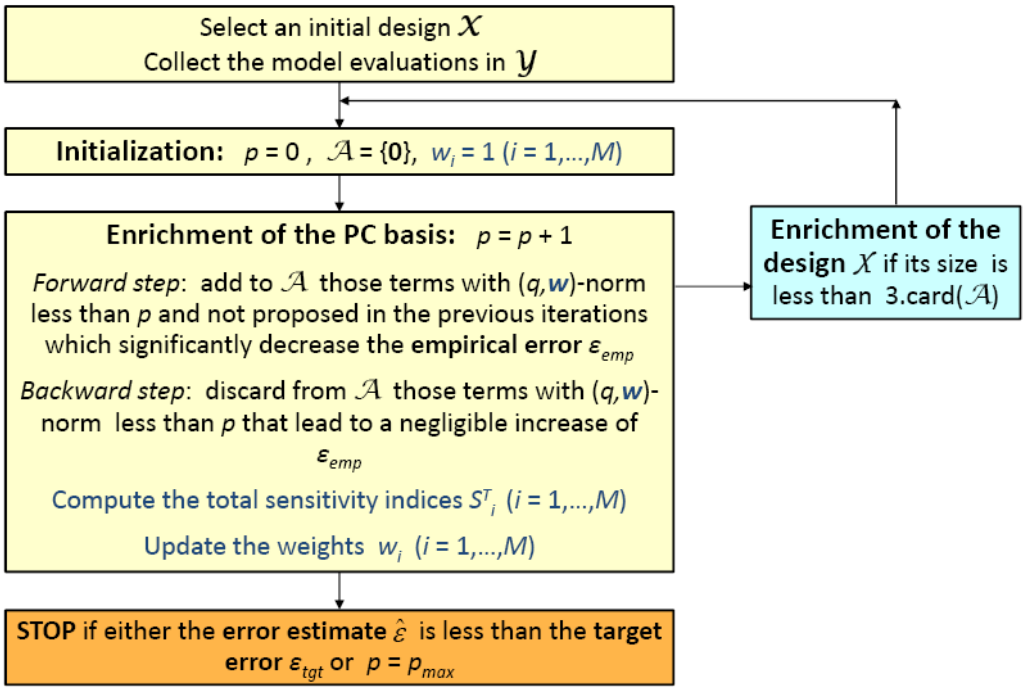


Figure 4.16: *Computational flowchart of the procedure for building up an anisotropic sparse polynomial chaos expansion. The statements in blue are specific to the anisotropic approach.*

tivity indices of the current metamodel are computed and the weights are updated (Eq.(4.20)). Note that the definition of the candidate sets has been used in the forward steps is slightly different than the one used in the isotropic approach (Figure 4.13), due to the updating of the weights. In the anisotropic approach, considering two estimations of the weights $\boldsymbol{w}_0$ and $\boldsymbol{w}$ at iterations $p-1$ and p, the candidate set is defined by $\mathcal{C} \equiv \mathcal{A}_{q,\boldsymbol{w}}^{M,p} \setminus \mathcal{A}_{q,\boldsymbol{w}_0}^{M,p-1}$.

4.5 Case of a vector-valued model response

In this section one considers a *vector-valued* model response $\boldsymbol{Y} \equiv \{Y^{(1)}, \ldots, Y^{(Q)}\}^\mathsf{T} \equiv \mathcal{M}(\boldsymbol{X})$. In stochastic finite element analysis, the components of random vector $\boldsymbol{Y}$ are typically the random components of a field, *e.g.* the nodal displacements or the strains or stresses at the finite element integration points. Vector $\boldsymbol{Y}$ may also be a set of sampled values of a time-dependent response.

In order to build up a sparse PC approximation of $\boldsymbol{Y}$, a naive approach would consist in restarting the stepwise regression procedure for *each* component $\{Y^{(q)}, q = 1, \ldots, Q\}$. Such a strategy may reveal time-consuming though in case of a large number Q of response components. A more efficient method is proposed. It is based on the assumption that the response component $Y^{(q+1)}$ is quite similar to the component $Y^{(q)}$, *i.e.* that the response random field or process features some autocorrelation structure. The method devised herein simply consists in starting the adaptive scheme for approximating $Y^{(q+1)}$ with the resulting sparse basis $\mathcal{A}^{(q)}$ determined for

$Y^{(q)}$. The computational flowchart of the procedure is presented in Figure 4.17.

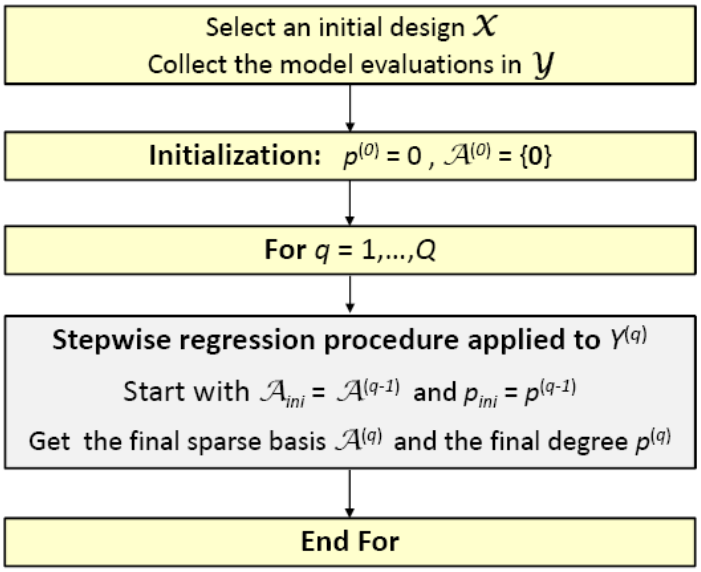


Figure 4.17: *Computational flowchart of the procedure for building up a sparse polynomial chaos expansion of a vector-valued random response*

The algorithm eventually produces a sparse PC approximation of each response component of the form:

$$Y^{(q)} \approx \sum_{\boldsymbol{\alpha} \in \mathcal{A}^{(q)}} a_{\boldsymbol{\alpha}}^{(q)} \, \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \quad , \quad q = 1, \ldots, Q \tag{4.44}$$

where $\mathcal{A}^{(q)}$ is the determined sparse PC basis and the $a_{\boldsymbol{\alpha}}^{(q)}$'s are the PC coefficients of $Y^{(q)}$. Let us denote by $p^{(q)}$ the degree of this PC representation.

Let us now define the PC coefficients matrix $\boldsymbol{A}$ by:

$$\boldsymbol{A} \equiv \left\{ a_i^{(q)} \, , \; q = 1, \ldots, Q \, , \; i = 0, \ldots, P-1 \right\} \tag{4.45}$$

where P is the number of terms that would contain a classical full PC expansion of degree $p \equiv \max_q(p^{(q)})$, *i.e.* $P \equiv \binom{M+p}{p}$. The matrix $\boldsymbol{A}$ is *sparse* and its non zero entries correspond to the coefficients $\{a_{q,\boldsymbol{\alpha}^{(q)}}, \; \boldsymbol{\alpha}^{(q)} \in \mathcal{A}^{(q)}, q = 1, \ldots, Q\}$. The *vector-valued* PC expansion of random vector $\boldsymbol{Y}$ is given by:

$$\boldsymbol{Y} \equiv \mathcal{M}(\boldsymbol{X}) \approx \boldsymbol{A} \, \boldsymbol{\psi}(\boldsymbol{X}) \tag{4.46}$$

where:

$$\boldsymbol{\psi}(\boldsymbol{X}) \equiv \{\psi_0(\boldsymbol{X}), \ldots, \psi_{P-1}(\boldsymbol{X})\}^{\mathsf{T}} \tag{4.47}$$

Just as in the scalar case, the second moments of $\boldsymbol{Y}$ may be obtained *analytically* from the PC coefficients in $\boldsymbol{A}$. Indeed, the *mean vector* $\boldsymbol{\mu_Y}$ of $\boldsymbol{Y}$ is approximated by:

$$\boldsymbol{\mu_Y} \approx \{a_0^{(1)}, \ldots, a_0^{(Q)}\}^{\mathsf{T}} \tag{4.48}$$

and its *covariance matrix* $\boldsymbol{C_Y}$ by:

$$\boldsymbol{C_Y} \approx \widetilde{\boldsymbol{A}}\,\widetilde{\boldsymbol{A}}^{\mathsf{T}} \tag{4.49}$$

where $\widetilde{\boldsymbol{A}}$ denotes the matrix made of all the columns of $\boldsymbol{A}$ except the first one.

4.6 Conclusion

Throughout this section, an iterative algorithm based on stepwise regression has been devised for building up a sparse PC approximation of the model response. The procedure is made of several ingredients, such as a specific approximation basis (say low-rank or hyperbolic), an error estimates and cut-off values for accepting or discarding the terms in the PC representation. In the following, the proposed adaptive scheme is tested on an analytical example while carrying out parametric studies on the tuning parameters.

5 Numerical example

Let us consider again the Sobol' function:

$$Y \equiv \mathcal{M}(\boldsymbol{X}) = \prod_{i=1}^{M} \frac{|4X_i - 2| + c_i}{1 + c_i} \tag{4.50}$$

where $M = 8$ and $\boldsymbol{c} = \{1, 2, 5, 10, 20, 50, 100, 500\}^{\mathsf{T}}$. The model response Y is approximated by various *sparse* PC expansions that are built using the proposed iterative procedure.

5.1 Parametric studies

The present section is focused on the definition of optimal tuning parameters for the proposed iterative procedure, namely:

- a robust and conservative error estimate to assess the PC approximation;
- optimal cut-off parameters for accepting or discarding terms in the PC expansion.

The sensitivity of the algorithm output to the choice of the experimental design is investigated as well.

In this report, each aspect is studied in the case of the analytical Sobol' function for the sake of illustration. Nonetheless various parametric studies have also been carried out on many other models, see Blatman and Sudret (2008a,c,d, 2009d,c,b,a).

5.1.1 Assessment of the error estimates

The various error estimates defined in Section 4.2 (*i.e.* ε_{emp}, ε_{LOO}, ε^*_{emp} and ε^*_{LOO}) are now compared in terms of the prediction of the approximation error, which may lead to more or less accurate metamodels for a given number of model evaluations. Various target errors ε_{tgt} are considered for building up sparse PC representations, namely 0.05, 0.01 and 0.005. The cut-off value θ for adding and rejecting the candidate terms is set equal to $\delta \times \varepsilon_{tgt}$, with $\delta = 0.01$. The relevance of such a choice will be discussed in the next section. The coefficients of the PC approximations are estimated by regression using a Sobol' quasi-random experimental design.

The error estimates (denoted by $\widehat{\varepsilon}$) are assessed as follows:

1. for each type of error estimate $\widehat{\varepsilon}$, run the iterative procedure until $\widehat{\varepsilon} < \varepsilon_{tgt}$;
2. compute a reference error ε_{ref}, and compare this value to the final estimated error $\widehat{\varepsilon}$.

The reference relative error is defined by:

$$\varepsilon_{ref} \equiv \frac{\sum_{i=1}^{\mathcal{N}} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \widehat{\mathcal{M}}_{\mathcal{A}}(\boldsymbol{x}^{(i)}) \right)^2}{\sum_{i=1}^{\mathcal{N}} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \bar{y} \right)^2} \quad , \quad \mathcal{N} = 50,000 \tag{4.51}$$

where the $\boldsymbol{x}^{(i)}$'s are random realizations of the input random vector $\boldsymbol{X}$, and $\bar{y}$ is defined by:

$$\bar{y} \equiv \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \mathcal{M}(\boldsymbol{x}^{(i)}) \tag{4.52}$$

The results are gathered in Table 4.1.

Table 4.1: *Sobol' function - Approximation errors and number of model evaluations when using various error estimates for building up sparse polynomial chaos expansions (the cut-off value θ is set equal to $0.01\varepsilon_{tgt}$, and hyperbolic index sets $\mathcal{A}^{M,p}_{0.4}$ are used)*

Error estimate	$\varepsilon_{tgt} = 0.05$		$\varepsilon_{tgt} = 0.01$		$\varepsilon_{tgt} = 0.005$	
	ε_{ref}	N	ε_{ref}	N	ε_{ref}	N
Empirical error	0.0759	80	0.0229	350	0.0101	450
LOO error	0.0564	110	0.0104	400	0.0060	500
Corrected empirical error	0.0581	110	0.0115	400	0.0083	500
Corrected LOO error	0.0355	170	0.0071	400	0.0041	550

As expected, the empirical error leads to a noticeable underestimation of the approximation error. Note that the corrected empirical error still overpredicts the accuracy of the PC metamodels. The use of the leave-one-out error estimates reduces this downward bias. However only

the corrected leave-one-out estimate provides a conservative assessment of the approximation error. As shown in Figure 4.18, it also leads to a better convergence rate of the PC approximation than the other error estimates. As a result, the corrected leave-one-out error estimate ε^*_{LOO} will be used from now on.

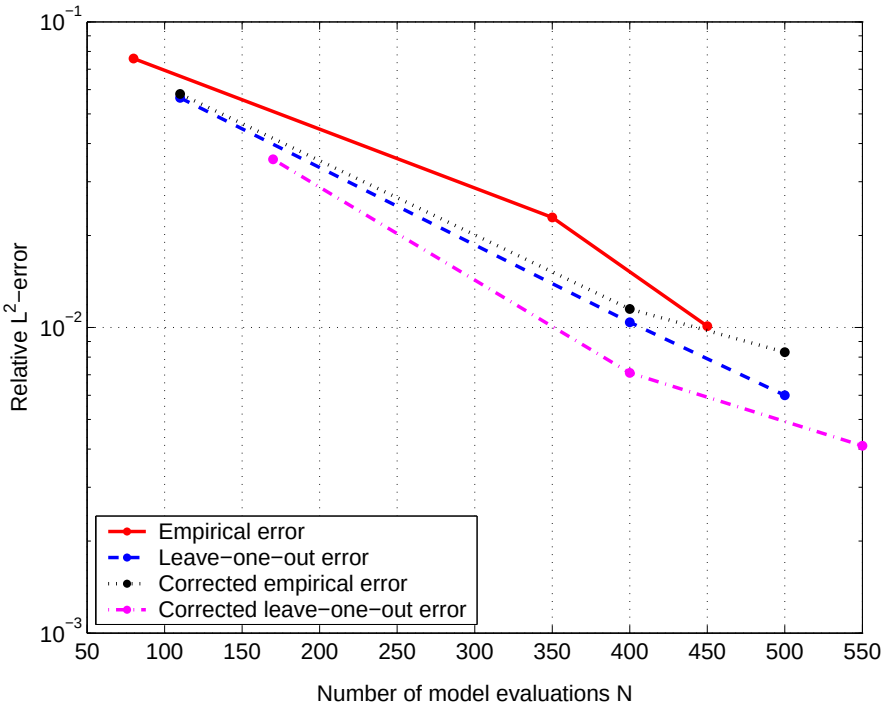


Figure 4.18: *Sobol' function - Convergence rates of the adaptive sparse PC approximations based on hyperbolic index sets* $\mathcal{A}^{M,p}_{0.4}$ *for various error estimates (the cut-off value* θ *is set equal to* $0.01\varepsilon_{tgt}$*)*

5.1.2 Sensitivity to the values of the cut-off parameters

The cut-off parameters $\theta = \theta_1 = \theta_2$ correspond to the threshold used to accept and discard candidate terms. Their value should depend on the prescribed error ε_{tgt}. Indeed it would not make any sense to require a small error ε_{tgt} (say 10^{-5}) while choosing a large selection threshold θ (say 0.1) since only few terms would be retained in the PC expansion, necessarily leading to a poor approximation. In the sequel one uses the thumb rule $\theta = \delta\ \varepsilon_{tgt}$, where δ is a constant whose value may range from 10^{-3} to 10^{-1}.

The sparse PC approximation is sought using *hyperbolic* candidate sets of indices $\mathcal{A}^{M,p}_q$. The tuning parameter q is set equal to 0.4 since this provided the best results in Section 2.2.2. The PC coefficients are computed by regression using Sobol' quasi-random experimental designs. The convergence rates of the various sparse PC approximations are depicted in Figure 4.19.

It appears that a large δ (say $\delta = 0.05$) leads to the fastest convergence rate in order to reach a low accuracy, say $\varepsilon_{tgt} < 10^{-2}$. However the corresponding sparse metamodel does not converge

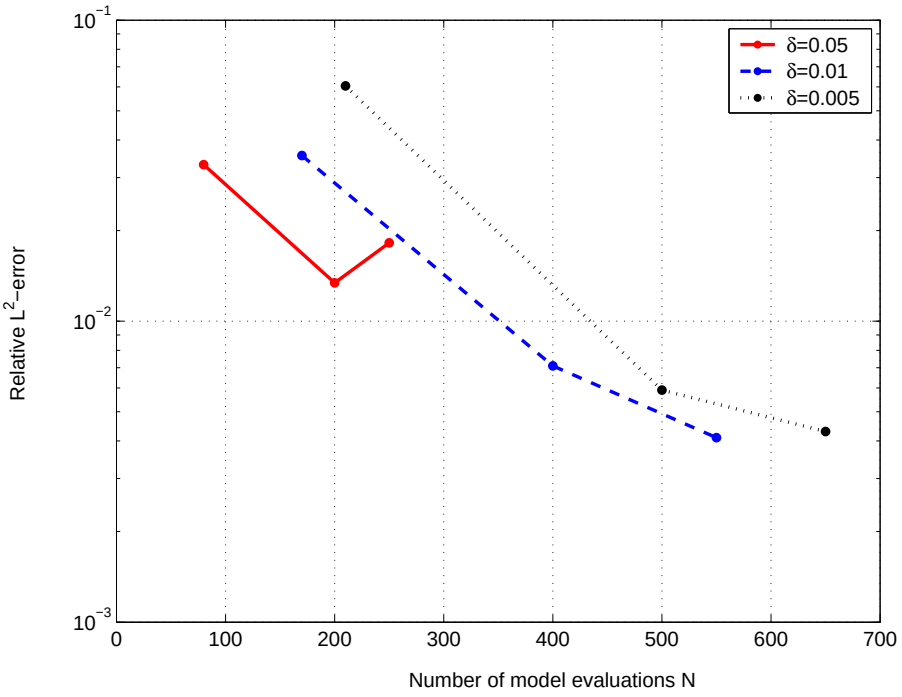


Figure 4.19: *Sobol' function - Convergence rates of the adaptive sparse PC approximations based on hyperbolic index sets* $\mathcal{A}_{0.4}^{M,p}$ *for various values of* δ *(*$\theta = \theta_1 = \theta_2 = \delta\ \varepsilon_{tgt}$*)*

when a higher target accuracy is required. This is due to a too severe overshrinkage of the PC approximation. When a higher accuracy is required, $\delta = 0.01$ leads to a better convergence rate than $\delta = 0.005$. In the lack of further information, the cut-off value θ will be set equal by default to 0.01 ε_{tgt} in the sequel.

5.1.3 Sensitivity to random NLHS designs

It may be noticed that using a NLHS design leads to a random output of the iterative algorithm since it is itself random. Consequently, it is necessary to investigate the sensitivity of the resulting sparse PC expansion (in terms of obtained sparse PC basis as well as coefficient estimates) to the random NLHS experimental design. This is carried out for two values of the target accuracy ε_{tgt}, say 0.05 and 0.01. To this end, the iterative procedure is repeatedly run from 50 independent NLHS designs. For each replicate, a particular sparse PC expansion is obtained. The statistics of the corresponding approximation errors and required number of model evaluations are represented in Figure 4.20 together with the values obtained when using a quasi-random design as in the previous section.

For a low prescribed accuracy (say $\varepsilon_{tgt} = 0.05$), it appears that the median of the relative errors is slightly less than the error obtained by quasi-Monte Carlo (QMC). Nonetheless QMC only requires a number N of model evaluations that is close to the lower quartile of the computational

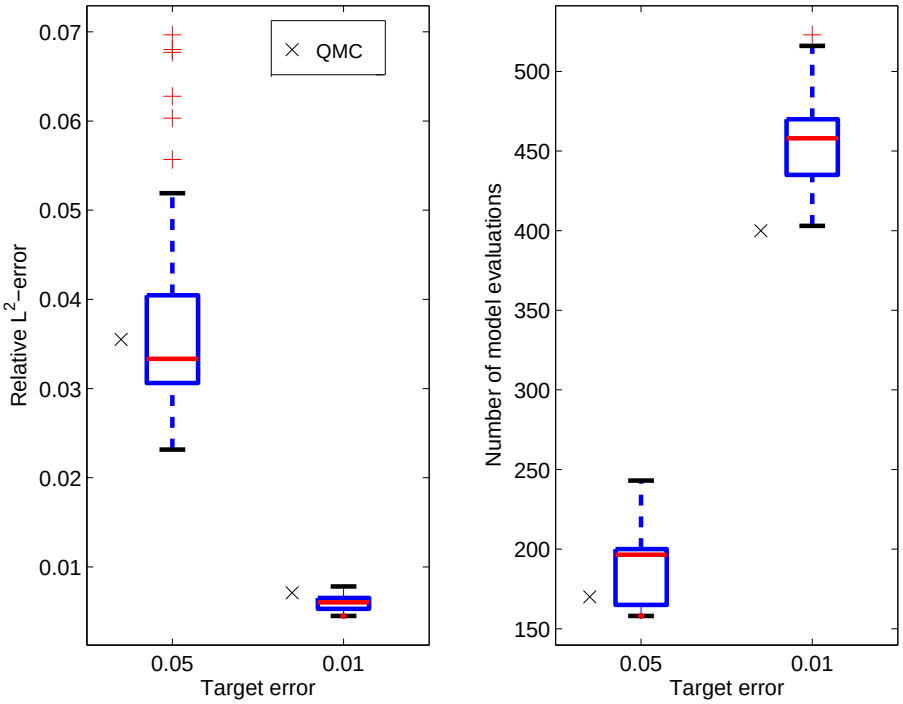


Figure 4.20: *Sobol' function - Box plots of the approximation errors and the required number of model evaluations corresponding to 50 independent NLHS designs. Each box has lines at the lower quartile, median, and upper quartile values. The whiskers are lines extending from each end of the boxes to show the extent of the data lying more than* $1.5 \cdot IQR$ *lower than the lower quartile or* $1.5 \cdot IQR$ *higher than the upper quartile (IQR stands for interquartile range). Outliers (red crosses) are data with values beyond the ends of the whiskers.*

costs associated with the NLHS designs. Moreover NLHS suffers a large dispersion of the results. In particular, it appears that a non negligible proportion of NLHS designs provide metamodels with a relatively poor accuracy (outliers in Figure 4.20). The dispersion in the approximation error considerably decreases when setting ε_{tgt} equal to 0.01, hence an almost deterministic error. In this respect, the proposed iterative procedure seems to converge when N increases. Note that a moderate dispersion in N is observed though, with a N typically ranging from 400 to 500 model evaluations.

5.2 Full versus sparse polynomial chaos approximations

5.2.1 Convergence results

This section is dedicated to the comparison of the various categories of PC approximation that have been investigated so far, namely:

- usual full PC approximations, for a degree p varying from 3 to 5 (as for the examples in Section 2, the PC coefficients are computed by regression using $N = 2P$ model evaluations,

where P denotes the PC size);

- full low-rank (the maximal rank j is set equal to 2) and hyperbolic (the value of the q-norm is set equal to 0.4) PC approximations (again $N = 2P$ model evaluations are used);
- a sparse low-rank PC metamodel, for a target accuracy ε_{tgt} which is successively set equal to 0.05, 0.01, 0.005 and 0.001;
- sparse hyperbolic PC representations (both the isotropic and anisotropic approaches are applied), for the same values of ε_{tgt}.

The coefficients of the PC expansions are evaluated from a quasi-random experimental design. The convergence rates of the various approaches are depicted in Figure 4.21.

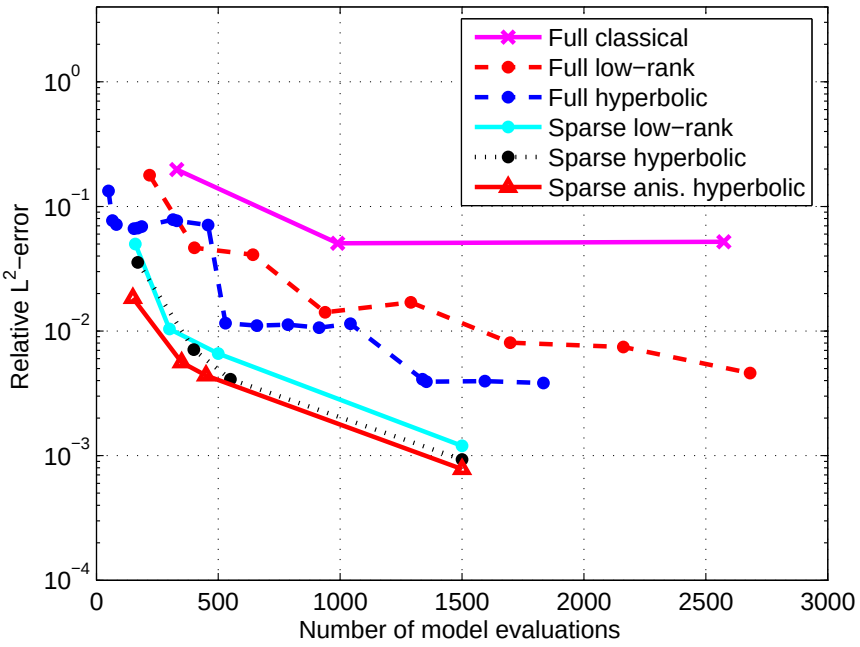


Figure 4.21: *Sobol' function - Convergence rates of the various PC approximations. The value $j = 2$ (resp. $q = 0.4$) has been selected for the low-rank (resp. hyperbolic) index sets.*

As already noticed in the previous examples, the usual full PC approximation appears to be the least efficient approach. The full low-rank metamodel is overperformed by the hyperbolic one. For instance, the latter requires about twice as many less model evaluations (*i.e.* $N = 500$) than the former ($N = 938$) in order to reach a 1% accuracy. On the other hand, the sparse PC expansions converge more rapidly than the full ones. The sparse hyperbolic metamodels outperform the low-rank ones. In particular, the anisotropic approach is the most efficient scheme, reaching a 0.5% accuracy using only $N = 400$ calculations, *i.e.* a noticeable computational gain factor of 4.5 compared to the full hyperbolic PC representation, as shown from the results in

Table 4.2. This gain is considerably large with respect to the usual full expansion. Indeed, the sparse metamodels provide a 10 times smaller relative error while using 10 times less model evaluations. Note that the "sparse" approaches yield accurate approximations that only contain a low number of terms, *i.e.* with an index of sparsity IS ranging from 3% to 16%.

Table 4.2: *Sobol' function - Comparison of the accuracy and the efficiency of various PC approximations. The value $j = 2$ (resp. $q = 0.4$) has been selected for the low-rank (resp. hyperbolic) index sets. The target error ε_{tgt} has been set equal to* 0.005 *for the sparse metamodels.*

	Full PCE's			Sparse PCE's - $\varepsilon_{tgt} = 0.005$		
	Classical	Low-rank	Hyperb.	Low-rank	Hyperb.	Anis. hyperb.
Relative $\mathcal{L}^2$-error	0.0507	0.0141	0.0041	0.0066	0.0042	0.0050
# model evaluations	2,574	2,682	1,834	500	550	400
PC degree	5	10	20	13	20	24
# basis terms	1,287	1,341	917	68	145	134
Index of sparsity (%)		-		3	16	5

5.2.2 Global sensitivity analysis

The sensitivity indices of the response of the Sobol' function can be derived analytically (Sobol', 2003). Estimates of the first-order and the total sensitivity indices are computed by post-processing several PC approximations among those that have been considered in the previous section, namely:

- the full "usual" fifth-order PC expansion (index set $\mathcal{A}^{M,5}$);
- the full hyperbolic PC metamodel (index set $\mathcal{A}^{M,5}_{0.4}$);
- the sparse hyperbolic PC metamodel;
- the anisotropic sparse hyperbolic PC metamodel.

The results are gathered in Table 4.3 together with the reference values.

It appears that the classical fifth-order full metamodel provides less accurate estimates than all other PC approximations. Considering only those indices that are greater than 0.1, a maximal relative error of 0.4% is observed for the full hyperbolic expansion. This error is equal to 1.2% for the sparse metamodels. However the latter allow a dramatic reduction of the computational cost compared to the full approximations, as shown in the previous section. The anisotropic approach appears to be the most efficient scheme. It is shown in Figure 4.22 that it yields a

Table 4.3: *Sobol' function - Estimates of the first-order and the total sensitivity indices by post-processing various PC approximations. The value $q = 0.4$ has been selected for hyperbolic index sets.*

Sensitivity Index	Analytical	Full PCE's		Sparse PCE's	
		Classical	Hyperb.	Hyperb.	Anis. Hyperb.
S_1	0.604	0.585	0.606	0.606	0.606
S_2	0.268	0.264	0.268	0.268	0.271
S_3	0.067	0.067	0.067	0.068	0.067
S_4	0.020	0.017	0.020	0.020	0.019
S_5	0.006	0.005	0.005	0.005	0.005
S_6	0.001	0.001	0.001	0.001	0.001
S_7	0.000	0.000	0.000	0.000	0.000
S_8	0.000	0.000	0.000	0.000	0.000
S_1^T	0.634	0.624	0.634	0.634	0.635
S_2^T	0.295	0.301	0.292	0.293	0.295
S_3^T	0.076	0.087	0.075	0.076	0.074
S_4^T	0.023	0.030	0.022	0.022	0.021
S_5^T	0.006	0.016	0.007	0.006	0.006
S_6^T	0.001	0.015	0.002	0.001	0.001
S_7^T	0.000	0.014	0.001	0.001	0.001
S_8^T	0.000	0.013	0.001	0.000	0.000
# model evaluations		2,574	1,834	550	400

sparse PC basis which contains a large number of polynomials in the significant variables, unlike the isotropic approach which provides a quite isotropic solution.

6 Conclusion

Computing the PC expansion of a model with random input parameters may lead to intractable calculations in high dimensions when the usual truncation scheme is applied. To overcome this difficulty, two alternative strategies for truncating the PC representation have been investigated, namely the *low-rank* and the *hyperbolic* index sets. Both approaches provide metamodels whose bases contain a small number of significant terms. Thus the corresponding PC coefficients may be computed by regression using a low number of model evaluations, *i.e.* at a reduced computational cost.

This cost may be further reduced by approximating the model response by a *sparse* PC representation, *i.e.* which only contains a small number of non zero coefficients. The present work

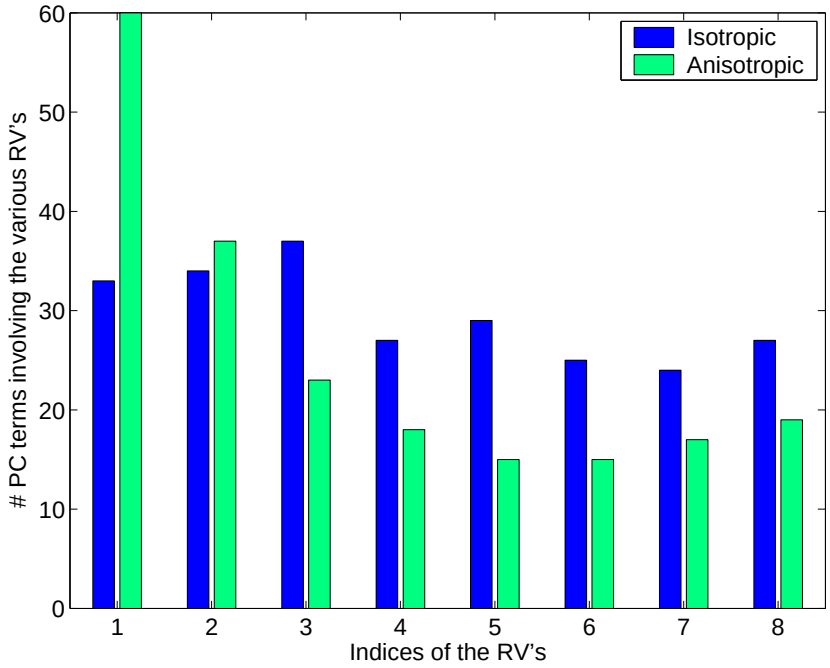


Figure 4.22: *Sobol' function - Anisotropy of the obtained sparse polynomial chaos basis when using isotropic and anisotropic hyperbolic index sets*

is focused on an incremental research of the significant terms. To this end, it was necessary to propose suitable error estimates. Then an algorithm has been proposed to build iteratively a sparse PC expansion. An anisotropic version has been devised in order to take benefit of the fact that the model response might be principally governed by a small group of significant random variables.

The various approaches have been applied to the approximation of the analytical *Sobol' function* for the sake of illustration. The hyperbolic truncation scheme appears to be the most efficient strategy, overperforming both the usual and low-rank indices approaches. The sparse PC approximations have allowed a dramatical reduction of the computational cost to reach a given accuracy, with a computational gain factor of 4.5 (resp. more than 10) with respect to the full hyperbolic (resp. usual) PC decomposition. The anisotropic approach yielded the best trade-off between accuracy and efficiency. This approach seems promising but requires further investigation in order to find a robust choice of the tuning parameters. Accordingly, the post-processing of the sparse metamodels have led to excellent estimates of the sensitivity indices of the Sobol' function using about 400 model evaluations.

Several parametric studies have allowed to select suitable values of the algorithm parameters, such as the cut-off values for accepting and discarding the terms in the PC approximation. However such a choice may depend on the model under consideration, hence the necessity of validation on other examples. As already mentioned, various analytical functions have been tested in order to validate the algorithms presented in this report, see Blatman and Sudret

(2008a,c, 2009d, 2008d, 2009b,a,c). In this respect, some modern techniques for model selection in statistics, such as the *Least Angle Regression* (LAR) scheme (Efron et al., 2004), seem particularly attractive insofar as they do not feature any tuning parameter. In addition, LARS allows one to perform regression calculations when the design size is smaller then the number of unknown coefficients, which completely matches our concern of minimizing the computational cost. Hence the next chapter is devoted to the analysis of this method.

Chapter 5

Adaptive sparse polynomial chaos approximations based on Least Angle Regression

1 Introduction

Computing the polynomial chaos (PC) expansion of the response of a physical model featuring a large number of input parameters may not be an easy task in practice. Indeed, the number of terms in the representation blows up with the dimensionality of the problem, and so does the required number of model evaluations (*i.e.* the computational cost) when using an ordinary least-square regression scheme. This problem has been tackled in the previous chapter by designing an iterative procedure to build up a *sparse* metamodel while mastering the approximation error. The algorithm produces a PC approximation with a low number of non-zero coefficients, which may be computed using a rather small number of model evaluations compared to a usual "full" expansion. Such an approach may be regarded as a *variable selection* method insofar as it selects a small set of variables among a large set of candidates that best explain the model response.

Variable selection in regression has received much interest in the field of statistics. The so-called *stepwise regression* (Efroymson, 1960) scheme is a classical method for selecting a subset of predictors. Actually the iterative methodology which has been proposed in the previous chapter appears to be a variant that involves cut-off parameters for including and discarding terms. In the statistics literature, stepwise regression is known to be *overgreedy* (*i.e.* to accept too easily new predictors) and quite unstable (*i.e.* sensitive to a change in the set of observations). The algorithms presented in the previous chapters seem to have reduced some of these limitations. However recent techniques such as *Least Angle Regression* (LAR) (Efron et al., 2004) are efficient alternatives that yield accurate and robust results at a low computational cost. In particular,

LAR seems particularly attractive since it is well suited to the case in which the number of predictors is of similar size to the number of observations N, or even possibly significantly larger than N. This fully matches our objective of minimizing the number of calls to the possible computationally demanding model.

The present chapter is aimed at developing an adaptive algorithm based on LAR in order to build up a sparse PC approximation by means of a small number of model evaluations. As in the previous chapter, such an adaptivity has two meanings, namely:

- *basis adaptivity*: the degree of the metamodel is iteratively increased;
- *experimental design adaptivity*: the experimental design is automatically enriched as soon as overfitting is detected.

The present chapter is organized as follows. First of all, various methods for variable selection are reviewed in Section 2. The so-called *Least Angle Regression* (LAR) scheme is given a special interest. It is shown that LAR provides a set of solutions rather than a unique solution as for ordinary least-square regression. Criteria are investigated in Section 3 to select the optimal LAR solution. Using the most appropriate criterion, an adaptive algorithm based on LAR is devised in Section 4. An extension of the procedure to adaptive experimental designs is also proposed. Lastly, the LAR-based iterative algorithm is applied in Section 6 to the sensitivity analysis of the so-called Sobol' function which involves 8 random variables.

2 Methods for regression with many predictors

2.1 Mathematical framework

Consider the mathematical model $\mathcal{M}$ of a physical system depending on M input parameters $\boldsymbol{x} \equiv \{x_1, \ldots, x_M\}^\mathsf{T}$. As the latter are assumed to be affected by uncertainty, they are represented by random variables gathered in random vector $\boldsymbol{X} \equiv \{X_1, \ldots, X_M\}^\mathsf{T}$. As a consequence the response of the model is also a random vector $\boldsymbol{Y} \equiv \mathcal{M}(\boldsymbol{X})$, which is assumed to be a scalar random variable Y for the sake of simplicity. The components of $\boldsymbol{X}$ are supposed to be independent.

As shown in Chapter 3, Section 2, the model response may be expanded onto the *polynomial chaos* (PC) *basis* as follows:

$$\mathcal{M}(\boldsymbol{X}) \;\approx\; \mathcal{M}_{\mathcal{A}}(\boldsymbol{X}) \;\equiv\; \sum_{\boldsymbol{\alpha}\in\mathcal{A}} a_{\boldsymbol{\alpha}}\, \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{5.1}$$

where $\mathcal{A}$ is a given set of multi-indices, *i.e.* a non empty finite subset of $\mathbb{N}^M$. $\mathcal{A}$ may be typically selected in such a way that only those polynomials of total degree not greater than p are retained.

Otherwise more sophisticated sets may be chosen, such that the *low-rank* or the *hyperbolic* sets that have been defined in Chapter 4. The basis polynomials $\psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$ will be sometimes referred to as the *predictors*.

Let us consider a set of realizations of the input random vector $\boldsymbol{X}$ (*i.e.* an experimental design) $\mathcal{X} \equiv \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}^\mathsf{T}$ as well as the vector $\mathcal{Y} \equiv \{\mathcal{M}(\boldsymbol{x}^{(1)}), \ldots, \mathcal{M}(\boldsymbol{x}^{(N)})\}^\mathsf{T} \equiv \{y^{(1)}, \ldots, y^{(N)}\}^\mathsf{T}$ of the corresponding model evaluations. It is assumed that the number of terms $P \equiv \mathrm{card}(\mathcal{A})$ in the PC basis is of similar size to N, or even possibly significantly larger than N. In such a situation it is not possible to compute the PC coefficients by ordinary least-square regression, since the corresponding system is ill-posed. Methods that may be employed as an alternative are reviewed in the sequel.

2.2 Stepwise regression and all-subsets regression

Stepwise regression is a method for selecting those predictors $\psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$ in Eq.(5.1) that have the greatest impact on the model response $\mathcal{M}(\boldsymbol{X})$, hence the label *variable selection* method. The so-called *forward stepwise* method begins by selecting the predictor which yields the best fit, *i.e.* that leads to largest drop in the residual sum of squares. Then one adds the predictor which provides the best fit in addition to the first predictor, and so on. The procedure stops when a stopping criterion is reached. In contrast, *backward elimination* starts with all the predictors, and then sequentially discards those terms which lead to the least contributions to the fit. A procedure developed in Efroymson (1960) combines the forward and the backward methods. In this setup, at each step of forward selection, the possibility of deleting a variable as in backward elimination is considered. Stepwise regression is known in statistics to be a greedy and quite unstable procedure (Hesterberg et al., 2008).

All-subsets regression (Furnival and Wilson, 1974) is a variant of stepwise regression which studies the impact of all the subsets of predictors of each size, *i.e.* all the subsets of the multi-index set $\mathcal{A}$. Although this approach is exhaustive, it may be computationally demanding since it considers a huge number of possible metamodels, especially in high dimensions (say $M \geq 8$).

2.3 Ridge regression

Ridge regression (Hoerl and Kennard, 1970) relies upon a penalization of the value of the coefficients, whose absolute value will be typically smaller than if they would have been computed by ordinary least-square regression. This corresponds to a regularization of the regression problem which ensures the existence and the uniqueness of a solution, even in the case $N < P$. The ridge

regression problem reads:

$$\text{Minimize} \qquad \sum_{i=1}^{N} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) \; - \; \boldsymbol{a}^{\mathsf{T}} \boldsymbol{\psi}(\boldsymbol{x}^{(i)}) \right)^2 \; + \; C \, \|\boldsymbol{a}\|_2^2 \tag{5.2}$$

where $\|\boldsymbol{a}\|_2^2 \equiv \sum_{j=0}^{\text{card}(\mathcal{A})-1} \alpha_j^2$ and C is a non negative constant.

The minimality condition with respect to $\boldsymbol{a}$ reads:

$$2 \, \boldsymbol{\Psi}^{\mathsf{T}} \, \boldsymbol{\Psi} \, \hat{\boldsymbol{a}} \; - \; 2 \, \boldsymbol{\Psi}^{\mathsf{T}} \, \mathcal{Y} \; + 2 \, C \, \hat{\boldsymbol{a}} \; = \; 0 \tag{5.3}$$

that is:

$$\left(\boldsymbol{\Psi}^{\mathsf{T}} \, \boldsymbol{\Psi} \; + \; C \, \boldsymbol{I} \right) \; \hat{\boldsymbol{a}} \; = \; \boldsymbol{\Psi}^{\mathsf{T}} \, \mathcal{Y} \tag{5.4}$$

where $\boldsymbol{I}$ denotes the unit matrix. $C = 0$ corresponds to ordinary least-square regression, and the coefficients tend to zero as C increases. In practice, one may retain the value of C that yields the minimum error of approximation. Such a choice could be carried out by cross-validation.

Ridge regression is not a proper variable selection method since it includes all the predictors in the metamodel, unless the true value of the corresponding coefficient is zero. Methods providing *sparse* metamodels are preferred in the present work since they provide more interpretable metamodels. A $\mathcal{L}^1$-penalized regression scheme may be used in this purpose.

2.4 LASSO

The so-called *LASSO* (for *Least absolute shrinkage and selection operator*) method (Tibshirani, 1996) is based on a $\mathcal{L}^1$-penalized regression as follows:

$$\text{Minimize} \qquad \sum_{i=1}^{N} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) \; - \; \boldsymbol{a}^{\mathsf{T}} \boldsymbol{\psi}(\boldsymbol{x}^{(i)}) \right)^2 \; + \; C \, \|\boldsymbol{a}\|_1^2 \tag{5.5}$$

where $\|\boldsymbol{a}\|_1 \equiv \sum_{j=0}^{\text{card}(\mathcal{A})-1} |a_j|$ and C is a non negative constant. A nice feature of LASSO is that it provides a *sparse* metamodel, *i.e.* it discards insignificant variables from the set of predictors. The obtained metamodel is all the sparser since the value of the tuning parameter C is high.

For a given $C \geq 0$, the solving procedure may be implemented via quadratic programming. Obtaining the whole set of coefficient estimates for C varying from 0 to a maximum value may be computationally expensive though since it requires solving the optimization problem for a fine grid of values of C.

2.5 Forward stagewise regression

Another procedure, known as *forward stagewise regression* (Hastie et al., 2001, Section 10.12.2), appears to be different from LASSO, but turns out to provide similar results. Similarly to stepwise regression, the procedure first picks the predictor that is most correlated with the vector of observations. However, the value of the corresponding coefficient is only increased by a small amount. Then the predictor with largest correlation with the current residual (possible the same term as in the previous step) is picked, and so on. Let us introduce the vector notation:

$$\boldsymbol{\psi}_{\alpha_j} \equiv \{\psi_{\alpha_j}(\boldsymbol{x}^{(1)}), \ldots, \psi_{\alpha_j}(\boldsymbol{x}^{(N)})\}^{\mathsf{T}} \tag{5.6}$$

The forward stagewise algorithm is outlined below:

1. Start with $\boldsymbol{R} = \mathcal{Y}$ and $a_0 = \cdots = a_{P-1} = 0$.
2. Find the predictor $\boldsymbol{\psi}_{\alpha_j}$ that is most correlated with $\boldsymbol{R}$.
3. Update $\hat{a}_j = \hat{a}_j + \delta_j$, where $\delta_j \equiv \epsilon \cdot \text{sign}(\boldsymbol{\psi}_{\alpha_j}^{\mathsf{T}} \boldsymbol{R})$.
4. Update $\boldsymbol{R} = \boldsymbol{R} - \delta_j \boldsymbol{\psi}_{\alpha_j}$ and repeats Steps 2 and 3 until no predictor has any correlation with $\boldsymbol{R}$.

Note that parameter ϵ has to be set equal to a small value in practice, *e.g.* $\epsilon = 0.01$. This procedure is known to be more stable than traditional stepwise regression.

2.6 Least Angle Regression

2.6.1 Description of the Least Angle Regression algorithm

Least Angle Regression (LAR) (Efron et al., 2004) may be viewed as a version of forward stagewise that uses mathematical derivations to speed up the computations. Indeed, instead of taking many small steps with the basis term most correlated with current residual $\boldsymbol{R}$, the related coefficient is directly increased up to the point where some other basis predictor has as much correlation with $\boldsymbol{R}$. Then the new predictor is entered, and the procedure is continued. The LAR algorithm is detailed below:

1. Initialize the coefficients to $a_{\boldsymbol{\alpha}_0}, \ldots, a_{\boldsymbol{\alpha}_{P-1}} = 0$. Set the initial residual equal to the vector of observations $\mathcal{Y}$.
2. Find the vector $\boldsymbol{\psi}_{\boldsymbol{\alpha}_j}$ which is most correlated with the current residual.

3. Move $a_{\boldsymbol{\alpha}_j}$ from 0 toward the least-square coefficient of the current residual on $\boldsymbol{\psi}_{\boldsymbol{\alpha}_j}$, until some other predictor $\boldsymbol{\psi}_{\boldsymbol{\alpha}_k}$ has as much correlation with the current residual as does $\boldsymbol{\psi}_{\boldsymbol{\alpha}_j}$.

4. Move jointly $\{a_{\boldsymbol{\alpha}_j}, a_{\boldsymbol{\alpha}_k}\}^\mathsf{T}$ in the direction defined by their joint least-square coefficient of the current residual on $\{\boldsymbol{\psi}_{\boldsymbol{\alpha}_j}, \boldsymbol{\psi}_{\boldsymbol{\alpha}_k}\}$, until some other predictor $\boldsymbol{\psi}_{\boldsymbol{\alpha}_l}$ has as much correlation with the current residual.

5. Continue this way until $m \equiv \min(P, N-1)$ predictors have been entered.

Steps 2 and 3 mention a "move" of the *active* coefficients toward their least-square value. It corresponds to an updating of the form $\hat{\boldsymbol{a}}^{(k+1)} = \hat{\boldsymbol{a}}^{(k)} + \gamma^{(k)}\tilde{\boldsymbol{w}}^{(k)}$. Vector $\tilde{\boldsymbol{w}}^{(k)}$ and coefficient $\gamma^{(k)}$ are referred to as the LAR *descent direction* and *step*, respectively. Both quantities may be derived algebraically, as shown in Appendix E.

Note that if $N \geq P$, then the last step of LAR provides the ordinary least-square solution. It is shown in Efron et al. (2004) that LAR is noticeably efficient since it only requires $\mathcal{O}(NP^2 + P^3)$ computations (*i.e.* the computational cost of ordinary least-square regression on P predictors) for producing a set of m metamodels. The optimal number of predictors in the metamodel (*i.e.* the optimal number of LAR steps) may be determined using a suitable criterion. This point will be discussed in Section 3.

2.6.2 LASSO as a variant of LAR

It has been shown in Efron et al. (2004); Hastie et al. (2007) that with only one modification, the LAR procedure provides in one shot the entire paths of LASSO solution coefficients as the tuning parameter C in Eq.(5.5) is increased from 0 up to a maximum value. The modified algorithm is as follows:

- Run the LAR procedure from Steps 1 to 4.
- If a non zero coefficient hits zero, discard it from the current metamodel and recompute the current joint least-square direction.
- Continue this way until $m \equiv \min(P, N-1)$ predictors have been entered.

Note that the LAR-based LASSO procedure may take more than m steps since the predictors are allowed to be discarded and introduced later again into the metamodel.

In a similar fashion, a limiting version of the forward stagewise method (Section 2.5) when $\epsilon \to 0$ may be obtained by slightly modifying the original LAR algorithm (Hastie et al., 2007). In the literature, one commonly uses the label LARS when referring to all these LAR-based algorithms (with S referring to *Stagewise* and *LASSO*).

2.6.3 Hybrid LARS

The so-called *hybrid* LARS procedure is a variant of the original LARS (Efron et al., 2004) (LARS refers to either native LAR or LASSO here). Let us assume that after k steps the LARS algorithm has included k predictors in a multi-index set $\mathcal{A}^{(k)}$. Instead of the associated LARS-based coefficients $\hat{\boldsymbol{a}}^{(k)}$ one may prefer the *least-square estimates* coefficients (denoted here by $\hat{\boldsymbol{a}}_{LS}^{(k)}$). In this setup, LARS is only used in order to select a set of predictors, but not to estimate the coefficients. It is shown in Efron et al. (2004) that hybrid LARS always increase the usual empirical measure of fit R^2 compared to the orginal LARS.

An extension has been proposed for the LARS-based LASSO algorithm under the name *relaxed LASSO* (Meinshausen et al., 2007). In this setup, for each submodel $\mathcal{A}^{(k)}$ in the set of the LARS solutions, one performs LASSO again, but with a smaller penalty parameter such that no more variable selection is performed. Hybrid LASSO corresponds to the particular case when no penalty is applied.

2.6.4 Numerical example

Let us consider the so-called Ishigami function which is widely used for benchmarking in global sensitivity analysis (Ishigami and Homma, 1990; Saltelli et al., 2000):

$$Y = \sin X_1 + 7 \sin^2 X_2 + 0.1 X_3^4 \sin X_1 \tag{5.7}$$

where the X_i's ($i = 1, ..., 3$) are independent random variables that are uniformly distributed over $[-\pi, \pi]$. Several projections of the Ishigami function are depicted in Figure 5.1. In addition, the probability density function of Y is obtained using a kernel method (Chapter 2, Section 2.2) and is plotted in Figure 5.2.

The model response is approximated by a tenth-order PC expansion made of normalized Legendre polynomials. Denoting by M (resp. p) the number of input random variables (resp. the PC degree), the corresponding multi-index sets is $\mathcal{A}^{M,p} = \mathcal{A}^{3,10}$ and its cardinality is given by $P \equiv \text{card}(\mathcal{A}^{3,10}) = \binom{3+10}{10} = 286$. The PC coefficients are computed by LAR and LASSO using a quasi-random experimental design of size $N = 75$ (which is obviously smaller than P). The resulting coefficients paths are depicted in Figure 5.3.

It is observed that LAR and LASSO provide identical coefficients profiles in this example. Each vertical line intersects the paths at coefficients values that constitute a particular solution, *i.e.* a particular metamodel.

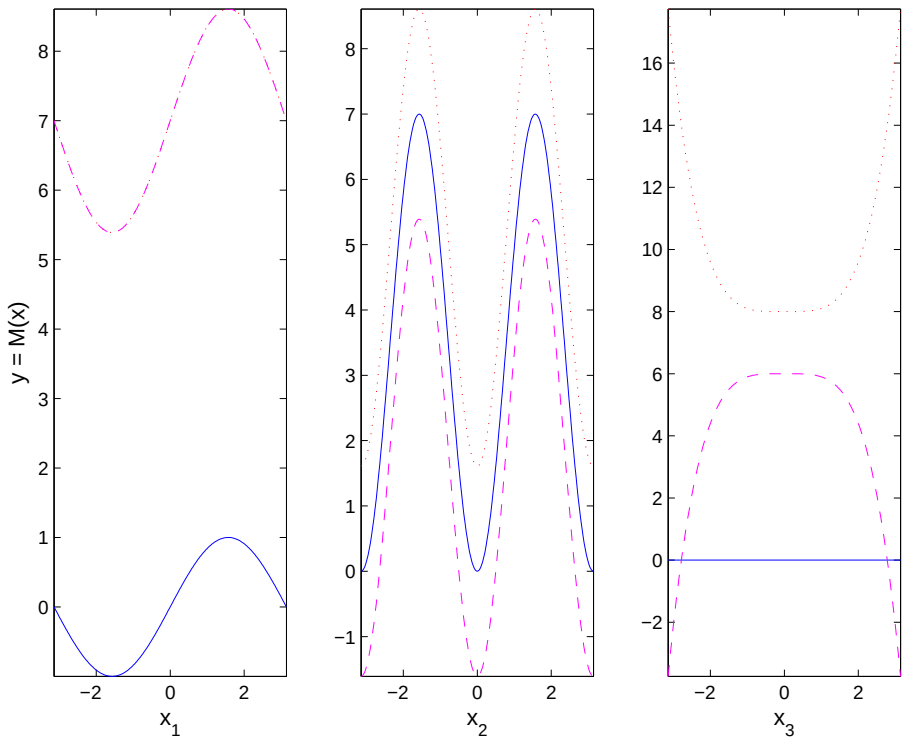


Figure 5.1: *Ishigami function - Several projections of the function. Solid (resp. dotted and dashed) lines correspond to fixed variables that are set equal to 0 (resp. $\pi/2$ and $-\pi/2$).*

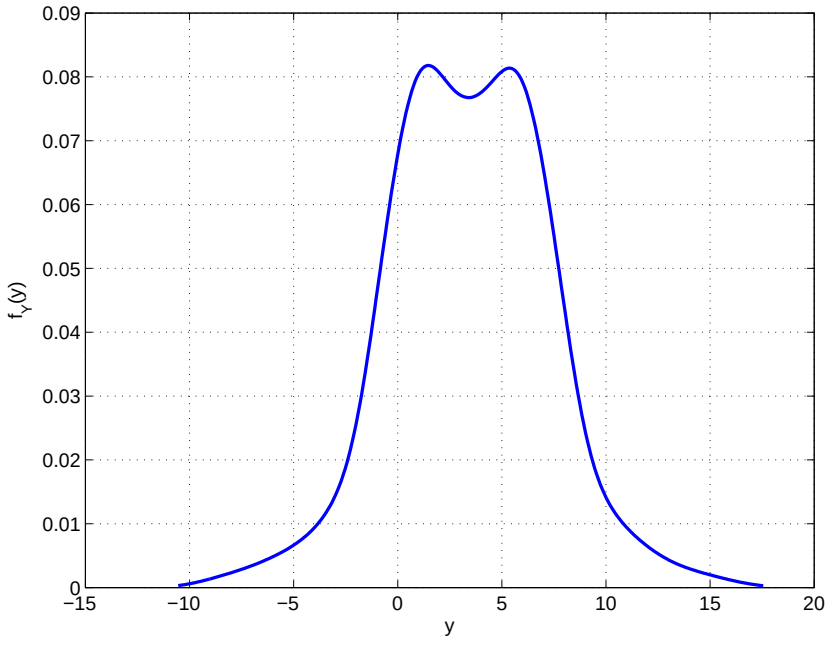


Figure 5.2: *Ishigami function - Probability density function obtained by a kernel method*

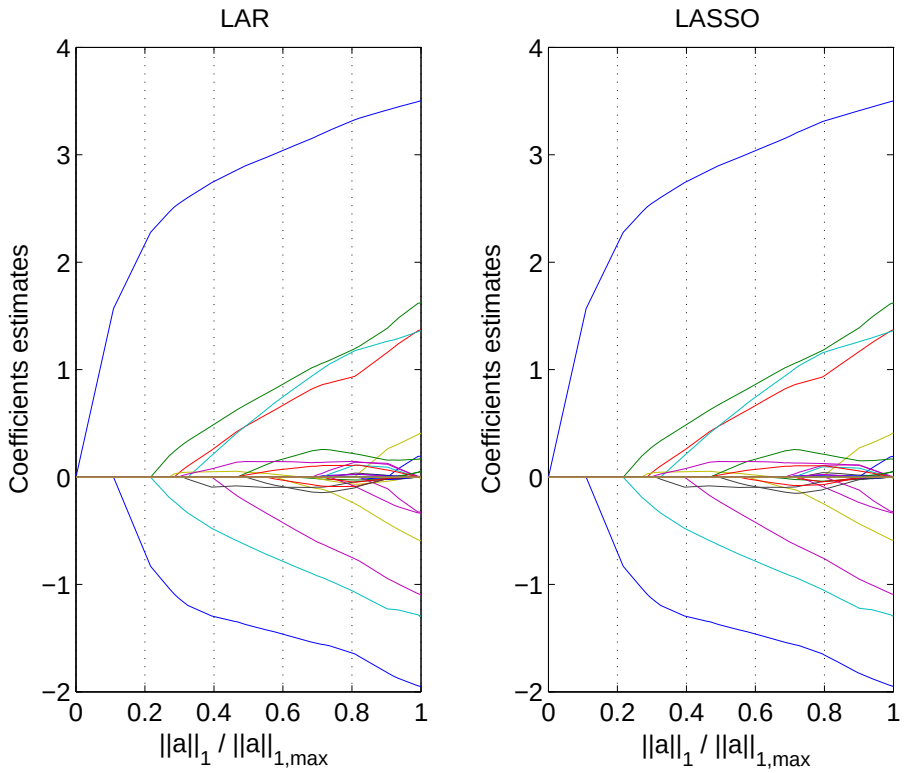


Figure 5.3: *Ishigami function - Profiles of the PC coefficients estimates* $\hat{\boldsymbol{a}}$ *based on LAR and LASSO*

2.7 Dantzig selector

An alternative variable selection method has received much interest lately, namely the *Dantzig selector* (Candes and Tao, 2007). The method is based on the following constrained optimization problem:

$$\text{Minimize} \qquad \left\| \boldsymbol{\Psi}^{\mathsf{T}}(\mathcal{Y} - \boldsymbol{\Psi}\boldsymbol{a}) \right\|_{\infty} \qquad \text{subject to} \qquad \|\boldsymbol{a}\|_1 \leq t \tag{5.8}$$

In a similar fashion to LARS, the Dantzig selector performs variable selection, *i.e.* it sets some coefficient estimates exactly equal to zero.

For any given $t \geq 0$, the problem (5.8) can be solved using a standard linear programming procedure. This is more efficient that the native LASSO which requires quadratic programming, but less than LAR. As for the original LASSO, obtaining the entire coefficient path may be computationally expensive though since it requires solving the optimization problem for many values of the tuning parameter t. James et al. (2008) proposed recently a LAR-type procedure in order to produce the entire coefficient path of the Dantzig selector in one shot. The algorithm is referred to as *DASSO* for *DAntzig Selector with Sequential Optimization*. However, numerical studies carried out in Efron et al. (2007); Meinshausen et al. (2007) show that LASSO performs as well as or better than the Dantzig selector in terms of prediction accuracy and model selection.

2.8 Conclusion

Various methods for performing variable selection have been reviewed. A special interest is devoted to LAR since it provides a set of sparse metamodels at the cost of an ordinary least-square regression. In particular, this method is well suited to situations in which the number of predictors is of similar size as the number of observations N, or even possibly significantly larger than N. This fully matches our objective of minimizing the number of calls to the possible computationally demanding model. Moreover, it has to be noticed that LAR is a non-parametric method in that it does not feature any tuning parameter. This makes LAR an attractive approach with respect to the cut-off algorithm described in the previous chapter.

It has been shown that a slight modification of LAR provides exactly the same results as the LASSO procedure. As it is believed that both variants yield similar solutions, only the native LAR algorithm will be considered in the following since it has a speed advantage over LASSO. Indeed, LAR takes only $m \equiv \min(P, N-1)$ steps, whereas LASSO may take more iterations since it allows the predictors to be discarded and to be introduced later again into the metamodel.

3 Criteria for selecting the optimal LARS metamodel

The LAR algorithm provides a large set of possible vectors of PC coefficients. Of course we want to retain a single *optimal* set in order to obtain predictions of the model response. Several rules have been proposed to this end.

3.1 Mallows' statistic C_p

The use of the so-called *Mallows' statistic* C_p (Mallows, 1973) has been recommended in Efron et al. (2004) as a criterion for selecting an optimal LAR solution, in an experimental context. The authors assume that the model response $\mathcal{M}(\boldsymbol{X})$ is random *conditionally to* $\boldsymbol{X}$, *i.e.* that $(\mathcal{M}(\boldsymbol{X})|\boldsymbol{X})$ is a random variable with mean μ and standard deviation σ. In this setup the Mallows' statistic is defined by:

$$C_p \equiv \frac{\|\mathcal{Y} - \hat{\mu}\|_2^2}{\sigma^2} - N + 2\,k \tag{5.9}$$

where $\hat{\mu}$ is the least-square regression-based estimate of the mean μ and k is the number of LAR steps. This formula originally applies only to LAR. An extension to LASSO is proposed in Zou et al. (2007). In the latter reference, the use of AIC and BIC criteria has been also discussed. The C_p-based criterion has the great advantage of not requiring any further calculations beyond those used for obtaining the LAR estimates.

However such a rule cannot be applied in our context since the model response $\mathcal{M}(\boldsymbol{X})$ is assumed to be *deterministic* given $\boldsymbol{X}$, *i.e.* $\sigma = 0$. Consequently other criteria are investigated in the sequel.

3.2 Cross-validation

The use of cross-validation for selecting a LARS solution has been suggested in Madigan and Ridgeway (2004). It is recalled that the so-called ν-*fold cross-validation* technique consists in splitting the data $\mathcal{Z} \equiv \{\mathcal{X}, \mathcal{Y}\}$ (*i.e.* the experimental design plus the corresponding model evaluations) into ν subsamples $\{\mathcal{Z}_1, \ldots, \mathcal{Z}_\nu\}$ of nearly equal size. One often sets ν equal to 10 in practice. The cross-validation selection of the optimal LARS solution is as follows:

1. For $i = 1, \ldots, \nu$:
   (a) Run the LARS procedure (Section 2.6.1) from the reduced data set $\mathcal{Z} \setminus \mathcal{Z}_i$ in order to build up a sparse PC approximation. Denoting the number of iterations by N_{iter}, one obtains a set of solution coefficients $\{\hat{\boldsymbol{a}}_i^{(1)}, \ldots, \hat{\boldsymbol{a}}_i^{(N_{iter})}\}$ with increasing $\mathcal{L}^1$-norm.
   (b) Compute the residual sums of squares $\{Err_i^{(1)}, \ldots, Err_i^{(N_{iter})}\}$ of the various coefficients estimates on the validation sets $\{\mathcal{Z}_1, \ldots, \mathcal{Z}_\nu\}$, respectively.
2. For $j = 1, \ldots, N_{iter}$: compute the mean residual sum of squares $\bar{Err}^{(j)} \equiv 1/\nu \sum_{i=1}^{\nu} Err_i^{(j)}$.
3. Find the optimal LARS step $j^* = \arg\min_j \bar{Err}_j$.
4. Run the LARS procedure from the whole data set $\mathcal{Z}$. One gets a set of solution coefficients $\{\hat{\boldsymbol{a}}^{(1)}, \ldots, \hat{\boldsymbol{a}}^{(N_{iter})}\}$.
5. Eventually return the set of coefficients estimates $\hat{\boldsymbol{a}}^{(j^*)}$.

This cross-validation scheme dedicated to the selection of the optimal LAR solution will be denoted by CV from now on.

In contrast to the C_p-based criterion described in Section 3.1, CV may be applied in our context of a deterministic model function $\mathcal{M}$. The procedure may be time-consuming though since it requires $(\nu + 1)$ calls to the LAR procedure. This may lead to a significant computational cost when applying an iterative strategy, which is the scope of the current chapter. A modified cross-validation scheme is proposed in the next section in order to overcome this difficulty.

3.3 Modified cross-validation scheme

Another cross-validation scheme is proposed that only requires a single call to the *hybrid* LAR procedure. This method is referred to as *modified cross-validation* (MCV) in the following. It

is recalled that hybrid LAR provides a set of metamodels $\{\widehat{\mathcal{M}}_{\mathcal{A}^{(1)}}, \ldots, \widehat{\mathcal{M}}_{\mathcal{A}^{(N_{iter})}}\}$ (where N_{iter} denotes the number of iterations) in two steps:

- perform variable selection using the original LAR procedure;
- compute the coefficients associated with the retained predictors by ordinary least-square regression.

The MCV strategy is designed as follows:

- run the LAR procedure once and for all;
- recompute the coefficients of each produced sparse metamodel by least-square regression;
- assess each metamodel using a cross-validation procedure;
- eventually retain the metamodel associated with the lowest error estimate.

It may be noticed that in contrast to the CV approach, MCV provides directly an estimate of the approximation error of the LAR-based PC approximations.

As suggested in Chapter 4, Section 3, *leave-one-out* cross-validation is employed for assessing the various sparse metamodels obtained by LAR. Indeed, it is recalled that this method can be computed analytically and requires no further regression calculations. The leave-one-out error estimate reads:

$$Err_{LOO} \;=\; \frac{1}{N}\sum_{i=1}^{N}\left(\frac{\mathcal{M}(\boldsymbol{x}^{(i)}) - \widehat{\mathcal{M}}_{\mathcal{A}^{(k)}}(\boldsymbol{x}^{(i)})}{1-h_i}\right)^2 \tag{5.10}$$

where h_i is the i-th diagonal term of the matrix $\boldsymbol{\Psi}_{\mathcal{A}^{(k)}}(\boldsymbol{\Psi}^{\mathsf{T}}_{\mathcal{A}^{(k)}}\boldsymbol{\Psi}_{\mathcal{A}^{(k)}})^{-1}\boldsymbol{\Psi}^{\mathsf{T}}_{\mathcal{A}^{(k)}}$. The relative leave-one-out error is given by:

$$\varepsilon_{LOO} \;\equiv\; \frac{Err_{LOO}}{\hat{\mathbb{V}}[\mathcal{Y}]} \tag{5.11}$$

where $\hat{\mathbb{V}}[\mathcal{Y}]$ denotes the empirical variance of the response sample $\mathcal{Y}$. As this error estimate may be too optimistic, one rather uses the following *corrected* relative leave-one-out error (Chapelle et al., 2002):

$$\varepsilon^{*}_{LOO} \;\equiv\; \varepsilon_{LOO} \;\times\; T(P,N) \tag{5.12}$$

where

$$T(P,N) \;\equiv\; \left(1-\frac{P}{N}\right)^{-1}\;\left(1 \;+\; \mathrm{tr}\big((\boldsymbol{\Psi}^{\mathsf{T}}_{\mathcal{A}^{(k)}}\boldsymbol{\Psi}_{\mathcal{A}^{(k)}})^{-1}\big)\right) \tag{5.13}$$

3.4 Numerical examples

The MCV strategy for selecting the optimal LAR solution is now compared to CV on two numerical examples, namely the Ishigami function and the Sobol' function.

3.4.1 Ishigami function

Let us consider the Ishigami function which has been already studied in Section 2.6.3:

$$Y = \sin X_1 + 7\sin^2 X_2 + 0.1X_3^4 \sin X_1 \tag{5.14}$$

where the X_i's $(i = 1, ..., 3)$ are independent random variables that are uniformly distributed over $[-\pi, \pi]$. The model response is approximated by a tenth-order PC expansion made of normalized Legendre polynomials which contain $\binom{10+3}{3} = 286$ terms. The PC coefficients are computed by LAR using two quasi-random experimental designs of sizes $N = 75$ and $N = 100$. The optimal solution is alternatively selected using CV and MCV. The LAR coefficients paths are depicted in Figure 5.4 together with the optimal solutions.

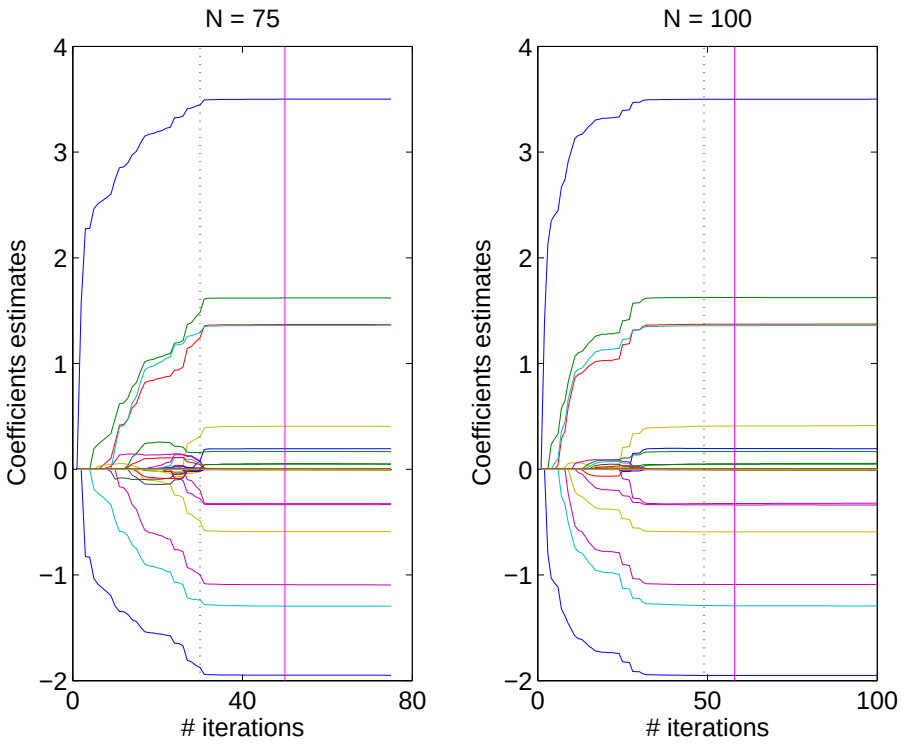


Figure 5.4: *Ishigami function - LAR coefficients paths using two quasi-random experimental designs of size $N = 75$ and $N = 100$. Dotted (resp. solid) vertical lines represent the optimal solution obtained by the classical (resp. modified) cross-validation scheme.*

It is observed that MCV yields less sparse solutions than CV, with 49 non-zero terms instead of 30 when $N = 75$. The number of non-zero coefficients is 57 instead of 49 when $N = 100$. The accuracy of the "optimal" metamodels is assessed by the following empirical relative $\mathcal{L}^2$-error:

$$\hat{\varepsilon} \equiv \frac{\sum_{i=1}^{\mathcal{N}} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \widehat{\mathcal{M}}_{\mathcal{A}}(\boldsymbol{x}^{(i)})\right)^2}{\sum_{i=1}^{\mathcal{N}} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \bar{y}\right)^2} \quad , \qquad \mathcal{N} = 50,000 \tag{5.15}$$

where the $\boldsymbol{x}^{(i)}$'s are random realizations of the input random vector $\boldsymbol{X}$, and $\bar{y}$ is defined by:

$$\bar{y} \equiv \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \mathcal{M}(\boldsymbol{x}^{(i)}) \tag{5.16}$$

The results are gathered in Table 5.1. It appears that the modified technique yields more accurate metamodels than the classical one. In particular, for $N = 75$, MCV provides a relative error with two orders of magnitude less.

Table 5.1: *Ishigami function - Assessment of the classical and modified cross-validation schemes for selecting the optimal LAR solutions*

	Original cross-validation		Modified cross-validation	
	$N = 75$	$N = 100$	$N = 75$	$N = 100$
Relative $\mathcal{L}^2$-error	$1.3 \cdot 10^{-3}$	$1.3 \cdot 10^{-5}$	$2.3 \cdot 10^{-5}$	$1.2 \cdot 10^{-5}$
# non-zero coefficients	30	49	49	57

The methods are now applied to an analytical example involving $M = 8$ input random variables, namely the Sobol' function.

3.4.2 Sobol' function

Let us consider now the Sobol' function already studied in Chapter 4, Section 5:

$$Y \equiv \mathcal{M}(\boldsymbol{X}) = \prod_{i=1}^{M} \frac{|4X_i - 2| + c_i}{1 + c_i} \tag{5.17}$$

where the input variables $X_i, i = 1, \ldots, M$ are uniformly distributed over $[0, 1]$ and c_i are non negative constants. The sensitivity indices of Y can be derived analytically. For numerical application, the number of input variables M is set equal to 8 and one selects $\boldsymbol{c} = \{1, 2, 5, 10, 20, 50, 100, 500\}^{\mathsf{T}}$.

According to the results obtained in Chapter 4, Section 5, the model response is approximated by a *hyperbolic* PC expansion of the form:

$$Y \approx \mathcal{M}_{\mathcal{A}_q^{M,p}}(\boldsymbol{X}) \equiv \sum_{\boldsymbol{\alpha} \in \mathcal{A}_q^{M,p}} a_{\boldsymbol{\alpha}} \, \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{5.18}$$

where

$$\mathcal{A}_q^{M,p} \equiv \left\{ \boldsymbol{\alpha} \in \mathbb{N}^M \; : \; \|\boldsymbol{\alpha}\|_q \equiv \left(\sum_{i=1}^{M} \alpha_i^q \right)^{1/q} \leq p \right\} \tag{5.19}$$

The PC degree p and the parameter q are respectively set equal to 20 and 0.4. Thus the total number of predictors is $P \equiv \mathrm{card}(\mathcal{A}_{0.4}^{8,20}) = 917$. The PC coefficients are estimated by LAR using

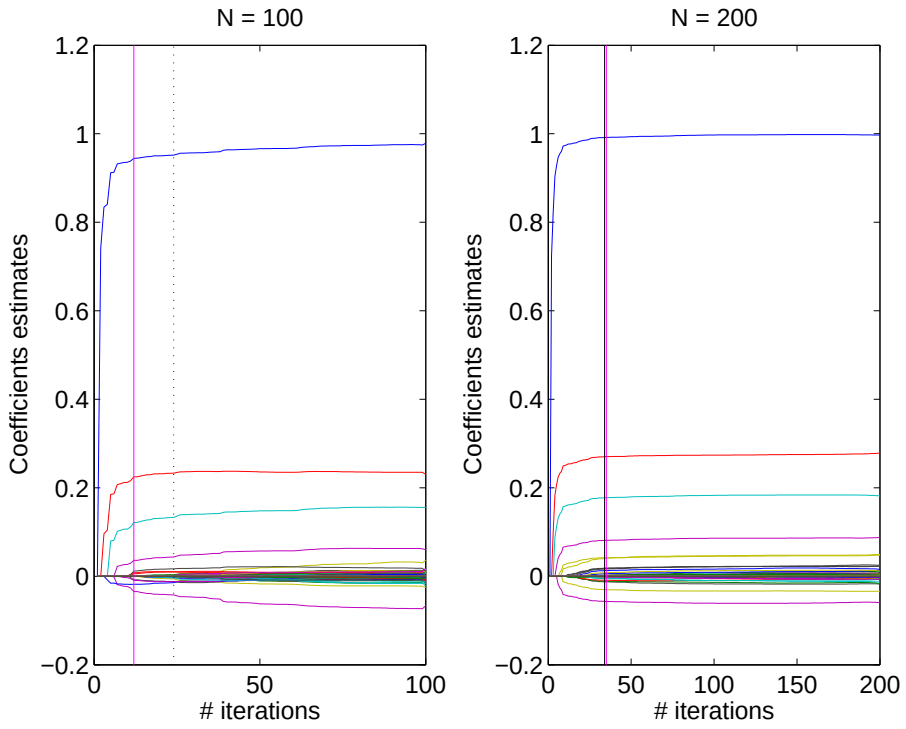


Figure 5.5: *Sobol' function - LAR coefficients paths using two quasi-random experimental designs of sizes $N = 100$ and $N = 200$. Dotted (resp. solid) vertical lines represent the optimal solution obtained by the classical (resp. modified) cross-validation scheme.*

two quasi-random experimental designs of size $N = 100$ and $N = 200$, respectively. The LAR coefficients paths are depicted in Figure 5.5 together with the optimal solutions.

It appears that MCV selects a sparser solution than CV for $N = 100$. Both strategies provide the same solution for $N = 200$. The accuracy of the "optimal" metamodels is reported in Table 5.2. It is observed that MCV yields slightly more accurate metamodels than CV in this second example.

Table 5.2: *Sobol' function - Assessment of the classical and modified cross-validation schemes for selecting the optimal LAR solutions*

	Original cross-validation		Modified cross-validation	
	$N = 100$	$N = 200$	$N = 100$	$N = 200$
Relative $\mathcal{L}^2$-error	0.16	0.02	0.14	0.01
# non-zero coefficients	24	34	11	34

From the two analytical examples it appears that the modified cross-validation scheme provides at least as accurate results as the original one. Hence this strategy may be used in order to perform an efficient selection of the optimal LAR solution. Indeed, it only requires a single call to the LAR algorithm. Using the MCV criterion, an adaptive LAR procedure may be devised

in order to iteratively enrich the basis of the PC approximation.

4 Basis-adaptive LAR algorithm to build up a sparse polynomial chaos approximation

4.1 Basis-adaptive LAR algorithm using a fixed experimental design

A limitation of LAR lies in the requirement of an *a priori* truncation set $\mathcal{A}$. To circumvent this difficulty, one proposes a procedure for enriching iteratively the multi-index set of the PC approximation, *i.e.* the set of active basis functions. In this section, a given set of model evaluations (*i.e.* a fixed experimental design) $\mathcal{X} \equiv \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}^\mathsf{T}$ is assumed. Without loss of generality, one considers PC approximations of the model response based on *hyperbolic multi-index sets* $\mathcal{A}_q^{M,p}$, that is:

$$Y \approx \mathcal{M}_{\mathcal{A}_q^{M,p}}(\boldsymbol{X}) \equiv \sum_{\boldsymbol{\alpha} \in \mathcal{A}_q^{M,p}} a_{\boldsymbol{\alpha}} \, \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{5.20}$$

where

$$\mathcal{A}_q^{M,p} \equiv \left\{ \boldsymbol{\alpha} \in \mathbb{N}^M \; : \; \|\boldsymbol{\alpha}\|_q \equiv \left(\sum_{i=1}^{M} \alpha_i^q \right)^{1/q} \leq p \right\} \tag{5.21}$$

The reader is referred to Chapter 4 for more details on the multi-index sets. It is recalled that setting the parameter q equal to 1 yields the usual truncation scheme, in which only those basis polynomials $\psi_{\boldsymbol{\alpha}}$ with total degree less than p are retained. Other kinds of multi-index sets may be also used, such as the *low-rank sets*.

The computational flowchart of the proposed basis adaptive LAR procedure is sketched in Figure 5.6. LAR is first applied to a first-order PC approximation corresponding to $\mathcal{A}_q^{M,1}$. The selection of the best LAR metamodel is performed using the modified cross-validation scheme outlined in Section 3.3. The associated optimal subset of multi-indices (resp. error estimate) is stored in the variable $\mathcal{A}^{(1)}$ (resp. $\varepsilon^{(1)}$). Note that $\varepsilon^{(1)}$ has been already computed in order to select the optimal LAR-based PC approximation ($\varepsilon^{(1)}$ is the *corrected leave-one-out error estimate* of the metamodel $\mathcal{M}_{\mathcal{A}^{(1)}}(\boldsymbol{X})$). One sets $\varepsilon^* \equiv \varepsilon^{(1)}$ and one denotes by $\mathcal{A}^*$ the corresponding set of multi-indices. If ε^* is less than a prescribed target error ε_{tgt}, one stops the algorithm. Otherwise one iterates with the second-order PC approximation related to $\mathcal{A}_q^{M,2}$ and sets $\varepsilon^* \equiv \min(\varepsilon^*, \varepsilon^{(2)})$, and so on. Thus the sparse PC approximation is sought among the best LAR metamodels for each degree $p = 1, \ldots, p_{max}$.

It is possible that the approximation error $\varepsilon^{(p)}$ increases from a given value of the degree p. This might be due to an *overfitting* situation, in which the number of accepted predictors gets too important with respect to the size N of the experimental design $\mathcal{X}$. In order to avoid

this phenomenon, the following heuristic criterion is introduced: if the approximation error $\varepsilon^{(p)}$ increases twice in a row (say $\varepsilon^{(p)} \geq \varepsilon^{(p-1)} \geq \varepsilon^{(p-2)}$), then the algorithm is stopped, returning a warning message of possible overfitting.

The optimal subset $\mathcal{A}^*$ is eventually retained. The coefficients of the related sparse PC approximation $\mathcal{M}_{\mathcal{A}^*}(\boldsymbol{X})$ are computed by ordinary least-square regression.

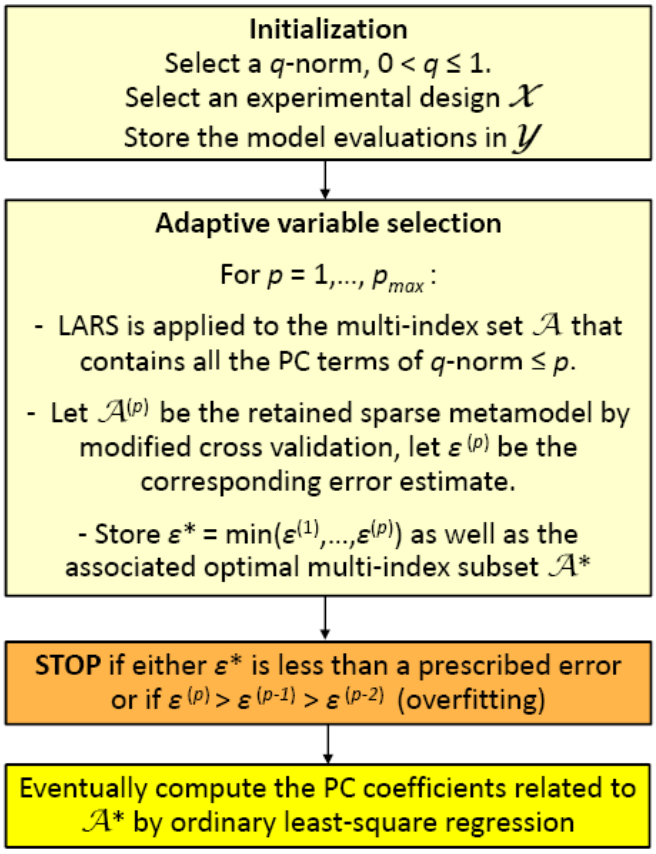


Figure 5.6: *Computational flowchart of the basis-adaptive LAR procedure for building up an adaptive sparse polynomial chaos expansion*

Numerical example

The basis adaptive LAR procedure is illustrated by the example of the Sobol' function, which involves $M = 8$ input random variables. The model response $\mathcal{M}(\boldsymbol{X})$ is approximated by hyperbolic PC expansions, *i.e.* PC expansions associated with a multi-index set of the form $\mathcal{A}_q^{M,p}$. The parameter q is set equal to 0.4 since this provides good results as shown in Chapter 4, Section 5. Two quasi-random experimental designs of sizes $N = 100$ and $N = 300$ are considered. The significant terms in the PC approximations are retained using the proposed adaptive procedure based on LAR. The target accuracy ε_{tgt} is set equal to 0 so that the maximum accuracy which can be reached using the available experimental designs will be obtained.

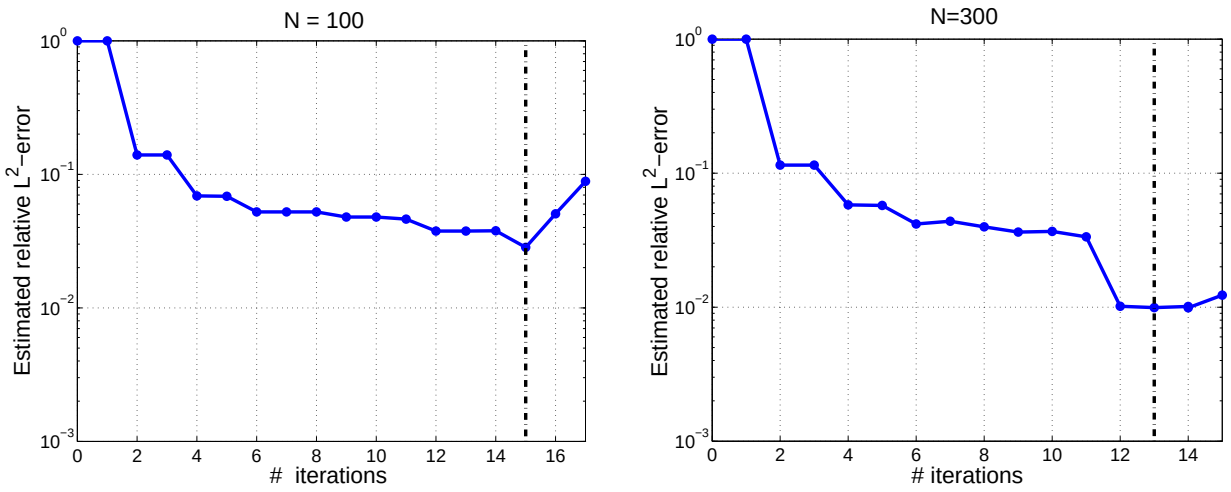


Figure 5.7: *Sobol' function - Building of sparse PC approximations using the basis adaptive LAR procedure (quasi-random experimental designs of sizes $N = 100$ and $N = 300$ are used). Vertical lines indicate the optimal metamodels.*

The evolution of the error estimates with the algorithm iterations are sketched in Figure 5.7. It is observed that for each experimental design, the estimate of the approximation error decreases down to a point that corresponds to the optimal solution. Overfitting occurs when performing further adaptive LAR iterations, hence the algorithm stops. Note that each iteration is related to an incrementation of the PC degree p.

Let us define the *index of sparsity* of a metamodel with index set $\mathcal{A}$ by $IS \equiv \text{card}(\mathcal{A})/\text{card}(\mathcal{A}^{M,p})$, where $\mathcal{A}^{M,p} \equiv \mathcal{A}_1^{M,p}$ is the index set corresponding to a full PC expansion of degree p. As the sparsity of the PC expansion depends both on the nature of the selected index set (*e.g.* hyperbolic or low-rank index set) and on the predictors selection that is achieved by LAR, the index of sparsity is split into two factors as follows:

$$IS = IS_1 \times IS_2 \tag{5.22}$$

where $IS_1 \equiv \text{card}(\mathcal{A}_q^{M,p})/\text{card}(\mathcal{A}^{M,p})$ and $IS_2 \equiv \text{card}(\mathcal{A})/\text{card}(\mathcal{A}_q^{M,p})$. The quantities IS_1 and IS_2 correspond to the sparsity due to the choice of the index set and the sparsity due to the LAR procedure, respectively.

The accuracy and the indices of sparsity of the two obtained sparse PC expansions are reported in Table 5.3 together with the reference $\mathcal{L}^2$-errors. As expected, increasing the size N of the experimental design yields a more accurate and less sparse PC approximation. Morevover, it is observed that the error estimate based on corrected leave-one-out cross-validation are very close to the reference error.

Table 5.3: *Sobol' function - Accuracy and sparsity of the sparse PC expansions obtained by the basis adaptive LAR procedure*

	$N = 100$	$N = 300$
Estimated $\mathcal{L}^2$-error	0.0284	0.0099
Reference $\mathcal{L}^2$-error	0.0256	0.0097
PC degree	15	13
IS_1	$457/490,314 \approx 9 \times 10^{-4}$	$329/203,490 \approx 2 \times 10^{-3}$
IS_2	$25/457 \approx 5\%$	$50/329 \approx 15\%$

4.2 Basis-adaptive LAR algorithm using a sequential experimental design

The algorithm proposed in the last subsection allows one to detect automatically the significant terms in the PC expansion. However it is based on a given experimental design whose size N is arbitrary. This problem is tackled by proposing a version of the procedure in which the design is automatically enriched, so that the target error may be reached. In this purpose, *sequential experimental designs* may be used, such as Monte Carlo sampling, Nested Latin Hypercube sampling (NLHS) or quasi-Monte Carlo sampling (QMC), see Section 4.3 for more details.

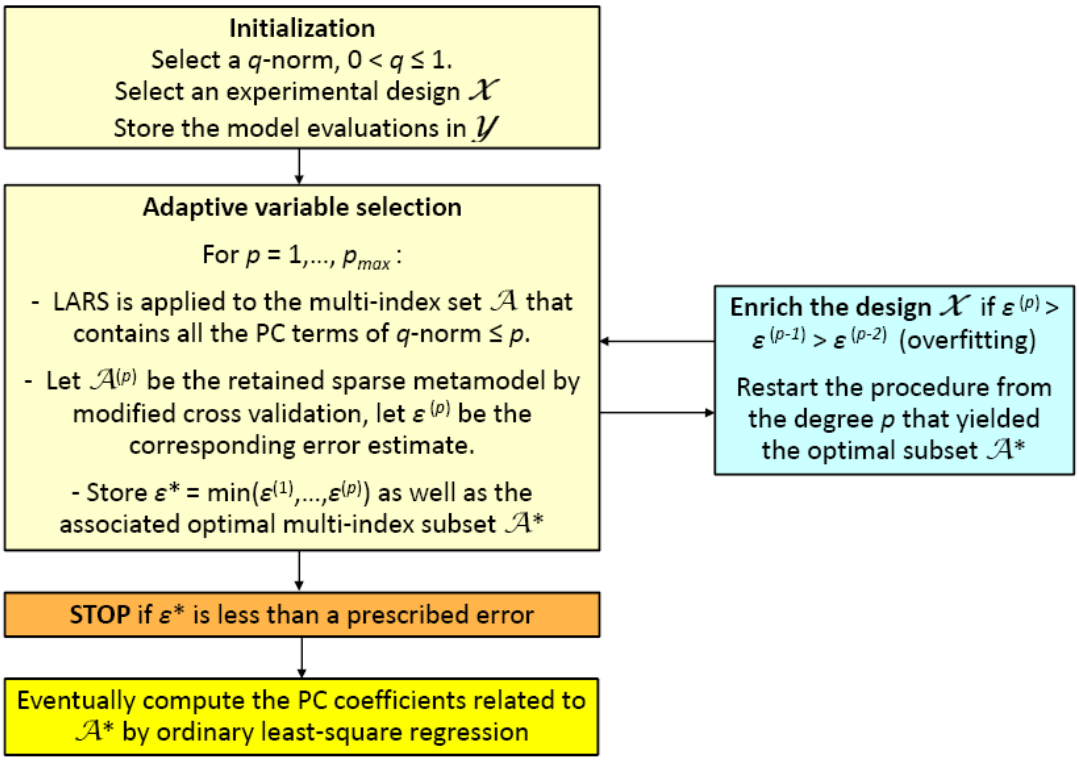


Figure 5.8: *Computational flowchart of the basis and experimental design adaptive LAR procedure for building up an adaptive sparse polynomial chaos expansion*

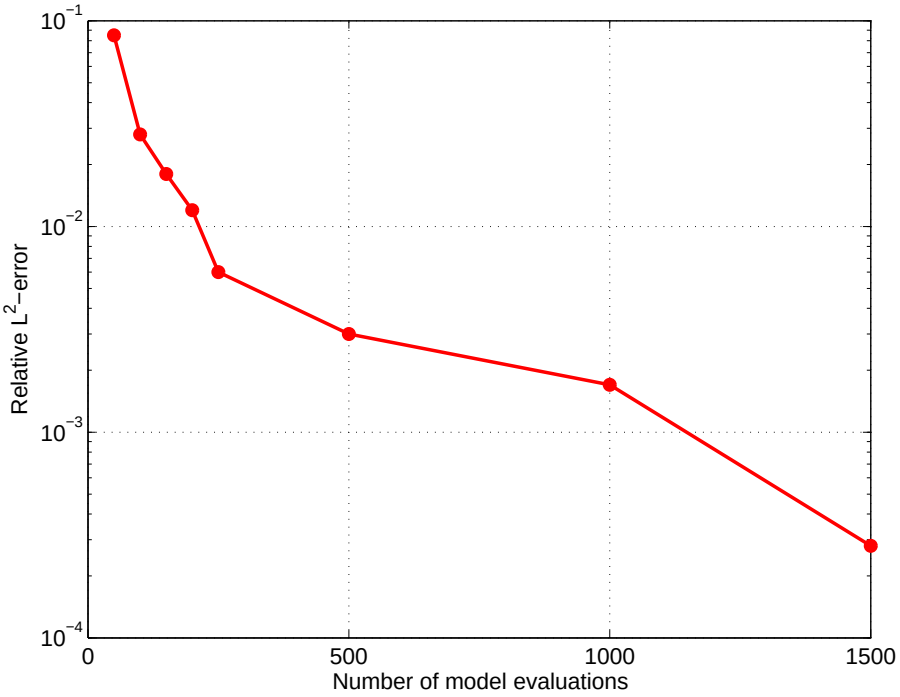


Figure 5.9: *Sobol' function - Convergence of the sparse PC approximation based on the basis and design adaptive LAR procedure (a sequential quasi-random experimental design is used)*

Using a sequential sampling scheme, a modification of the basis adaptive LAR procedure outlined in Section 3.4 is devised. As soon as overfitting is detected in the adaptive LAR iterations (*i.e.* when the error estimate increases twice in a row), the experimental design is enriched. Note that the number of additional points is fixed *a priori*. Then the procedure is restarded from the PC degree p that yielded the best metamodel prior to complementing the design. The computational flowchart of the algorithm is sketched in Figure 5.8.

Numerical example

Let us consider again the Sobol' function. The model response is approximated by a sparse PC expansion using the basis- and design-adaptive LAR algorithm. As in the previous example the original LAR scheme is used rather than LASSO. Moreover, the PC approximation is assumed to be hyperbolic setting the parameter q of the truncation norm equal to 0.4. A sequential quasi-random experimental design is used, with initial size $N_{ini} = 50$. The number of additional points when enriching the design is set equal to 20. The target relative approximation error ε_{tgt} is set equal to 0.001.

The convergence of the sparse PC approximation with respect to the number of points in the experimental design is depicted in Figure 5.9. The obtained sparse PC expansion has degree $p = 16$ and contains 57 non zero coefficients, hence an index of sparsity $IS = 57/669 \approx 8\%$.

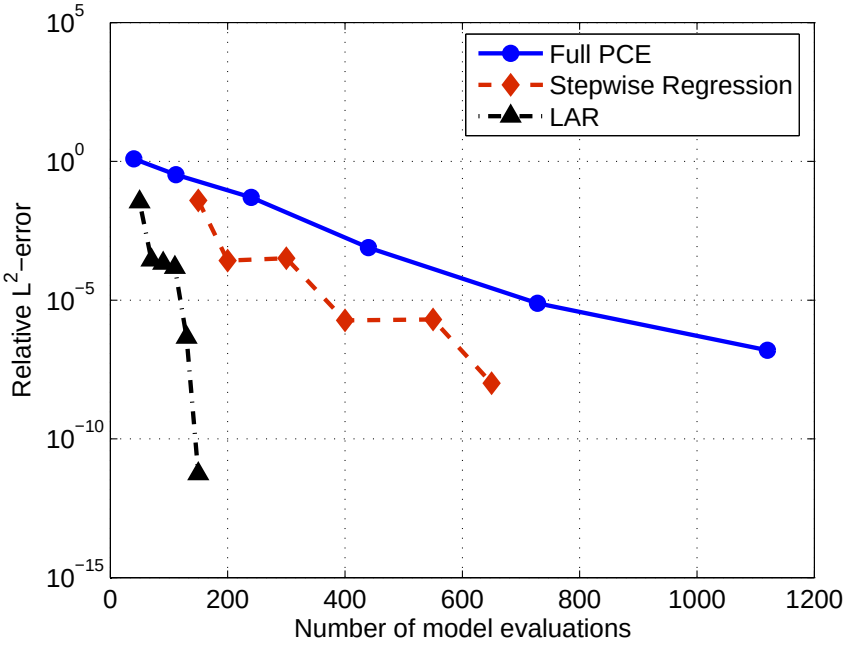


Figure 5.10: *Ishigami function - Convergence rates of full and sparse polynomial chaos expansions (quasi-random experimental designs are used)*

The reference relative $\mathcal{L}^2$-error ε_{REF} of the metamodel is computed by Monte Carlo simulation. One gets $\varepsilon_{REF} = 0.007$.

5 Illustration of convergence

The Ishigami function (Section 2.6.4) is considered again:

$$Y = \sin X_1 + 7 \sin^2 X_2 + 0.1 X_3^4 \sin X_1 \quad , \quad X_i \sim \mathcal{U}([-\pi, \pi]) \quad , \quad i = 1, 2, 3 \tag{5.23}$$

The convergence of the three following methods is investigated:

- classical full PC expansions with degree varying from 3 to 13 ($N = 2P$ model evaluations are performed, where P denotes the PC size);
- sparse PC expansions based on the stepwise regression algorithm presented in Chapter 4;
- sparse PC expansions based on the adaptive LAR procedure.

Whatever the computational scheme, the PC coefficients are computed from a quasi-random experimental design. The convergence rate of the metamodels is reported in Figure 5.10.

As expected, the full PC expansion has the slowest rate of convergence. In contrast, the efficiency of the LAR-based approach is impressive. Indeed, it yields a relative error less than 10^{-10} using

about 150 model evaluations. This shows how the LAR procedure can take advantage of the genuine sparsity of the model response, especially in the case of a smooth function, say of class $\mathcal{C}^\infty$.

6 Application to the analytical Sobol' function

6.1 Convergence rate of the LAR-based sparse PC approximations

Let us consider again the Sobol' function:

$$Y \equiv \mathcal{M}(\boldsymbol{X}) = \prod_{i=1}^{M} \frac{|4X_i - 2| + c_i}{1 + c_i} \tag{5.24}$$

The model response Y is approximated by various *sparse* PC expansions that are built using the proposed iterative procedure.

Of interest is the comparison of various PC approximations, namely:

- usual full PC approximations, for a degree p varying from 3 to 5 (the PC coefficients are computed by regression using $N = 2P$ model evaluations, where P denotes the PC size);
- full hyperbolic PC approximations (the parameter q of the norm is set equal to 0.4) ;
- a LAR-based sparse hyperbolic PC representations for a target accuracy ε_{tgt} which is progressively decreased.

The coefficients of the PC expansions are evaluated from a quasi-random experimental design. The convergence rates of the various approaches are depicted in Figure 5.11.

The usual full PC approximation appears to be the least efficient approach. As already noted in the previous chapter, a full hyperbolic metamodel is much more efficient. Indeed, the hyperbolic PC expansion reaches a 1%-accuracy using about $1,000$ model evaluations whereas the corresponding relative error is equal to 5% when using a classical PC representation. On the other hand, the sparse metamodel based on LAR noticeably outperforms the full approximations, reaching a relative error of 0.5% with a computational cost divided by 4 with respect to the full hyperbolic expansion.

6.2 Sensitivity analysis

The global sensitivity indices of the response of the Sobol' function are of interest. Estimates of the first-order and the total sensitivity indices are computed by post-processing several PC approximations, namely:

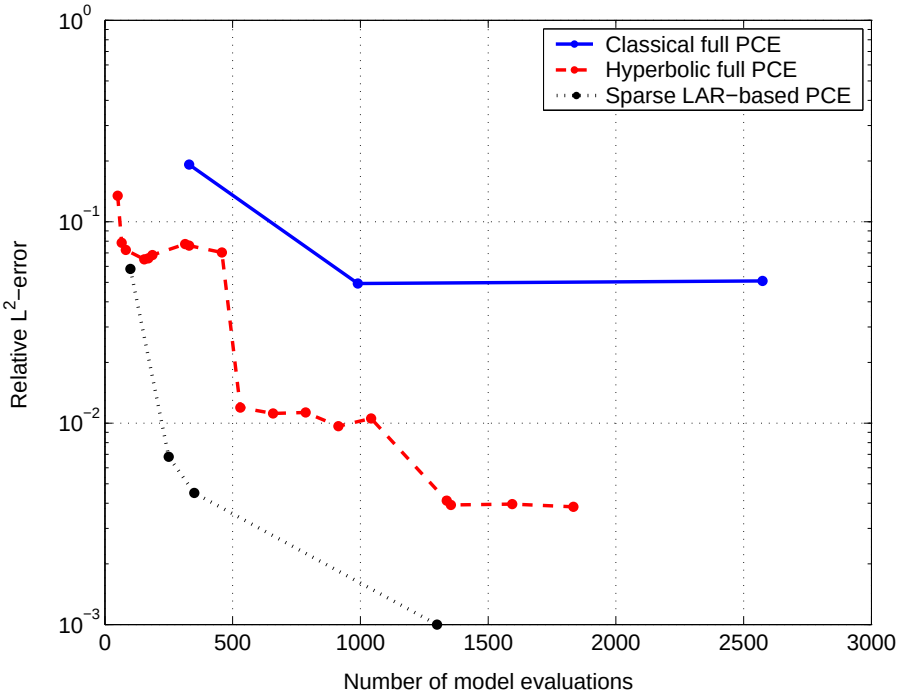


Figure 5.11: *Sobol' function - Convergence rates of full and LAR-based polynomial chaos expansions (quasi-random experimental designs are used)*

- a usual full third-order PC expansion;
- LAR-based hyperbolic sparse PC approximations (using $q = 0.4$) obtained by setting the target error ϵ_{tgt} equal to 0.05, 0.01 and 0.005.

The results are gathered in Table 5.4 together with the (analytical) reference values.

As expected, the accuracy of the estimates based on the sparse metamodels increases when the target error ε_{tgt} is decreased. Considering only those indices that are greater than 0.1, a maximal relative error of 5% is observed when setting ε_{tgt} equal to 0.05, using only 100 model evaluations. This error reduces to 2% and 1% when setting ε_{tgt} equal to 0.01 and 0.005, respectively. It has to be noticed that the sparse PC approximation corresponding to 0.005 yields more accurate estimates than the usual full third-order PC expansion, with a computational gain factor of 10.

6.3 Conclusion

A rapid survey of variable selection methods has been presented in this chapter. Even if the stepwise regression technique is a classical and simple to implement procedure, it may reveal quite greedy and unstable in general. The selection criteria that have been introduced in Chapter 4 have allowed one to make it stable and converging. However it is interesting to compare this classical least-square approach to more recent techniques such as Least Angle Regression (LAR).

Table 5.4: *Sobol' function - Estimates of the first-order and the total sensitivity indices by post-processing a full and LAR-based sparse PC approximations*

Sensitivity Index	Analytical	Full PCE's Classical	LAR-based sparse PCE's $\varepsilon_{tgt} = 0.05$	$\varepsilon_{tgt} = 0.01$	$\varepsilon_{tgt} = 0.005$
S_1	0.604	0.585	0.627	0.610	0.608
S_2	0.268	0.264	0.278	0.274	0.271
S_3	0.067	0.067	0.064	0.063	0.065
S_4	0.020	0.017	0.016	0.018	0.019
S_5	0.006	0.005	0.003	0.005	0.005
S_6	0.001	0.001	0.003	0.001	0.001
S_7	0.000	0.000	0.000	0.000	0.000
S_8	0.000	0.000	0.000	0.000	0.000
S_1^T	0.634	0.624	0.631	0.637	0.636
S_2^T	0.295	0.301	0.280	0.298	0.295
S_3^T	0.076	0.087	0.064	0.069	0.072
S_4^T	0.023	0.030	0.019	0.019	0.021
S_5^T	0.006	0.016	0.007	0.006	0.006
S_6^T	0.001	0.015	0.005	0.001	0.001
S_7^T	0.000	0.014	0.003	0.001	0.000
S_8^T	0.000	0.013	0.000	0.000	0.000
# model evaluations		2,574	100	250	350

The LAR algorithm allows the analyst to obtain a full set of solutions to a variable selection problem, with an increasing $\mathcal{L}^1$-norm. A slight modification of LAR produces exactly the same results as LASSO, which is a $\mathcal{L}^1$-penalized regression problem. For the sake of completeness, the Dantzig selector method has also been mentioned.

LAR has been applied to the problem of building up a *sparse* polynomial chaos (PC) approximation of the response of a model under consideration. This required a criterion in order to select the optimal LAR-based PC coefficients among the whole set of solutions. Two common rules have been described in this purpose, namely Mallows' C_p statistic and cross-validation (CV). However, the validity of the former criterion does not hold in our framework of deterministic models, and the latter may reveal time-consuming since it requires many calls to the LAR procedure. A modified cross-validation (MCV) scheme has been proposed to overcome this difficulty. It only requires a single call to the LAR procedure and relies upon error estimates that have been presented in the previous chapter. MCV offers the advantage to provide an estimate of the approximation error of the LAR metamodels in addition to selecting the best solution. MCV was observed to yield at least as good results as CV for two analytical examples

respectively involving 3 and 8 random variables. Both methods have been compared on three other benchmark problems not presented here and the same conclusion applies.

Using the LAR procedure together with MCV, a step-by-step algorithm has been devised for building up a sparse PC approximation of the model response by increasing iteratively the degree of the representation. A limitation of the procedure lies in the arbitrary size of the experimental design though. To circumvent this problem and following the approach developed in Chapter 4, Section 4.3, a variant has been developed that automatically enriches the design in order to avoid overfitting. The adaptive LAR procedure has been successfully applied to the global sensitivity analysis of the so-called Sobol' function, for which a computational gain factor of 10 has been observed with respect to an ordinary "full" third-order PC expansion.

It is worth mentioning that recent extensions of LAR and LASSO could be also integrated in our adaptive algorithm, such as a LAR formulation of the Dantzig selector (James et al., 2008), the adaptive LASSO (Zou, 2006) and the so-called *SCAD* (Smoothly Clipped Absolute Deviation)-penalty method (Fan and Li, 2001; Zou and Li, 2008). Moreover, the chosen method for variable selection could be complemented by a previous cleaning of a very large set of candidate predictors using the *Sure Independence Screening* approach (Fan and Li, 2006).

Chapter 6

Application to academic and industrial problems

1 Introduction

The previous two chapters have been devoted to the iterative building of sparse polynomial chaos (PC) approximations of the random response of a mathematical model. Two algorithms have been devised, namely stepwise regression (Chapter 4) and adaptive Least Angle Regression (LAR) (Chapter 5). Both procedures allow the analyst to determine accurate PC metamodels which only contains few coefficients. The latter may then be computed by means of a low number of possibly costly model evaluations.

In this section the stepwise regression and the LAR schemes are applied to various problems of uncertainty propagation. Five academic problems are addressed first, namely:

- Example #1: the analytical *Morris function* (20 random variables);
- Example #2: the maximum deflection of a truss structure (10 random variables);
- Example #3: the maximum top-floor displacement of a frame structure (21 correlated random variables);
- Example #4: the settlement of a foundation (38 random variables);
- Example #5: the bending of a simply supported beam (100 random variables).

Then an industrial problem in nuclear engineering that is of interest at the Research and Development Division of EDF is tackled. It deals with the analysis of the integrity of the reactor pressure vessel of a nuclear powerplant.

2 Academic problems

2.1 Methodology

The following sections are dedicated to the uncertainty, sensitivity and reliability analysis of five academic application examples. Estimates of the related quantities of interest, namely the statistical moments, the Sobol' indices and the probabilities of failure are obtained by post-processing sparse polynomial chaos (PC) expansions of the response of the model being studied. The stepwise regression and LAR methods are employed in order to build up these sparse metamodels. Several choices have to be made in this purpose.

First, we will only consider *hyperbolic* PC expansions of the form:

$$Y \approx \widehat{\mathcal{M}}(\boldsymbol{X}) \equiv \sum_{\boldsymbol{\alpha}\mathcal{A}_q^{M,p}} a_{\boldsymbol{\alpha}}\, \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{6.1}$$

where:

$$\mathcal{A}_q^{M,p} \equiv \left\{ \boldsymbol{\alpha} \in \mathbb{N}^M \,:\, \|\boldsymbol{\alpha}\|_q \equiv \left(\sum_{i=1}^{M} \alpha_i^q \right)^{1/q} \leq p \right\} \tag{6.2}$$

The parameter q will be set equal to 0.4. Indeed, such a choice has often led to good results, as shown in the illustrating example in Chapter 4, Section 2.2 and from the author's experience. The sparsity of the obtained sparse PC expansions will be quantified by means of *indices of sparsity* whose definition is recalled:

$$IS_1 \equiv \frac{\mathrm{card}(\mathcal{A}_q^{M,p})}{\mathrm{card}(\mathcal{A}^{M,p})} \tag{6.3}$$

$$IS_2 \equiv \frac{\mathrm{card}(\mathcal{A})}{\mathrm{card}(\mathcal{A}_q^{M,p})} \tag{6.4}$$

where $\mathcal{A}$ is the final index set that has been eventually returned by the algorithms. These quantities respectively correspond to the sparsity due to the choice of the hyperbolic index set and the sparsity due to the adaptive selection of the terms in the PC decomposition. Note that the "total" index of sparsity of a PC expansion with respect to a full representation of same maximal degree is $IS_1 \times IS_2$.

Second, the stepwise regression cut-off parameters θ_1, θ_2 for accepting or discarding the terms in the PC expansion will be chosen according to the thumb rule $\theta_1 = \theta_2 = 0.01 \times \varepsilon_{tgt}$, where ε_{tgt} denotes the target error of approximation. Whatever the approach for building a PC representation (may it be sparse or full), the PC coefficients will be systematically computed using an experimental design made of Sobol' quasi-random numbers. When applying one of the adaptive approaches (*i.e.* stepwise regression or LAR), one will employ a sequential design strategy. The initial size of the design (resp. the number of additional sample when detecting overfitting) will be set equal by default to $N_{ini} = 100$ (resp. $N_{add} = 100$), unless alternative choices are specified.

Third, the accuracy of the PC approximations will be quantified by the following reference relative error:

$$\varepsilon_{ref} \equiv \frac{\sum_{i=1}^{\mathcal{N}} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}_{\mathcal{A}}(\boldsymbol{x}^{(i)})\right)^2}{\sum_{i=1}^{\mathcal{N}} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \bar{y}\right)^2} \quad , \quad \mathcal{N} = 50,000 \tag{6.5}$$

where:

$$\bar{y} \equiv \frac{1}{N} \sum_{i=1}^{N} \mathcal{M}(\boldsymbol{x}^{(i)}) \tag{6.6}$$

and where the $\boldsymbol{x}^{(i)}$'s are randomly drawn from the distribution of the input random vector $\boldsymbol{X}$. Such a calculation will be only possible for those models that are relatively fast to evaluate, namely the problems #1-#3. *Ad-hoc* alternatives will be employed for Examples #4 and #5. Also, the PC-based Sobol' indices and probabilities of failure will be compared either to analytical solutions (when available) or to reference results based on crude Monte Carlo simulation.

2.2 Example #1: Analytical model - the Morris function

Let us consider the so-called Morris function which has been used as a benchmark in the context of sensitivity analysis (Morris, 1991; Saltelli et al., 2000; Blatman and Sudret, 2009b). This function involves 20 input variables and is defined by:

$$Y = \sum_{i=1}^{20} \beta_i w_i + \sum_{i<j}^{20} \beta_{ij} w_i w_j + \sum_{i<j<l}^{20} \beta_{ijl} w_i w_j w_l + \sum_{i<j<l<s}^{20} \beta_{ijls} w_i w_j w_l w_s \tag{6.7}$$

where

$$w_i = \begin{cases} 2\left(1.1X_i/(X_i+0.1) - 0.5\right) & \text{if } i = 3,5,7 \\ 2(X_i - 0.5) & \text{otherwise} \end{cases} \tag{6.8}$$

and the $\{X_i, i = 1,\ldots,20\}$ are uniformly distributed over $[0,1]$. The coefficients β_i are assigned as follows:

$$\begin{cases} \beta_i = 20 & \text{for } i = 1,\ldots,10 \\ \beta_{ij} = -15 & \text{for } i,j = 1,\ldots,6 \\ \beta_{ijl} = -10 & \text{for } i,j,l = 1,\ldots,5 \\ \beta_{ijls} = 5 & \text{for } i,j,l,s = 1,\ldots,4 \end{cases} \tag{6.9}$$

The remaining coefficients are defined by $\beta_i = (-1)^i$ and $\beta_{ij} = (-1)^{i+j}$. The probability density function of random variable Y is plotted in Figure 6.1. By construction of the function, the variables $X_{11} - X_{20}$ have a negligible influence onto the variance of the response Y.

We apply the polynomial chaos method in order to estimate the total Sobol' indices of Y. A full second-order PC expansion is considered first (the PC coefficients are computed using

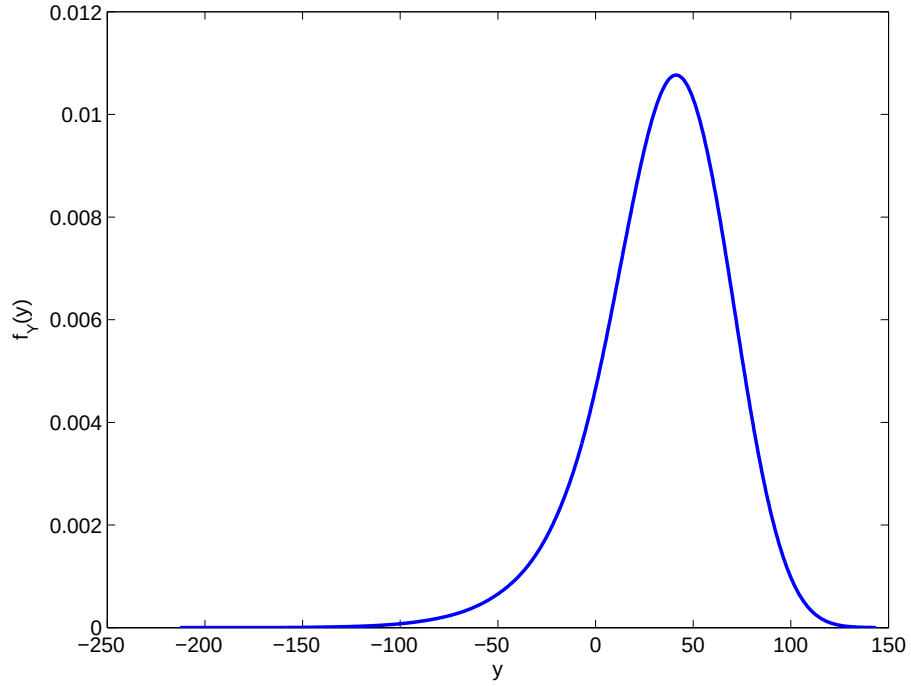


Figure 6.1: *Example #1: Morris function - Probability density function obtained using a kernel method*

an experimental design made of $N = 2P$ quasi-random numbers, where $P = \binom{20+2}{2} = 231$ denotes the number of unknown coefficients). This full representation is compared to two sparse approximations produced by the stepwise regression and the LAR algorithms. The target error is set equal to 0.1. Indeed it is believed that such a choice for the stopping criterion will provide a reasonable approximation at a quite low computational cost for sensitivity analysis. As the final number of model evaluations is expected to be relatively low, the number N_{add} of additional points in the sequential design is set equal to 50 rather than 100. For the sake of validation, reference values are computed by direct Monte Carlo simulation of the problem ($440,000$ samples are used, which corresponds to $20,000$ samples for the evalutation of each Sobol' index), and 95%-confidence intervals are provided by bootstrap ($1,000$ replicates are used).

The results are as expected insofar as the three PC expansions yield total estimates of the total Sobol' indices of the non-important variables $X_{11} - X_{20}$ that are less than 0.01. The results corresponding to the other (significant) indices are reported in Table 6.1 and depicted in Figure 6.2.

It appears that three groups of input variables (among the "significant" total Sobol' indices) may be distinguished, namely:

- a group of important variables: X_1, X_2, X_4;
- one variable with intermediate significance: X_9;

Table 6.1: *Example #1: Morris function - Estimates of significant total Sobol' indices when setting the target accuracy ε_{tgt} equal to* 0.1 *and the degree p of the full PC expansion to 2*

Variables	Total Sobol' indices			
	Reference 95%-CI	Full second-order-PCE	Stepwise	LAR
X_4	[0.244 , 0.270]	0.221	0.240	0.240
X_1	[0.242 , 0.267]	0.244	0.245	0.241
X_2	[0.240 , 0.268]	0.216	0.248	0.254
X_9	[0.140 , 0.161]	0.158	0.170	0.164
X_3	[0.102 , 0.119]	0.088	0.093	0.091
X_5	[0.100 , 0.117]	0.076	0.088	0.095
X_8	[0.091 , 0.109]	0.114	0.107	0.116
X_{10}	[0.090 , 0.107]	0.124	0.112	0.105
X_6	[0.084 , 0.100]	0.078	0.095	0.082
X_7	[0.063 , 0.077]	0.087	0.072	0.079
Number of runs	$440,000$	462	500	450
Relative $\mathcal{L}^2$-error		0.16	0.07	0.07
PC degree		2	8	7
Number of PC terms		231	122	91
Index of sparsity IS_1		-	4×10^{-4}	8×10^{-4}
Index of sparsity IS_2		-	9%	13%

- a group of little significant variables: $X_3, X_5 - X_{20}$.

It appears that the two sparse metamodels provide more accurate estimates of the Sobol' indices than the full second-order PC expansion, at a similar computational cost though. This is consistent with the fact that the approximation error of the sparse representations (0.07) is observed to be twice as less as the one of the full expansion (0.16). Thus the "sparse" approaches may be considered to be more efficient in this application. It is hard to say which of these two techniques performs best though. It may be noted that both sparse approximations have a similar level of sparsity, with an index IS_2 close to 10%, which shows that the reponse Y of the Morris function may be represented by means of a small number of terms.

As LAR has a slight advantage in terms of number of model evaluations (450 instead of 500), it is used again with a smaller target error (say $\varepsilon = 0.05$) in order to get more accurate results, and then to illustrate the convergence of the method. Such an "accurate" LAR-based metamodel is compared to a full third-order PC expansion. The results are presented in Table 6.2 and Figure 6.3.

As expected, both the full and the sparse representations provide accurate estimates of the

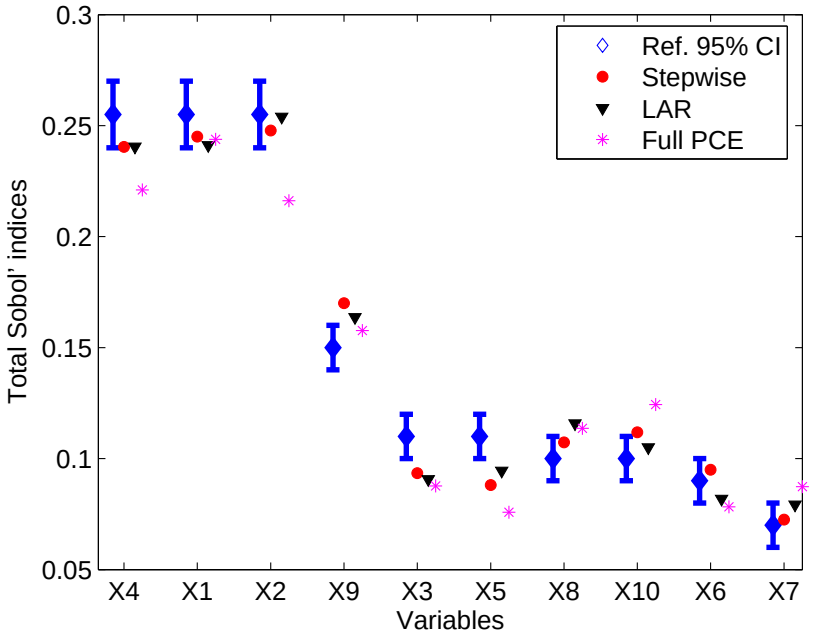


Figure 6.2: *Example #1: Morris function - Estimates of significant total Sobol' indices when setting the target accuracy ε_{tgt} equal to* 0.1 *and the degree p of the full PC expansion to 2*

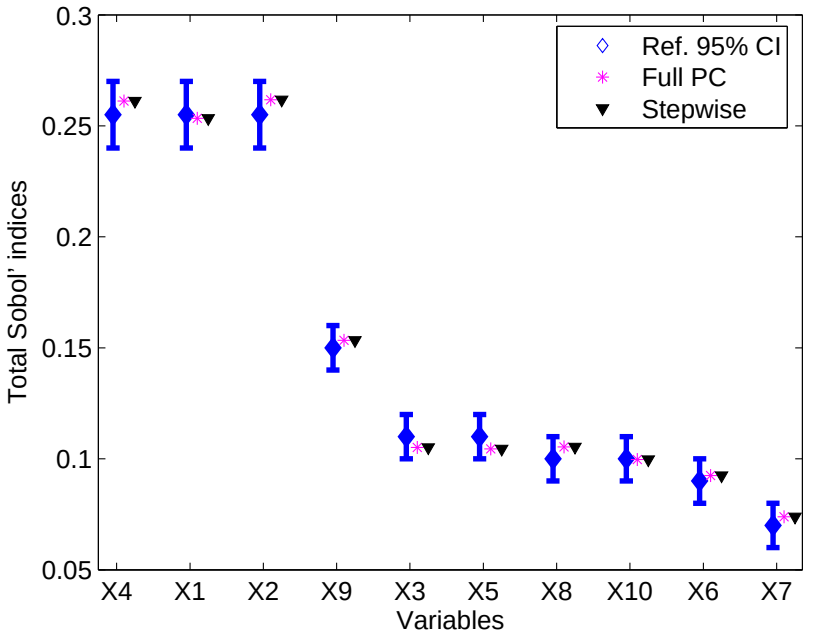


Figure 6.3: *Example #1: Morris function - Estimates of significant total Sobol' indices when setting the target accuracy ε_{tgt} equal to* 0.05 *and the degree p of the full PC expansion to 3*

Table 6.2: *Example #1: Morris function - Estimates of significant total Sobol' indices when setting the target accuracy ε_{tgt} equal to* 0.05 *and the degree p of the full PC expansion to 3*

Variables	Total Sobol' indices		
	Reference 95%-CI	Full PCE - $p = 3$	LAR
X_4	[0.244 , 0.270]	0.255	0.261
X_1	[0.242 , 0.267]	0.253	0.235
X_2	[0.240 , 0.268]	0.253	0.262
X_9	[0.140 , 0.161]	0.158	0.153
X_3	[0.102 , 0.119]	0.097	0.105
X_5	[0.100 , 0.117]	0.105	0.104
X_8	[0.091 , 0.109]	0.105	0.105
X_{10}	[0.090 , 0.107]	0.105	0.100
X_6	[0.084 , 0.100]	0.096	0.092
X_7	[0.063 , 0.077]	0.070	0.074
Number of runs	$440,000$	3,542	1,100
Relative $\mathcal{L}^2$-error		0.054	0.035
PC degree		3	11
Number of PC terms		1,771	203
Index of sparsity IS_1		1	5×10^{-5}
Index of sparsity IS_2		-	8×10^{-2}

total Sobol' indices which lie in the 95%-confidence intervals. LAR noticeably outperforms the classical "full" approach since it provides similar estimates at a computational cost divided by more than 3. In addition, the $\mathcal{L}^2$-error of the sparse metamodel is also smaller.

This examples shows how the PC expansions may be used in order to conduct the sensitivity analysis of a mathematical model featuring a large number of input parameters (say 20) at a low computational cost. The LAR approach revealed particularly efficient, allowing a correct approximation using only 450 model evaluations, which is three orders of magnitude smaller than the cost associated with crude Monte Carlo simulation. Moreover, it has been shown that very accurate estimates of the sensitivity indices could be obtained by LAR when decreasing the target error.

2.3 Example #2: Maximum deflection of a truss structure

2.3.1 Problem statement

Let us consider the truss structure sketched in Figure 6.4. The structure comprises 23 members, namely 11 horizontal bars and 12 oblical bars. The upper portion of the truss is subjected to vertical loads. A finite element model made of 23 bar elements is used. Ten parameters are assumed to be random and are modelled by independent input random variables, namely the Young's moduli and the cross-section areas of the horizontal and the oblical bars (respectively denoted by E_1, A_1 and E_2, A_2) and the applied loads (denoted by P_i, $i = 1, \ldots, 6$) (Lee and Kwak, 2006; Blatman et al., 2007; Blatman and Sudret, 2008d, 2009d), whose mean and standard deviation are reported in Table 6.3. Thus the input random vector is defined by:

$$\boldsymbol{Z} = \{E_1, E_2, A_1, A_2, P_1, ..., P_6\}^{\mathsf{T}} \tag{6.10}$$

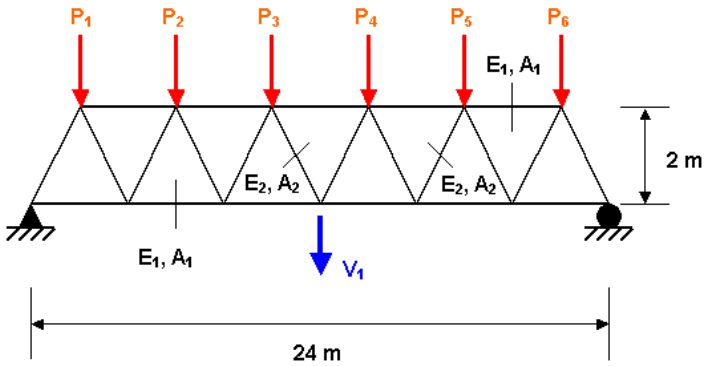


Figure 6.4: *Example #2: Truss structure comprising 23 members*

Table 6.3: *Example #2: Truss example - Input random variables*

Variable	Distribution	Mean	Standard Deviation
E_1, E_2 (Pa)	Lognormal	2.10×10^{11}	2.10×10^{10}
A_1 (m^2)	Lognormal	2.0×10^{-3}	2.0×10^{-4}
A_2 (m^2)	Lognormal	1.0×10^{-3}	1.0×10^{-4}
P_1-P_6 (N)	Gumbel	5.0×10^{4}	7.5×10^{3}

The model random response Y is the deflection at midspan denoted by V_1. It is approximated by a truncated PC expansion made of normalized Hermite polynomials. In this respect, the random vector $\boldsymbol{Z}$ is recast as a standard Gaussian random vector $\boldsymbol{X}$ by transforming the random variables Z_i as follows:

$$X_i = \Phi^{-1}\left(F_{Z_i}(Z_i)\right) \quad , i = 1, \ldots, 10 \tag{6.11}$$

.

Table 6.4: *Example #2: Truss structure - estimates of the total Sobol' indices*

Variables	Total Sobol' indices		
	Reference	Stepwise	LAR
A_1	0.388	0.374	0.374
E_1	0.367	0.378	0.373
P_3	0.075	0.073	0.073
P_4	0.079	0.069	0.079
P_5	0.035	0.036	0.034
P_2	0.031	0.037	0.037
A_2	0.014	0.013	0.013
E_2	0.010	0.014	0.012
P_6	0.005	0.005	0.004
P_1	0.004	0.005	0.005
Number of FE runs	5,500,000	60	70
Relative $\mathcal{L}^2$-error		6×10^{-3}	5×10^{-3}
PC degree		3	3
Index of sparsity 1		$76/286 \approx 27\%$	$76/286 \approx 27\%$
Index of sparsity 2		$21/76 \approx 28\%$	$32/76 \approx 42\%$

where Φ denotes the standard normal cumulative distribution function (CDF) and F_{Z_i} denotes the CDF of Z_i. This leads to the following metamodel:

$$Y \equiv V_1(\boldsymbol{X}) \simeq \sum_{\boldsymbol{\alpha} \in \mathcal{A}} a_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{6.12}$$

2.3.2 Sensitivity analysis

Of interest are the total Sobol' indices of the maximum deflection of the truss structure. Estimates are computed by post-processing PC approximations of the model response. In this respect, sparse metamodels are built up using stepwise regression and LAR with a target accuracy $\varepsilon = 0.01$. Reference results are obtained using crude Monte Carlo simulation ($5,500,000$ finite element runs are performed as a whole). In preliminary calculations, one had set the size of the initial experimental design equal to its default value 100. However both sparse metamodels could converge without adding extra points in the design. In order to check if one could reach the target accuracy with less calls to the finite element model, the initial size N_{ini} of the experimental design (resp. the number N_{add} of added points in case of overfitting) is now set equal to 50 (resp. 10). The results are reported in Table 6.4.

It can be concluded from the Sobol' indices that the variability of the deflection v_1 is much more

sensitive to the variables E_1 and A_1, than E_2 and A_2. This makes sense from a physical point of view since the properties of the horizontal bars are more influential on the displacement at midspan than the oblique ones. It can be also observed that the Sobol' indices associated with E_1 and A_1 (resp. E_2 and A_2) are similar. This is due to the fact that these variables have the same type of PDF and coefficient of variation, and that the displacement v_1 only depends on them through the products E_1A_1 and E_2A_2. Finally, the Sobol' indices reflect the symmetry of the problem, giving similar importances to the loads that are symmetrically applied (e.g. P_3 and P_4). Greater sensitivity indices are logically attributed to the forces that are close to the midspan than those located at the ends.

Such physically meaningful indices are obtained from the sparse PC expansions at a very low computational cost, say $60 - 70$ model evaluations. Hence the sparse PC approaches provide a huge computational gain factor with respect to Monte Carlo. Stepwise regression and LAR have the same efficiency in this example. Note that a full second-order PC approximation, which would be believed to provide accurate estimates of the sensitivity indices, would require about $2P$ model evaluations, where $P \equiv \binom{10+2}{2} = 66$, hence a computational cost multiplied by 2.

2.3.3 Reliability analysis

The serviceability of the structure with respect to an admissible maximal deflection v_{max} is studied. The associated limit state function reads:

$$g(\boldsymbol{x}) \;=\; v_{max} \;-\; |v_1(\boldsymbol{x})| \;\leq\; 0 \tag{6.13}$$

The reference value of the probability of failure P_f^{REF} is obtained by importance sampling using 500,000 evaluations of the model (the sampling density is a multinormal density centered on the design point resulting from a FORM (first-order reliability method) analysis). The corresponding *generalized reliability index* is given by $\beta^{REF} = -\Phi^{-1}(P_f^{REF})$.

The model response is approximated by sparse PC expansions that are built using stepwise regression and LAR using $\varepsilon_{tgt} = 0.001$. The probability of failure is then computed by Monte Carlo simulation of the metamodel (10^7 samples are used). A parametric study is carried out varying the threshold v_{max} from 10 to 14 cm. For the sake of comparison, the probabilities of failure are also estimated by FORM. The results are reported in Table 6.5 in terms of generalized reliability indices.

It appears that both sparse PC metamodels yield accurate estimates of β, say with a relative error not greater than 6% to the reference values. Only 200 runs to the finite element model were necessary to obtain these estimates. Note that stepwise regression and LAR provide similar results, even if the stepwise-based PC approximation is sparser than the LAR-based one.

.

Table 6.5: *Example #2: Truss structure - Estimates of the generalized reliability index* $\widehat{\beta} = -\Phi^{-1}(P_f)$ *and relative error* ε *for various values of the threshold*

Threshold	Reference	Stepwise		LAR		FORM	
(cm)		$\widehat{\beta}$	ϵ (%)	$\widehat{\beta}$	ϵ (%)	$\widehat{\beta}$	ϵ (%)
10	1.75	1.73	1	1.73	1	1.91	9
11	2.38	2.42	2	2.42	2	2.57	8
12	2.97	3.05	3	3.05	3	3.17	7
13	3.50	3.65	4	3.65	4	3.71	6
14	3.98	4.21	6	4.21	6	4.21	6
Relative $\mathcal{L}^2$-error		5×10^{-4}		4×10^{-4}			
PC degree		3		3			
Number of PC terms		21		32			
Index of sparsity IS_1		27%		27%			
Index of sparsity IS_2		57%		80%			
Number of FE runs	5,000	200		200		121 †	

† Number of model evaluations that was used to compute a single β, *i.e.* the one associated with a threshold equal to 14 cm (10 FORM iterations were performed).

As expected, the discrepancy between the PC-based and the reference solutions increases with the threshold value, *i.e.* when the probability of failure decreases. Accordingly, the PC-based approaches outperform FORM all the more since β is low. FORM becomes competitive when the obtained reliability index β is close to 4. Note however that a *single* sparse PC expansion is determined to get the reliability indices associated with the various values of the threshold v_{max}. In contrast, FORM has to be restarted for each value of v_{max}. Such an approach based on sparse PC approximations may be particularly appealing when the evolution of β with respect to a threshold is investigated.

2.3.4 Probability density function of the maximum deflection

The probability density function (PDF) of the maximal deflection can be estimated by post-processing the various sparse PC approximations that have been obtained in the previous sections. The reference solution corresponds to 1,000,000 runs of the finite element model. The PC-based PDF corresponds to 1,000,000 samples of the PC metamodels respectively obtained by stepwise regression and LAR, with a target accuracy ε_{tgt} set equal to 0.01 and 0.001. In all cases, a kernel density of the sample set of maximal deflection is used (Figures 6.5, 6.6).

It is not easy to distinguish the various methods when using a linear scale as in Figure 6.5. However, a logarithmic scale (Figure 6.6) reveals that the PDF obtained from the sparse meta-

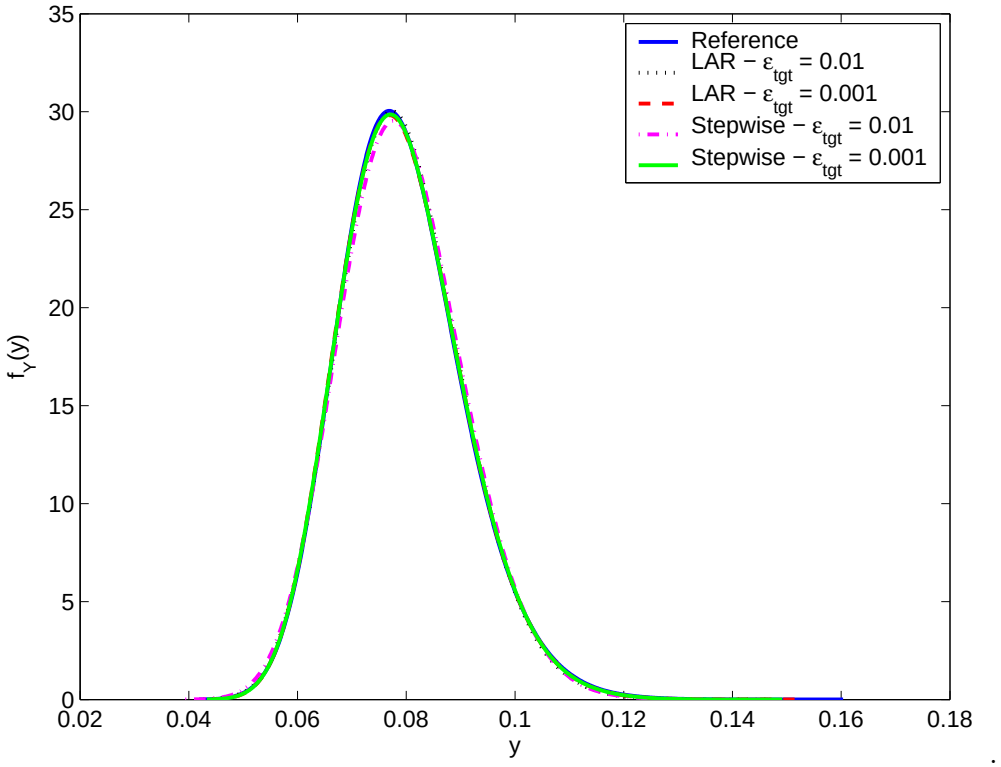


Figure 6.5: *Example #2: Truss structure - probability density function of the maximal deflection*

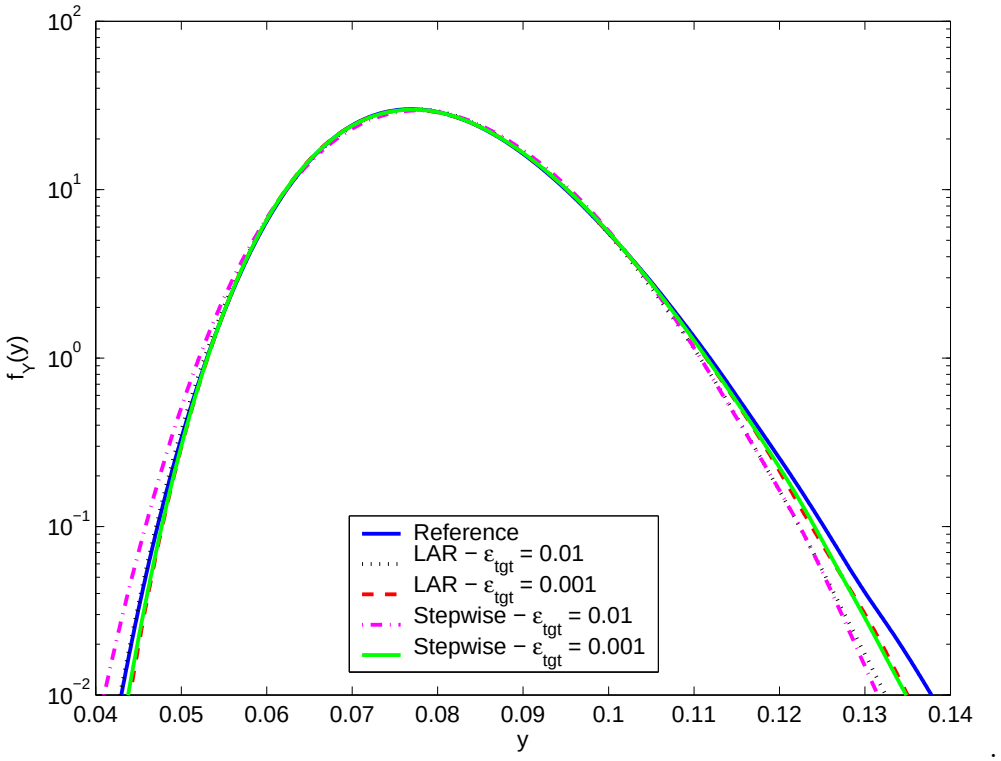


Figure 6.6: *Example #2: Truss structure - log-probability density function of the maximal deflection*

models corresponding to $\varepsilon = 0.001$ are closer to the reference solution than those derived from the sparse expansions associated with $\varepsilon = 0.01$, especially in the tails. It is observed that the various approximations deviate all the more from the reference PDF since the probability is low. This explains the decreasing accuracy of the PC-based reliability indices (Table 6.5) when decreasing the deflection threshold.

It may be retained from this example that sensitivity, distribution and reliability analysis could be carried out at a very low computational cost, say less than 200 runs of the finite element model. In particular, physically meaningful results in terms of sensitivity indices were obtained, which stresses the consistency of global sensitivity analysis in order to explain the behaviour of physical systems.

2.3.5 Convergence and complexity analysis

The convergence of full and sparse PC approximations are now compared. To this end, full PC expansions of degree p equal to 2 and 3 are built ($N = 2P$ model evaluations are used, where P is the number of PC terms). On the other hand, sparse metamodels are constructed by LAR and stepwise regression with a target accuracy ε_{tgt} set equal to 10^{-3}, 10^{-4}, 5×10^{-5} and 10^{-5}. The empirical convergence rates are depicted in Figure 6.7.

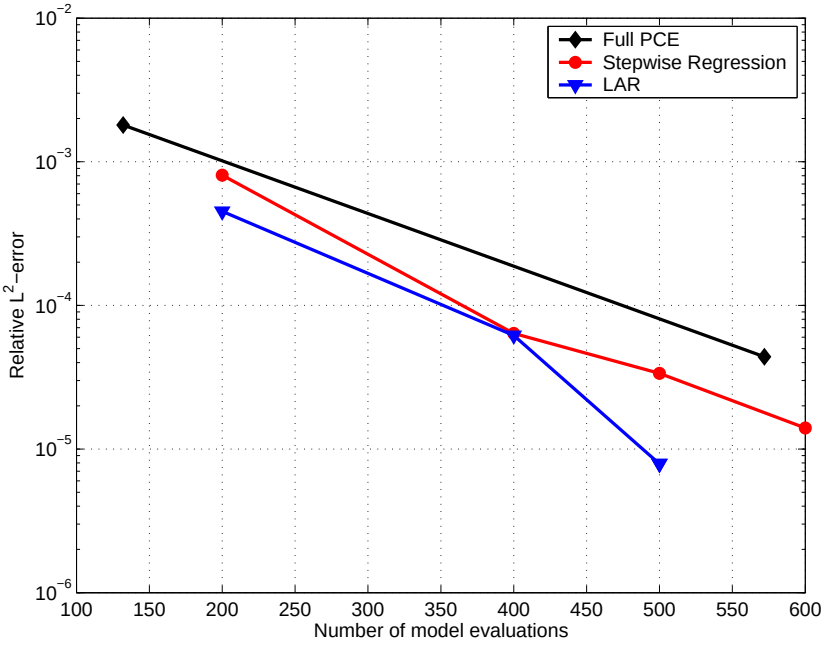


Figure 6.7: *Example #2: Truss structure - Convergence of full and sparse polynomial chaos approximations*

It is observed that the sparse PC expansions converge more rapidly than the full ones. LAR appears to be the most efficient scheme. It allows one to reach an error 10 times less (10^{-5})

than the one associated with a full third-order PC representation (10^{-4}) when using $N = 500$ model evaluations.

Moreover, the efficiency of the LAR and stepwise regression schemes are compared in terms of computer processing time. Note that this time corresponds to the complexity of the algorithms since the simple model under consideration can be evaluated at negligible time. The results are presented in Figure 6.8.

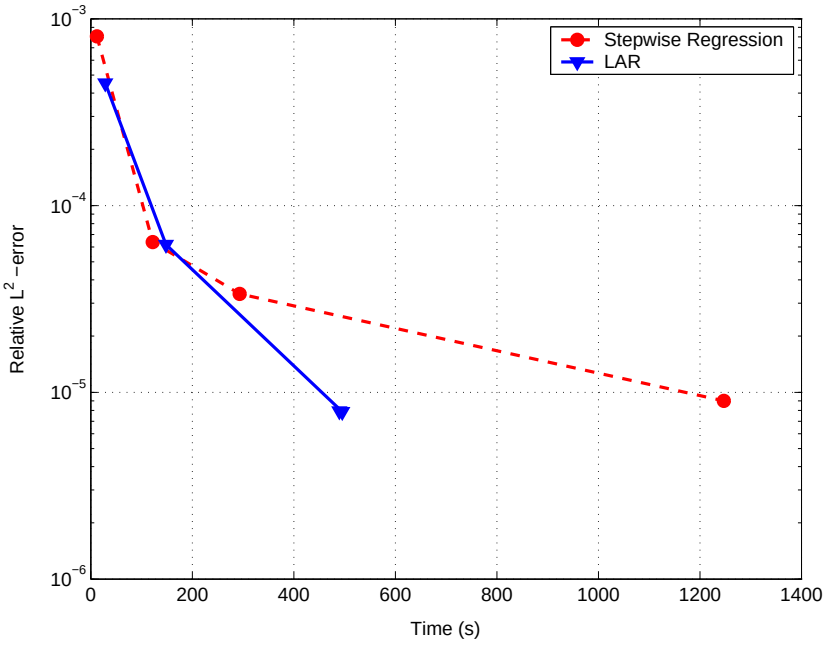


Figure 6.8: *Example #2: Truss structure - Efficiency of the LAR and stepwise regression schemes*

Both procedures behave quite similarly when the relative error is greater than 5×10^{-4}. However, LAR performs noticeably faster when higher relative error are targeted. In particular, a computational gain factor greater than 2 (*i.e.* 500 instead of $1,250$ seconds) is obtained for reaching the relative error $\varepsilon = 10^{-5}$.

2.4 Example #3: Top-floor displacement of a frame structure

2.4.1 Problem statement

Let us consider now the structure sketched in Figure 6.9, already studied in (Liu and Der Kiureghian, 1991; Wei and Rahman, 2007; Blatman and Sudret, 2009d). It is a three-span, five-

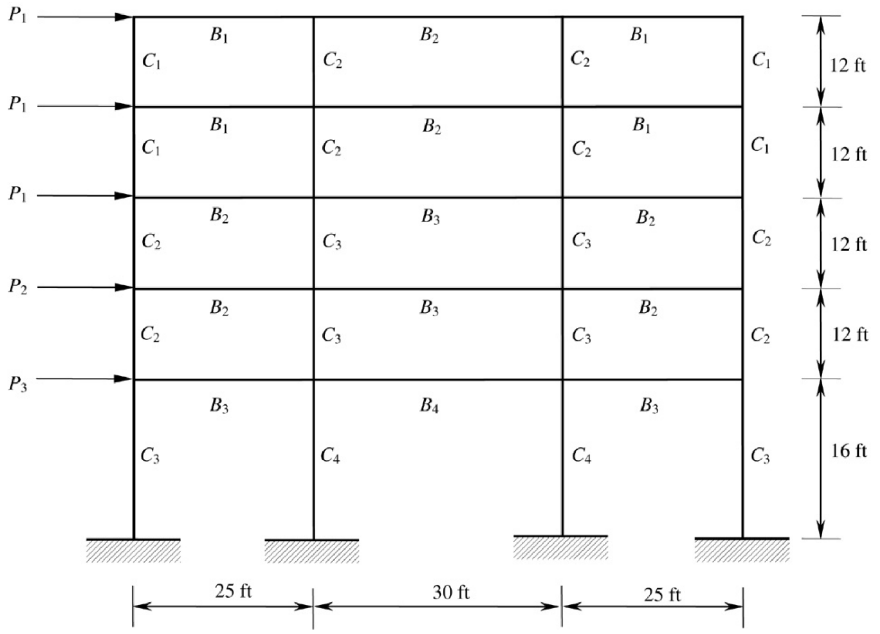


Figure 6.9: *Example of a 3-span, 5-story frame structure subjected to lateral loads*

story frame structure subjected to horizontal loads. The frame elements are made of 8 different materials, whose properties are gathered in Table 6.6.

Table 6.6: *Example #3: Frame structure - Element properties*

Element	Young's modulus	Moment of inertia	Cross-sectional area
B_1	E_4	I_{10}	A_{18}
B_2	E_4	I_{11}	A_{19}
B_3	E_4	I_{12}	A_{20}
B_4	E_4	I_{13}	A_{21}
C_1	E_5	I_6	A_{14}
C_2	E_5	I_7	A_{15}
C_3	E_5	I_8	A_{16}
C_4	E_5	I_9	A_{17}

The response of interest is the horizontal component of the top-floor displacement at the top right corner, which is denoted by u.

2.4.2 Probabilistic model

The 3 applied loads and the 2 Young's moduli, the 8 moments of inertia and the 8 cross-section areas of the frame components are modelled by random variables. They are gathered in random vector $\boldsymbol{Z} = (P_1, P_2, P_3, I_6, ..., I_{13}, A_{14}, ..., A_{21})$ of size $M = 21$.

The applied loads (resp. the material properties) are assumed to follow a lognormal distribution (resp. truncated Gaussian distribution over $[0, +\infty)$). The mean and the standard deviation of the random variables are reported in Table 6.7. Note that truncated Gaussian distributions are used in contrast to the original example in Liu and Der Kiureghian (1991). Indeed, it is not possible to obtain a reference solution by Monte Carlo simulation when using Gaussian distributions due to non physical negative realizations of the geometrical and material properties.

Table 6.7: *Example #3: Frame structure - Input random variables properties*

Variable	Distribution	Mean †	Standard Deviation †
P_1 (kN)	Lognormal	133.454	40.04
P_2 (kN)	”	88.97	35.59
P_3 (kN)		71.175	28.47
E_4 (kN/m^2)	Truncated Gaussian over $[0, +\infty)$	2.1738×10^{7}	1.9152×10^{6}
E_5 (kN/m^2)	”	2.3796×10^{7}	1.9152×10^{6}
I_6 (m^4)		8.1344×10^{-3}	1.0834×10^{-3}
I_7 (m^4)		1.1509×10^{-2}	1.2980×10^{-3}
I_8 (m^4)		2.1375×10^{-2}	2.5961×10^{-3}
I_9 (m^4)		2.5961×10^{-2}	3.0288×10^{-3}
I_{10}(m^4)	”	1.0812×10^{-2}	2.5961×10^{-3}
I_{11}(m^4)		1.4105×10^{-2}	3.4615×10^{-3}
I_{12}(m^4)		2.3279×10^{-2}	5.6249×10^{-3}
I_{13}(m^4)		2.5961×10^{-2}	6.4902×10^{-3}
A_{14}(m^2)		3.1256×10^{-1}	5.5815×10^{-2}
A_{15}(m^2)		3.7210×10^{-1}	7.4420×10^{-2}
A_{16}(m^2)		5.0606×10^{-1}	9.3025×10^{-2}
A_{17}(m^2)		5.5815×10^{-1}	1.1163×10^{-1}
A_{18}(m^2)	”	2.5302×10^{-1}	9.3025×10^{-2}
A_{19}(m^2)		2.9117×10^{-1}	1.0232×10^{-1}
A_{20}(m^2)		3.7303×10^{-1}	1.2093×10^{-1}
A_{21}(m^2)		4.1860×10^{-1}	1.9537×10^{-1}

† The mean value and standard deviation of the cross sections, moments of inertia and Young's moduli are those of the untruncated Gaussian distributions

Moreover the various input random variables are correlated using a Nataf distribution. The correlation matrix is defined as follows:

- the correlation coefficient of the $\widehat{Z}_i$'s (Gaussian variables obtained by transforming the marginal distribution of the Z_i's, see Chapter 3, Section 2.3) associated with the cross section areas and the moments of inertia of a given member is set equal to $\rho_{A_i,I_i} = 0.95$;

- otherwise the correlation coefficients of the geometrical properties are set equal to $\rho_{A_i,I_j} = \rho_{I_i,I_j} = \rho_{A_i,A_j} = 0.13$;
- the correlation coefficient of the two Young's moduli is set equal to $\rho_{E_4,E_5} = 0.9$;
- the remaining correlation coefficients in $\boldsymbol{R}$ are zero.

Note that these values are *not* the correlation coefficients of the input variables in $\boldsymbol{Z}$, the latter being insignificantly different though.

The model response is recast as a function of *independent* standard Gaussian random variables ξ_i so that it may be expanded onto a PC expansion made of normalized Hermite polynomials.

2.4.3 Sensitivity analysis

Of interest are the total sensitivity indices of the random displacement at the top right corner U. Reference values are obtained by direct Monte Carlo simulation of the problem ($460,000$ samples are used), and 95%-confidence intervals are derived by bootstrap ($1,000$ replicates are used). On the other hand, the sensitivity indices are estimated by postprocessing sparse PC expansions based on stepwise regression and LAR. The estimates of the *significant* total sensitivity indices (*i.e.* those indices for which the upper bound of the confidence interval is greater than 0.01) are reported in Table 6.8 and depicted in Figure 6.10.

It appears that the response variance is mainly explained by the random variable ξ_1, which only depends on the loading P_1. The sparse PC expansions provide accurate estimates of the Sobol' indices, *i.e.* which lie in the 95%-confidence intervals, using no more than 350 model evaluations. This computational cost is quite low insofar as a full second-order PC approximation would require already about $2P = 2\binom{21+2}{2} = 506$ finite element runs. LAR overperforms stepwise regression in this example since it yields a more accurate PC approximation at a lower computational cost.

2.4.4 Reliability analysis

Let us study the serviceability of the frame structure with respect to the limit state function:

$$g(\boldsymbol{X}) = u_{max} - \mathcal{M}(\boldsymbol{X}) \tag{6.14}$$

where u_{max} is a given threshold. It is approximated using a PC expansion as follows:

$$g_{PC}(\boldsymbol{X}) = u_{max} - \mathcal{M}_{PC}(\boldsymbol{X}) \tag{6.15}$$

Table 6.8: *Example #3: Frame structure - Estimates of the significant total Sobol' indices*

Variables	Total Sobol' indices		
	Reference 95%-CI[†]	Stepwise	LAR
ξ_1	[0.737 , 0.863]	0.766	0.765
ξ_2	[-0.001 , 0.015]	0.014	0.012
ξ_3	[0.055 , 0.097]	0.078	0.080
ξ_4	[0.011 , 0.029]	0.010	0.011
ξ_5	[0.002 , 0.021]	0.014	0.014
ξ_6	[0.000 , 0.032]	0.017	0.018
ξ_7	[0.000 , 0.014]	0.007	0.006
ξ_8	[0.000 , 0.021]	0.012	0.012
ξ_9	[0.046 , 0.086]	0.052	0.052
ξ_{10}	[0.018 , 0.055]	0.037	0.040
Number of FE runs	$460,000$	350	250
Relative $\mathcal{L}^2$-error		10^{-2}	9×10^{-3}
PC degree		6	6
Number of PC terms		57	42
Index of sparsity IS_1		17%	17%
Index of sparsity IS_2		17%	12%

A parametric study is carried out varying the threshold u_{max} from 4 to 9 cm. Reference values of the probabilities of failure P_f are obtained by FORM followed by importance sampling ($500,000$ model evaluations are used to get a coefficient of variation less than 1.0% on P_f).

Estimates of the reliability index are computed by post-processing sparse PC approximations. The latter are built up with a target approximation error ε_{tgt} set equal to 10^{-3}. The estimates of the various generalized reliability indices $\beta = -\Phi^{-1}(P_f)$ are reported in Table 6.9.

Stepwise regression and LAR both provide accurate estimates of the generalized sensitivity indices, with relative errors less than 5% with respect to the reference values for values of β ranging from 2 to 3.5. As observed in the truss example (Section 2.3), the estimation error increases with the threshold value. Stepwise regression and LAR perform quite similarly in this example, with a slight advantage for stepwise regression. Empirical results in Berveiller (2005) have shown that a full third-order PC expansion provides accurate estimates of β. However such an expansion would require about $2\binom{21+3}{3} = 4,048$ calls to the finite element model, hence a computational cost multiplied by 4 compared to the approaches based on sparse metamodels.

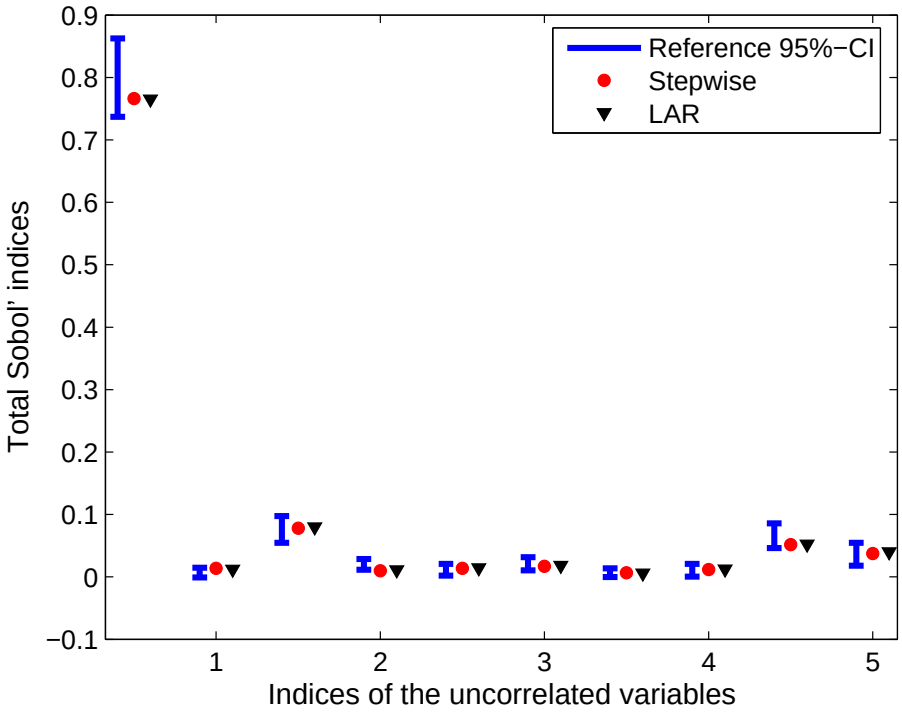


Figure 6.10: *Example #3: Frame structure - Estimates of the significant total sensitivity indices*

Table 6.9: *Example #3: Frame structure - Estimates of the generalized reliability index* $\widehat{\beta} = -\Phi^{-1}(P_f)$ *and relative error* ϵ *for various values of the threshold*

Threshold (cm)	Reference	Stepwise		LAR	
		$\widehat{\beta}$	ϵ (%)	$\widehat{\beta}$	ϵ (%)
4	2.27	2.28	1	2.30	1
5	2.96	3.01	2	3.04	3
6	3.51	3.61	3	3.65	4
7	3.96	4.12	4	4.19	6
Relative $\mathcal{L}^2$-error		10^{-3}		1.2×10^{-3}	
PC degree		7		6	
Number of PC terms		249		166	
Index of sparsity IS_1		3×10^{-4}		0.1%	
Index of sparsity IS_2		69%		49%	
Number of FE runs		1,000		900	

2.5 Example #4: Settlement of a foundation

2.5.1 Problem statement

Let us study the problem of the settlement of a foundation on an elastic soil layer showing spatial variability in its material properties, already addressed in Sudret and Der Kiureghian (2000). A structure to be founded on this soil mass is idealized as a uniform pressure P applied over a length $2B$ of the free surface (see Figure 6.11). The soil is modelled as an elastic linear isotropic material. A plane strain analysis is carried out.

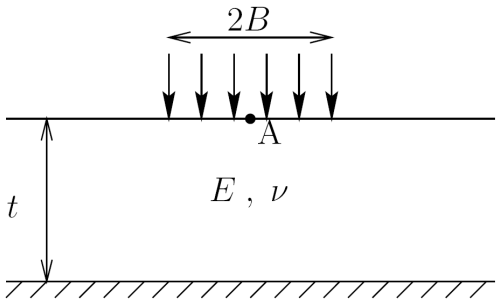
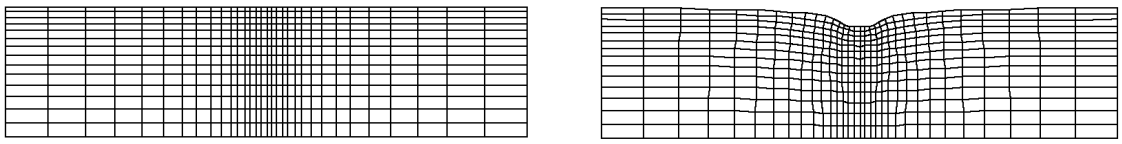


Figure 6.11: *Example #4: Settlement of a foundation - problem definition*

The finite element model displayed in Figure 6.12-a was chosen. The foundation width is equal to 20 m and the soil mesh width is equal to 120 m. The soil layer thickness is equal to 30 m and its Poisson's ratio to 0.3. The finite element mesh is made of 448 Q4-elements and 495 nodes.



a - Finite element mesh *b - Deformed mesh*

Figure 6.12: *Example #4: Settlement of a foundation - finite element mesh of the soil layer*

2.5.2 Probabilistic model

The Young's modulus of the soil is considered to vary both in the vertical and the horizontal directions. It is modelled by a two-dimensional homogeneous lognormal random field. Its mean value is set equal to $\mu_E = 50\,\mathrm{MPa}$ and its coefficient of variation is $\delta_E = \sigma_E/\mu_E = 0.3$. The autocorrelation coefficient function of the underlying Gaussian field $N(\boldsymbol{x}, \omega)$ is:

$$\rho_N(\boldsymbol{x}, \boldsymbol{x}') = \exp\left[-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{\ell^2}\right] \tag{6.16}$$

where $\ell = 15$ m. The Gaussian field $N(\boldsymbol{x}, \omega)$ is discretized using the Karhunen-Loève (KL) decomposition. As no analytical solution is available for autocorrelation functions such as in Eq.(6.16), the latter is expanded onto an orthogonal polynomial basis, which allows to compute a numerical KL decomposition (see Appendix B for more details). The problem is then recast as a function of M independent standard Gaussian random variables $\boldsymbol{\xi} = \{\xi_1, \ldots, \xi_M\}^\mathsf{T}$. A relative variance error less than 1% is obtained by selecting $M = 38$ random variables. The discretization error over the structure is illustrated by Figure 6.13.

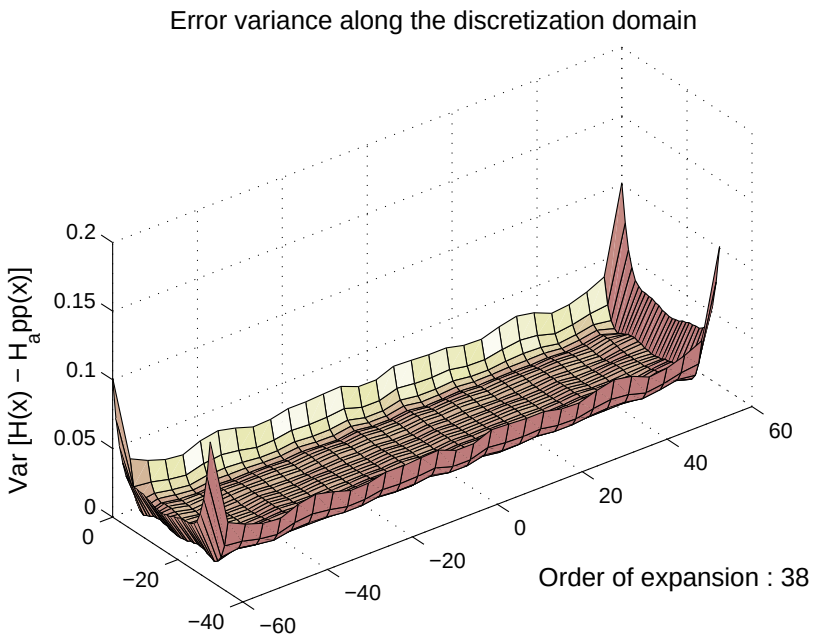


Figure 6.13: *Example #4: Settlement of a foundation - Discretization error over the structure of the Young's modulus random field using a* 38*-term Karhunen-Loève expansion*

2.5.3 Average vertical settlement under the foundation

The average vertical displacement $\bar{u}$ under the foundation is of interest. It may be regarded as a random variable denoted by $Y = \mathcal{M}(\boldsymbol{\xi})$ (where $\boldsymbol{\xi} \equiv \{\xi_1, \ldots, \xi_M\}$ is a set of independent standard normal variables) due to the probabilistic assumptions presented in the previous section.

The sensitivity of the average vertical displacement to each eigenmode ξ_i in the Karhunen-Loève expansion of the Young's modulus is investigated. In this purpose, the total Sobol' indices of the response are derived by post-processing sparse PC approximations obtained by stepwise regression and LAR.

It appears that both methods provide total Sobol' indices whose sum is approximately equal to 1. This means that the interaction effects between the eigenmodes are negligible and allows one

to treat the total indices as the first-order ones, *i.e.* as ratios between partial variances and the total variance. The sensitivity indices associated with the 15 first eigenmodes are represented in Figure 6.14.

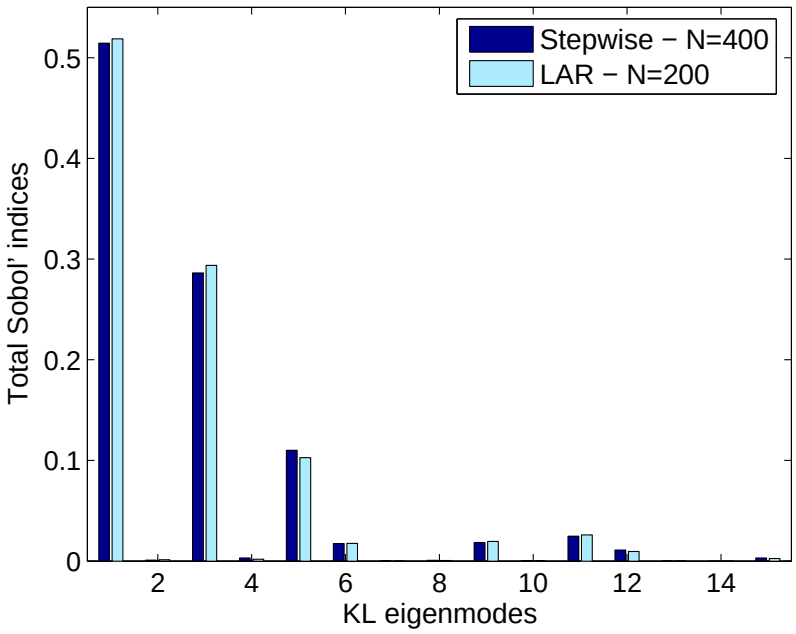


Figure 6.14: *Example #4: Settlement of a foundation - Total Sobol' indices of the 15 first eigenmodes of the Karhunen-Loève expansion (which explain more than 99% of the total response variance)*

It is observed that stepwise regression and LAR yield very similar results. However LAR is more efficient since it makes use of twice as less runs of the finite element model, say only $N = 200$ model evaluations. In comparison, using a full second-order PC expansion would require performing more than $\binom{M+p}{p} = \binom{38+2}{2} = 780$ model evaluations.

A very fast decay of the importance of the eigenmodes is noticed, with the 5 first eigenmodes explaining more than 90% of the response total variance. This was expected since the model response of interest is an averaged quantity over the domain of application of the load, which is therefore quite insensitive to small-scale fluctuations of the spatially variable random Young's modulus.

In addition, it appears that only the sensitivity indices corresponding to those eigenfunctions that are symmetric with respect to the vertical axis are non zero, reflecting the genuine symmetry of the problem under consideration. This is made clear by associating the results in Figure 6.14 with the plot of the 9 first eigenmodes in Figure 6.15. Accordingly, many coefficients should vanish in the true PC expansion of the model response, hence a very sparse structure. The antisymmetric modes #2, 4, 7 and 8 have a zero sensitivity index as expected. This shows the small *effective dimension* of the response in spite of a large nominal dimension, say $M = 38$.

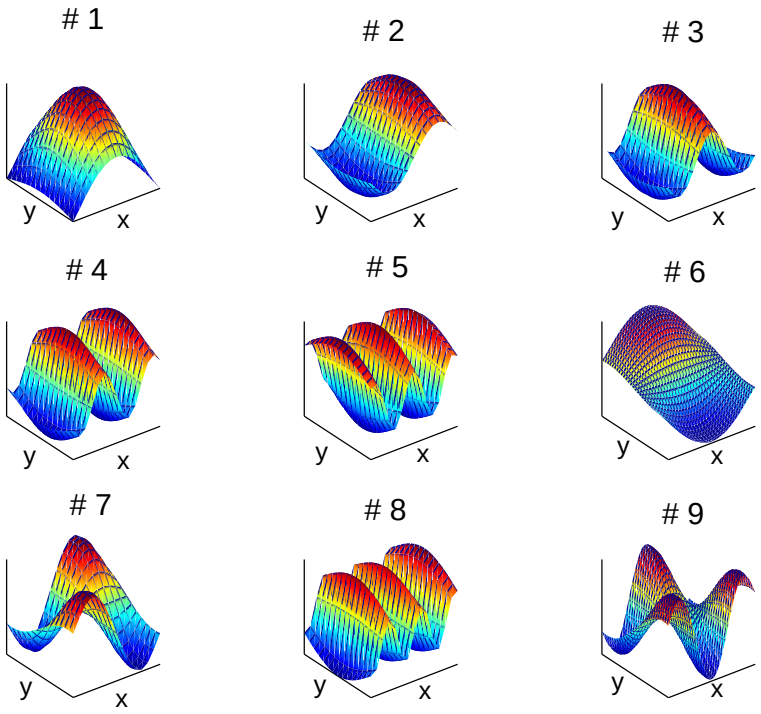


Figure 6.15: *Example #4: Settlement of a foundation - Representation of the 9 first eigenmodes of the Karhunen-Loève expansion of the Young's modulus random field*

2.5.4 Vertical displacement field over the soil layer

The mean and standard deviation of the vertical displacement at *each node* of the soil mesh are now of interest. The vertical displacement field is then regarded as the response of interest and is represented by a random vector denoted by $\boldsymbol{Y} \equiv \mathcal{M}(\boldsymbol{X})$. Each component of $\boldsymbol{Y}$ (*i.e.* each random nodal vertical displacement) is approximated by a sparse PC expansion that is built up using the LAR procedure, which revealed particularly efficient in the previous section. In this purpose, one uses the procedure for applying Stepwise or LAR to a vector-valued response (Chapter 4, Section 4.5). The target accuracy ε_{tgt} is set equal to 0.05 in order to ensure a moderate number of calls to the deterministic finite element model $\mathcal{M}$. The estimates of the mean and the standard deviation of the displacement field (resp. the relative $\mathcal{L}^2$-approximation error) are plotted in Figure 6.16 (resp. Figure 6.17).

As expected, the fields of second moments reflect the genuine symmetry of the problem (Figure 6.16). Such physically meaningful results are obtained by LAR using only 400 evaluations of the finite element model. As a consequence, it will be possible to study only one half of the structure in further investigations, as done empirically in Sudret and Der Kiureghian (2000); Sudret (2007).

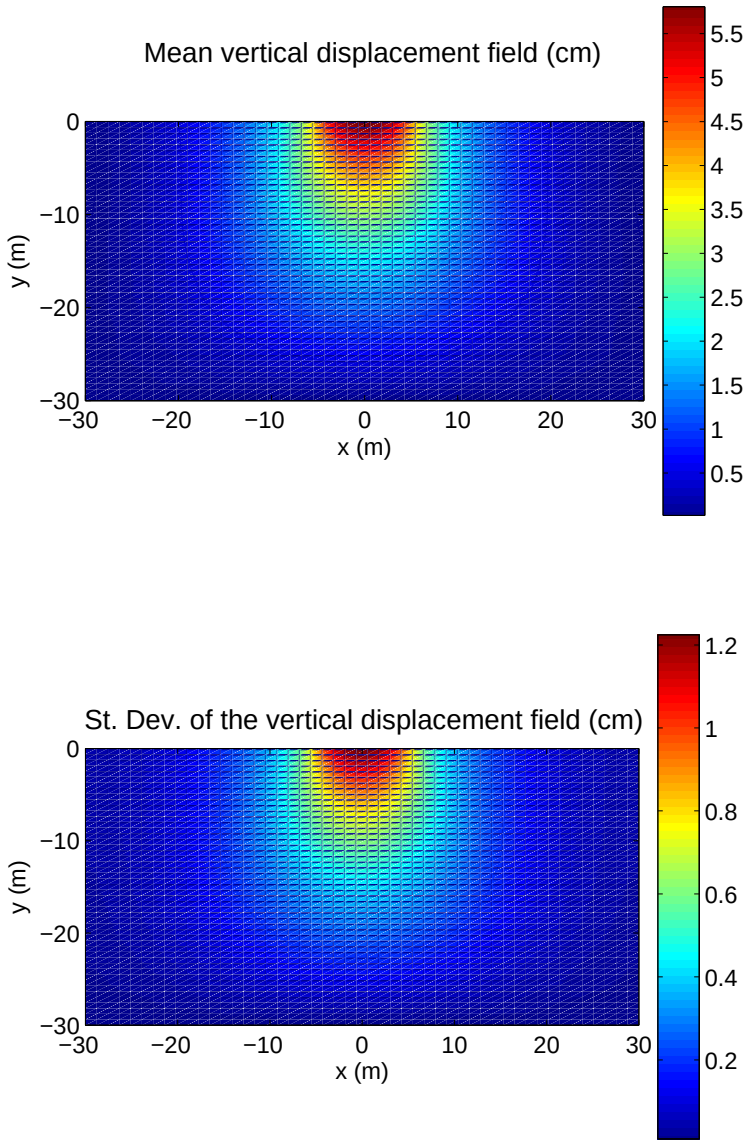


Figure 6.16: *Example #4: Settlement of a foundation - Estimated second moments of the random field of vertical displacements*

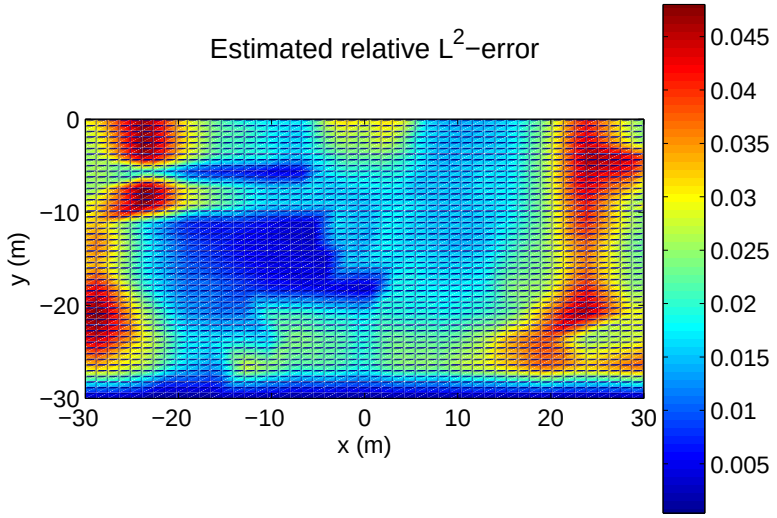


Figure 6.17: *Example #4: Settlement of a foundation - Estimated $\mathcal{L}^2$-error of approximation of the random field of vertical displacements*

Figure 6.17 shows the repartition of the estimated $\mathcal{L}^2$-error (*i.e.* the values of the corrected leave-one-out error estimates ε^*_{LOO}, see Chapter 4, Section 3 for more details) over the soil structure, which corresponds to the prescribed accuracy. A nice feature of the PC expansions is that they can also provide information about the correlation structure of the random displacements field (*e.g.* the two-point correlation coefficients) by means of elementary algebraic operations on the PC coefficients (see Chapter 4, Section 4.5). The correlation coefficients between the greatest vertical displacement (located at the origin ($x = y = 0$) in Figure 6.16) and the other nodal vertical displacements are represented in Figure 6.18

As expected, the correlation decreases as the distance from the maximum displacement increases. The correlation coefficients are equal to $0.8 - 1$ (resp. $0.5 - 0.8$) inside a semi-disk of radius 10 m (resp. 20 m) around the maximum displacement. They become insignificant (say less than 0.2 in absolute value) when the distance along the x-axis (resp. y-axis) is equal to $20 - 25$ m (resp. 28 m).

2.6 Example #5 - Bending of a simply supported beam

2.6.1 Problem statement

We now investigate a problem of very low effective dimension in spite of a particularly large number of random input variables. In this purpose, let us consider the elastic beam bending problem depicted in Figure 6.19. The beam is simply supported and subjected to an uniformly

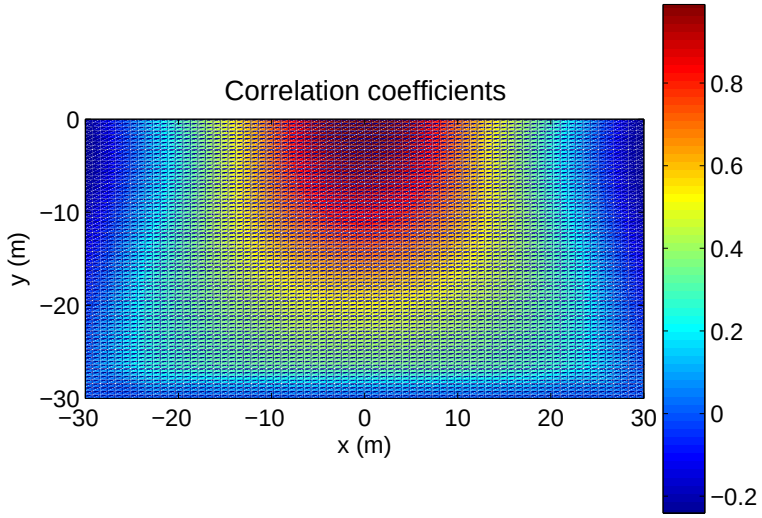


Figure 6.18: *Example #4: Settlement of a foundation - Estimated correlation coefficients between the greatest vertical displacement (located at the origin* $(x = y = 0)$*) and the other nodal vertical displacements*

distributed load. In this study, the beam length is set equal to 3 m and its moment of inertia to $0.8 \times 10^{-5}\,\mathrm{m}^4$. The applied pressure is equal to 13 kN. The finite element mesh is made of 100 beam elements. The maximum deflection U_{max} of the structure is the output of interest.

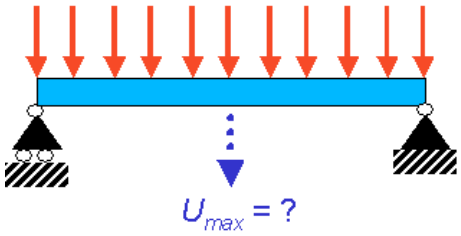


Figure 6.19: *Example #5 - Problem of an elastic beam bending*

The Young's modulus of the beam is modelled by an homogeneous lognormal random field. Its mean value is set equal to $\mu_E = 210\,\mathrm{MPa}$ and its coefficient of variation is $\delta_E = \sigma_E/\mu_E = 20\%$. The autocorrelation coefficient function of the underlying Gaussian field $N(\boldsymbol{x}, \omega)$ is:

$$\rho_N(\boldsymbol{x}, \boldsymbol{x}') = \exp\left[-\frac{|x - x'|}{\ell}\right] \tag{6.17}$$

where $\ell = 0.5$ m. The Gaussian field $N(\boldsymbol{x}, \omega)$ is discretized using the Karhunen-Loève decomposition, which allows one to recast the problem as a function of M independent standard Gaussian random variables $\boldsymbol{\xi} = \{\xi_1, \ldots, \xi_M\}^\mathsf{T}$ (see Chapter 3, Section 2.4). A relative variance error of about 1% is obtained by selecting $M = 100$ random variables. The discretization error along the beam length is illustrated in Figure 6.20. The maximum deflection may thus be regarded

as a random variable denoted by $Y = \mathcal{M}(\boldsymbol{\xi})$.

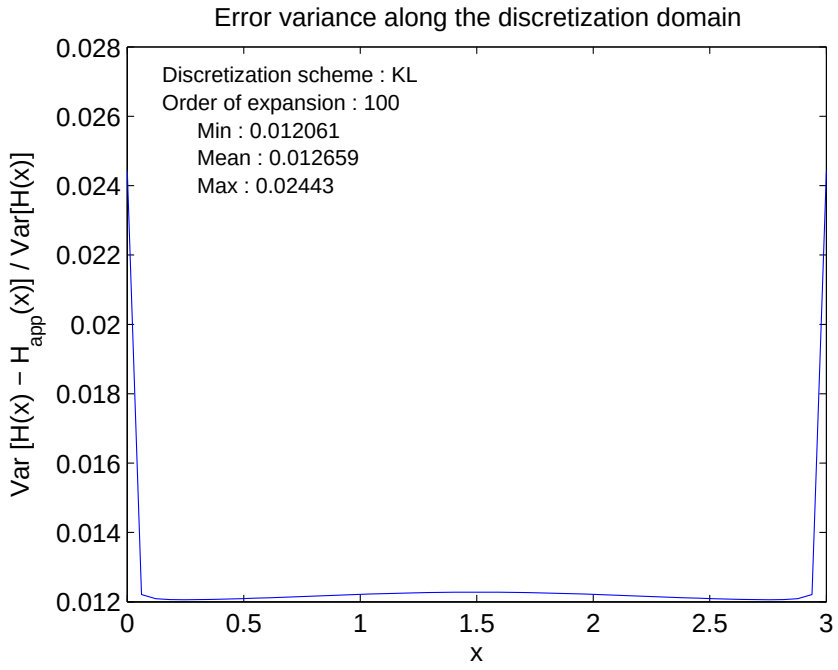


Figure 6.20: *Example #5: Beam bending - Discretization error along the beam length of the Young's modulus random field using a 100-term Karhunen-Loève expansion*

2.6.2 Statistical moments of the maximum deflection

The statistical moments of the response are now considered. Reference results are obtained using crude Monte Carlo simulation of the problem with $10,000$ samples. In addition, 95%-confidence intervals of the skewness and kurtosis coefficients have been computed by bootstrap using $1,000$ replicates. On the other hand, two sparse PC expansions based on LAR are post-processed, namely the one used for sensitivity analysis in the previous section ($\varepsilon_{tgt} = 0.01$) and a more accurate approximation with ε_{tgt} set equal to 0.001. The results are gathered in Table 6.10.

The sparse PC approximation associated with $\varepsilon_{tgt} = 0.01$ provides accurate estimates of the second moments of the response at a low computational cost. However the estimates of the third and fourth moments are outside the 95%-confidence intervals. In contrast, the metamodel associated with $\varepsilon_{tgt} = 0.001$ yields estimates which lie inside these intervals, at the computational cost of $1,200$ model evaluations. In comparison, a full second-order expansion would contain $P\binom{100+2}{2} = 5,151$ terms and would thus require about $2P = 10,302$ finite element runs in order to get accurate results, hence a computational cost multiplied by 10 compared to the LAR approach. Note that a simple full first-order expansion could not provide satisfactory estimates of the skewness and kurtosis coefficients γ_Y and κ_Y, since it corresponds to a Gaussian approximation which would yield the estimates $\widehat{\gamma}_Y = 0$ and $\widehat{\kappa}_Y = 3$.

Table 6.10: Example #5: Beam bending - Estimation of the four first statistical moments

	Reference	LAR - $\varepsilon_{tgt} = 0.01$	LAR - $\varepsilon_{tgt} = 0.001$
Mean (mm)	2.83	2.81	2.83
Standard Deviation (mm)	0.37	0.36	0.36
Skewness	[0.38 ; 0.52] †	0.30	0.41
Kurtosis	[3.12 ; 3.70] †	3.09	3.30
Number of FE runs	10,000	200	1,200
Error estimate		10^{-3}	4×10^{-4}
PC degree		3	6
Number of PC terms		10	246
Index of sparsity IS_1		2×10^{-3}	3×10^{-6}
Index of sparsity IS_2		7%	4%

† 95%-confidence intervals obtained by bootstrap

2.6.3 Sensitivity analysis of the maximum deflection

The sensitivity of the maximal deflection to each eigenmode ξ_i in the Karhunen-Loève expansion of the Young's modulus is investigated. In this purpose, the total Sobol' indices of the response are derived by post-processing a sparse PC approximation obtained by LAR. Indeed, this method revealed efficient in the previous example which also involved a Karhunen-Loève (KL) expansion of a Young's modulus random field. The target accuracy ε_{tgt} is set equal to 0.01.

The sparse PC approximation could be determined using only 200 finite element runs, which is especially low since a full second-order PC representation contains $5,151$ terms. The estimates of the total Sobol' indices of the 20 first KL eigenmodes are depicted in Figure 6.21.

As expected, a dramatically fast decay of the sensitivity indices is observed. In addition, exactly as for the previous example, only the symmetric modes are non zero. For the sake of clarity, the four first KL eigenmodes are plotted in Figure 6.22. Thus it appears that more that 99.7% of the variance of the maximum deflection is explained by the eigenmodes #1 and #3.

Therefore, one considers now a crude KL expansion of the Young's modulus random field that contains only 3 terms. The corresponding average variance error of discretization is quite large, say about 25%. Thus the maximal deflection U_{max} is approximated by a LAR-based sparse PC expansion featuring 2 standard normal random variables. The target accuracy is $\varepsilon_{tgt} = 0.001$. As the required number of model evaluations is expected to be small, one uses a sequential experimental design of initial size $N_{ini} = 10$, and the number of additional points when detecting overfitting is $N_{add} = 10$.

The adaptive LAR procedure converges at the cost of only $N = 40$ runs of the model. The

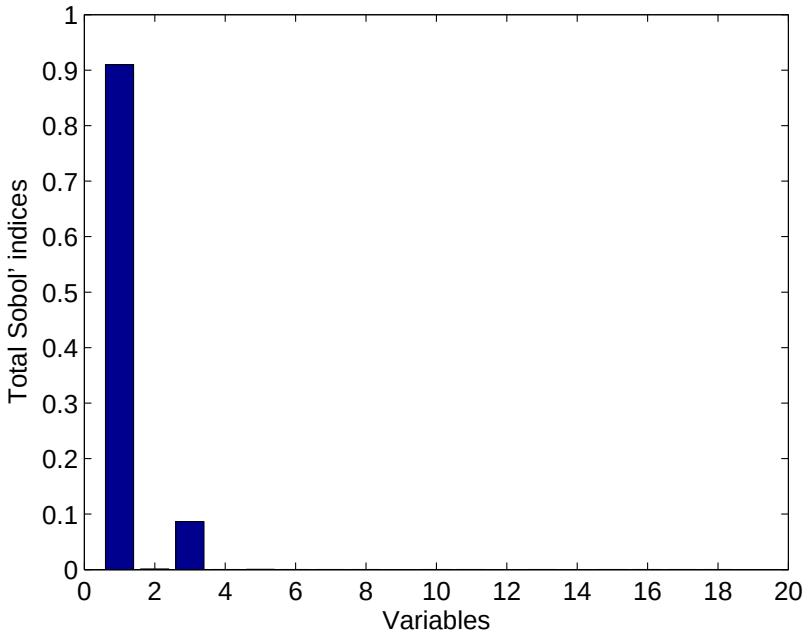


Figure 6.21: *Example #5: Beam bending - Estimates of the total Sobol' indices of the 20 first eigenmodes of the Young's modulus Karhunen-Loève decomposition*

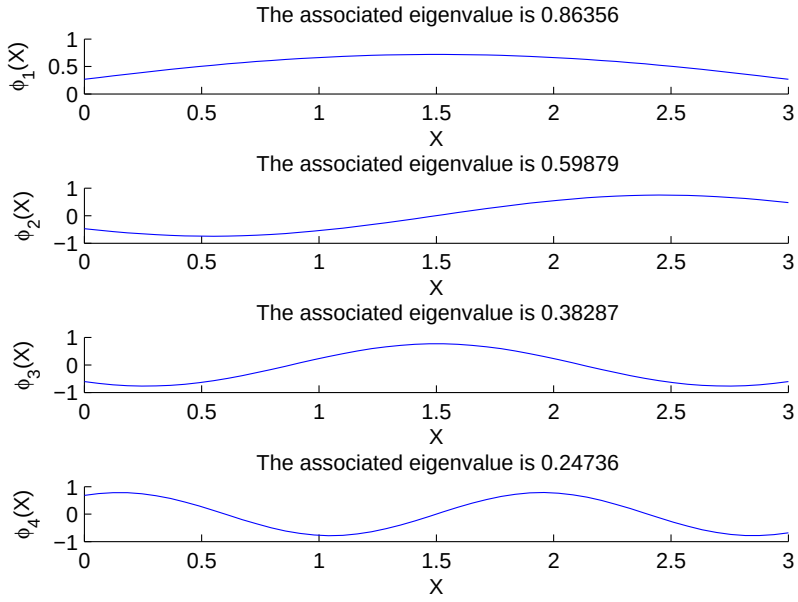


Figure 6.22: *Example #5: Beam bending - Four first eigenmodes of the Karhunen-Loève decomposition of the spatially variable Young's modulus*

obtained sparse PC approximation contains $P = 18$ terms, and its degree is $p = 5$. The estimate

of the approximation error (*corrected leave-one-out error*) is equal to 2×10^{-4}. The second moments of the maximal deflection are estimated again from this reduced model. The estimates of the mean and the standard deviation are respectively 2.81 and 0.36 mm. These results are as accurate (comparing with the reference values) as those obtained from the "fine" KL discretization ($M = 100$ terms) using $N = 200$ model evaluations, hence a computational gain factor greater than 5.

2.7 Conclusions

The five academic examples studied in this section have allowed one to validate the use of the stepwise regression and LAR approaches for propagation, sensitivity and reliability analysis. From the obtained results and other benchmark problems investigated elsewhere, the following conclusions may be drawn:

- when dealing with second moment and sensitivity analysis, sufficiently accurate results are often obtained when setting the target approximation error equal to $0.05 - 0.01$. When dealing with reliability analysis, a smaller target error may be required, say 0.001, especially when low probabilities of failure are sought.
- LAR appears to be the most efficient scheme. Moreover, it is a quite fast procedure whose complexity is equivalent to the one of an ordinary least-square regression. Hence LAR should be preferred to stepwise regression. Note that this approach has provided satisfactory results in cases for which the use of classical full PC expansions was not affordable.
- Examples #4 and #5, which involved a large number of input parameter (*i.e.* 38 and 100, respectively) show tracks to further noticeable improvement in stochastic finite element analysis. It appears that there is not always a need for an accurate discretization of the input random field when certain quantity of interest (here, the average settlement or the beam maximal deflection) is considered. Indeed, in Example #5, accurate estimates of the second moments of the beam maximal deflection have been obtained from a very crude KL expansion of input random field (only 3 eigenmodes have been retained). Such results could be obtained using only 40 model evaluations instead of 200 when using a fine KL expansion.

It will be relevant to devise a procedure for *iteratively* adding terms in the KL expansion of an input random field. Such an adaptation will be oriented by the error of approximation of the random model response, and *not* by the error of discretization of the random field as done usually. This strategy corresponds to a step-by-step building of an optimal stochastic basis for approximating the response.

3 Analysis of the integrity of the reactor pressure vessel of a nuclear powerplant

3.1 Introduction

The reactor pressure vessel (RPV) is an essential component which could potentially limit the lifetime duration of pressurized water reactor (PWR) nuclear power plants. The vessel is submitted to thermal and mechanical loadings resulting from the operation of the reactor in the normal situation, but also in an incidental or accidental situation. These situations may lead to high stress values inside the vessel wall and therefore load any pre-existing crack in the structure severely. As a result the evaluation of the risk of a crack fracture initiation in the vessel is of crucial importance. A computational tool, named *SECURE* (for *Système pour l'Évaluation des CUves REP* in French), has been developed at EDF for this purpose.

Assessing the integrity of the RPV is not an easy task since many parameters are affected by uncertainty, *e.g.* the material properties as well as the position and the geometry of the potential cracks inside the vessel wall. This makes probabilistic methods attractive in order to give insight in the study of the safety margin. It is worth mentioning that a benchmark was proposed in the context of the *PROSIR* (Probabilistic structural integrity of a PWR reactor pressure vessel) project (PROSIR, 2006). An example of reliability analysis of the RPV in the context of pressurized thermal shock may be found in Deheeger (2008).

In this work, the scenario of a *loss of coolant accident* (LOCA) is considered. The probabilistic model is set up both from (confidential) experimental data and from data already used in the PROSIR project. Uncertainty and sensitivity analysis of the safety margin of the RPV is then carried out. In this purpose, the safety margin is regarded as the output of a model featuring independent random parameters. It is approximated by a sparse PC expansion using the adaptive LAR method.

3.2 Statement of the physical problem

3.2.1 Overview

The pressure vessel is made up of shells that are welded together. These are made of low-alloy 16 MND 5 ferritic steel (*base metal*). A cladding, the function of which is to limit the effect of radiation exposure on the base metal and protect it from corrosion and oxydation, is then settled onto the internal wall of the vessel. It is made of austenitic stainless steel. In the present report the study of the vessel is restricted to the core zone (Figure 6.23) because it is there that radiation exposure is maximum. It is also assumed that radiation exposure is homogeneous in this zone.

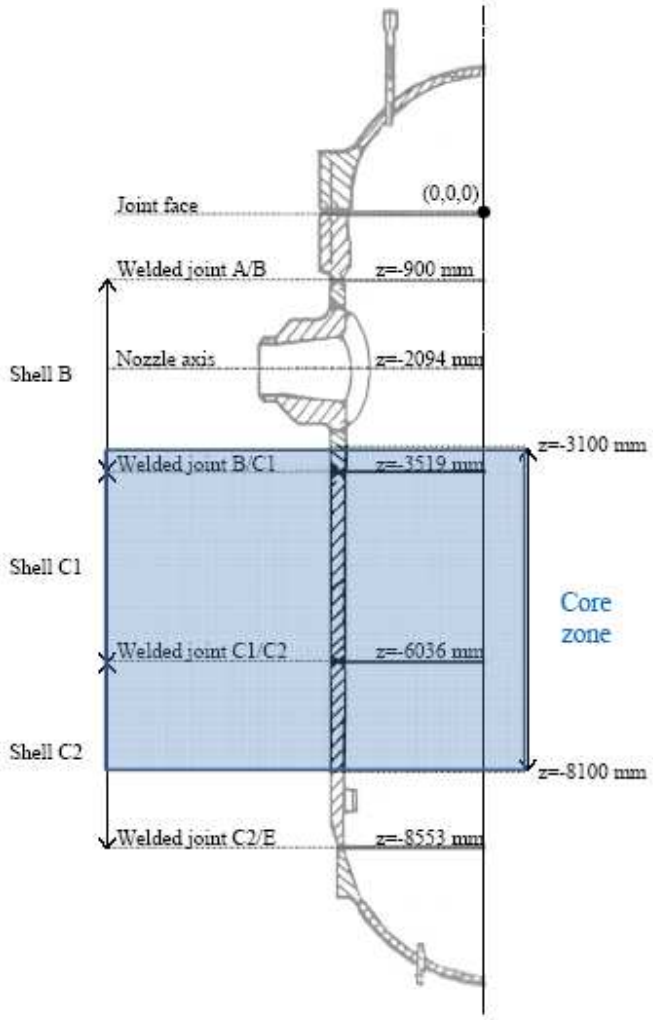


Figure 6.23: *Scheme of a reactor pressure vessel*

The vessel core is subjected to time-varying thermal and mechanical loadings, called *transients*, resulting from the operation of the reactor in the normal situation, but also in an incidental or accidental situation. Certain types of transients, like those induced by a *loss of coolant accident* (LOCA), may induce rapid cooldown of the RPV with relatively high or increasing system pressure, which is called the *pressurized thermal shock* (PTS). This induces high stress values inside the vessel wall which may load any pre-existing crack in the structure severely.

The structural integrity of the RPV during PTS is assessed by assuming the presence of a crack as sketched in Figure 6.24. In this work only longitudinal underclad cracks in the base metal are considered. Moreover we suppose a given LOCA-induced transient, which corresponds to a quite unlikely scenario.

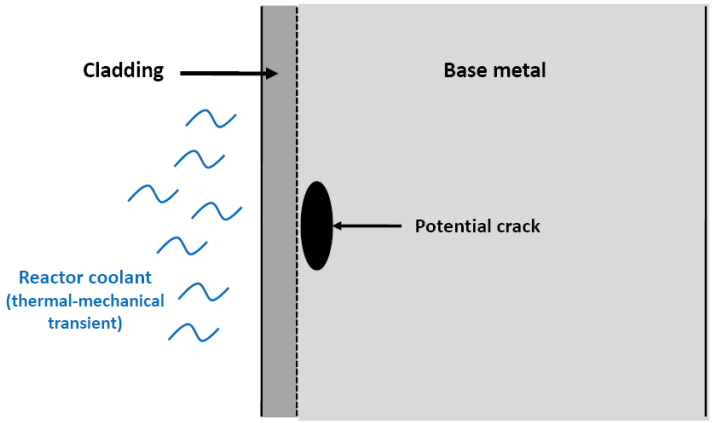


Figure 6.24: *RPV structural integrity - Scheme of a RPV subjected to a thermal-mechanical transient and featuring an underclad crack*

3.2.2 Deterministic assessment of the structure

The structural integrity of the RPV is assessed by estimating the risk of crack propagation in the structure. To this end one defines the following safety margin:

$$z(t) \equiv K_{IC}(t) - K_{\beta}(t) \tag{6.18}$$

where $K_{IC}(t)$ (resp. $K_{\beta}(t)$) is the material fracture toughness (resp. the *plastic stress intensity factor*) at the crack edge in the base metal (Figure 6.24), at instant t of the transient loading. The integrity of the structure is thus quantified by the *minimal* value of $z(t)$ during the transient, that is:

$$z_{min} \equiv \min_{t\in[0,T]} \left(K_{IC}(t) - K_{\beta}(t)\right) \tag{6.19}$$

where T is the total duration of the transient. Note that the following *margin factor* is also commonly used:

$$F_m \equiv \min_{t\in[0,T]} \left(\frac{K_{IC}(t)}{K_{\beta}(t)}\right) \tag{6.20}$$

From now on the shorthand K_{IC} (resp. K_{β}) is used to denote that value of the toughness (resp. plastic stress intensity factor) during the transient which leads to the minimal safety margin.

Computation of the plastic stress intensity factor K_{β}

The vessel materials are assumed to have a linear elastic constitutive law. This allows the analyst to compute the stress intensity factor using a *superposition principle*, which states that the stress intensity factor of the cracked structure is the same as would be produced by the application, to the edges of the crack, of stresses exerted on the uncracked structure (Figure 6.25). Estimating the stress intensity factor then relies upon the two following steps:

- compute the stress field on the uncracked structure;
- derive analytically the stress intensity factors by exerting the previously computed stress field to the crack edges.

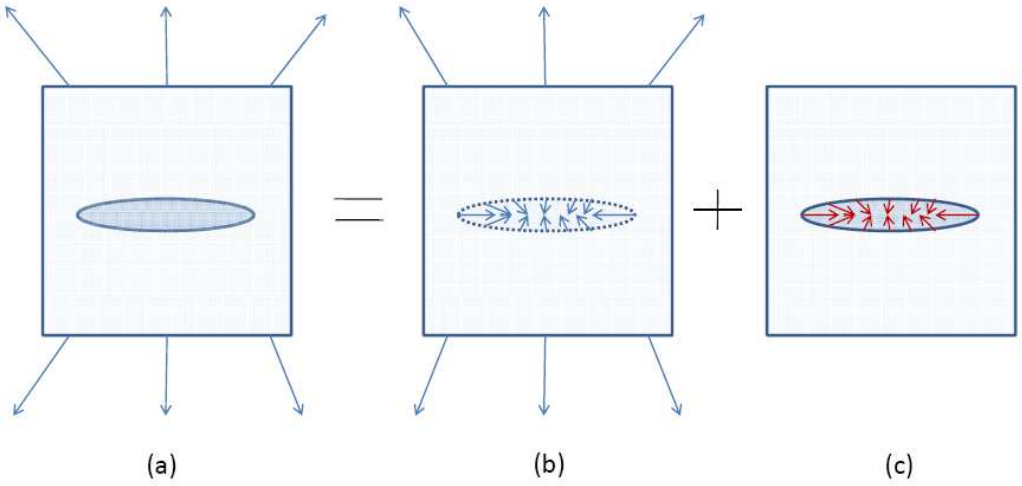


Figure 6.25: *RPV structural integrity - Superposition principle. (a) Cracked structure subjected to loading. (b) Uncracked structure subjected to the same loading. (c) Cracked structure loaded only at the crack location.*

The first point is dealt with by solving a thermal-mechanical problem (the thermal and mechanical problems are sketched in Figures 6.26 and 6.27, respectively). To this end, a one-dimensional axisymmetric model is built by means of EDF's finite element code Code_Aster (Figure 6.28). Generalized plane strain conditions are applied. Note that the thermal properties of the materials vary with temperature, hence a *non linear* thermal problem.

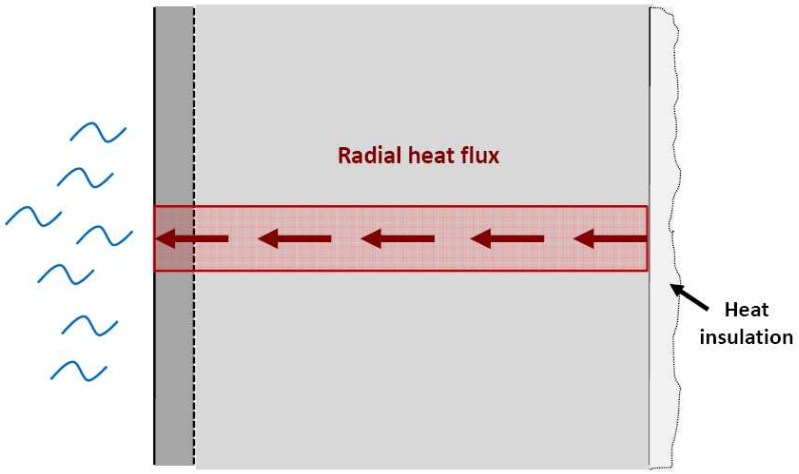


Figure 6.26: *RPV structural integrity - Scheme of the thermal problem*

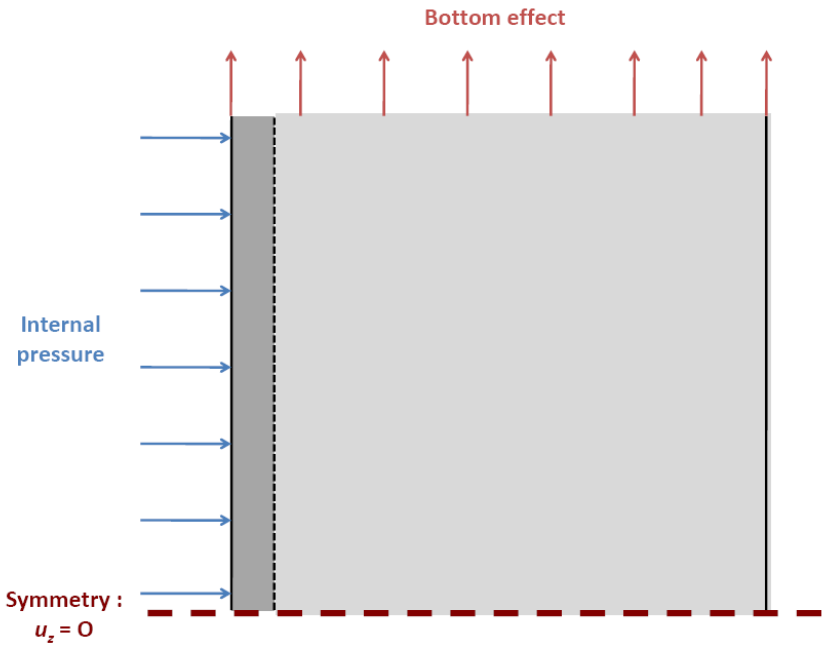


Figure 6.27: *RPV structural integrity - Scheme of the mechanical problem*

Let us assume the presence of an elliptical longitudinal underclad crack with height $2a$ and length $2b$ as depicted in Figure 6.29 (although such a definition of the height may be confusing, it is adopted because it corresponds to the conventional terminology in nuclear engineering). The points in the crack edges that are closest to and furthest of the cladding are denoted by A and B, respectively. One focuses on the stress intensity factor at point B, since the corresponding zone is more affected by brittle facture than the vicinity of A.

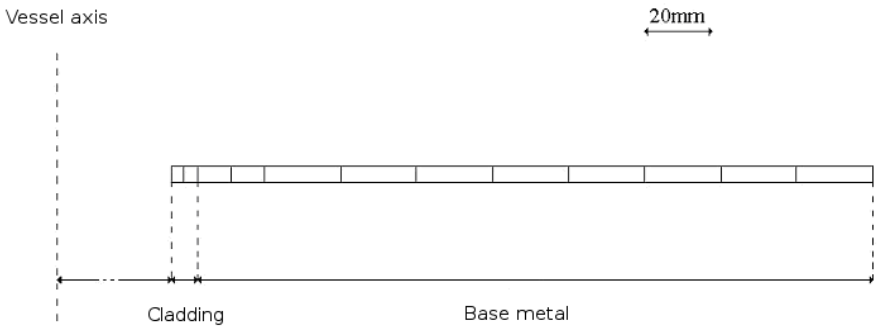


Figure 6.28: *RPV structural integrity - One-dimensional finite element model of the RPV built up by EDF's Code_Aster*

It is first assumed that the crack lies inside an infinite plate submitted to the stress field $\boldsymbol{\sigma}$ computed in the previous section. Thus the stress intensity factor reads:

$$K_{IB}^{\infty} = \int_{-a}^{a} \frac{\sigma_{\theta\theta}(x)}{\sqrt{\pi\, a}} \frac{a+x}{a-x}\, dx \tag{6.21}$$

where $\sigma_{\theta\theta}(x)$ corresponds to the orthoradial stresses.

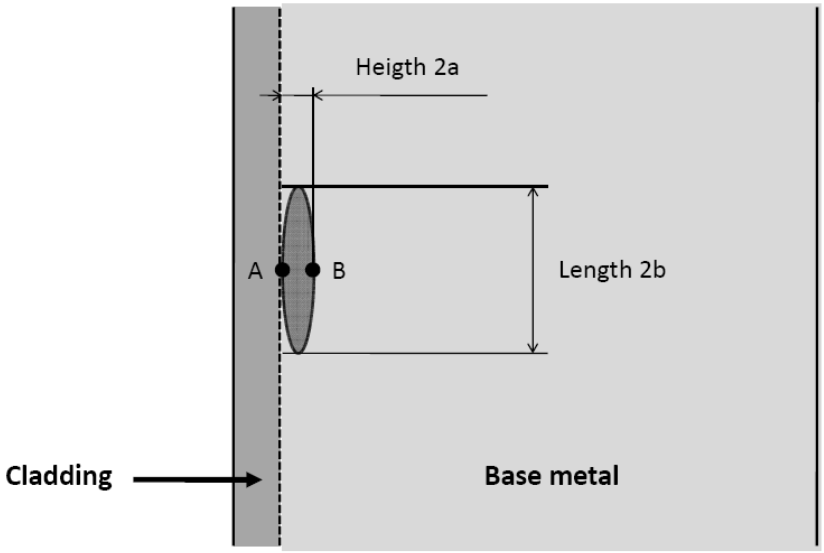


Figure 6.29: *RPV structural integrity - Geometry of a longitudinal underclad crack*

A correction is then applied take into account the fact that the structure is finite (factor F_{bB}) and that the crack is elliptical (factor f_B):

$$K_{IB} = F_{bB} \times f_B \times K_{IB}^{\infty} \tag{6.22}$$

Lastly, the yielding that occurs at the crack tip is taken into account by the so-called β *correction*, which is specific to underclad cracks bonded to the interface (the procedure is detailed in French Standard AFCEN (1997)). This leads to the *plastic stress intensity factor* K_{β}.

Computation of the toughness K_{IC}

The steel of the vessel is embrittled due to direct exposition to irradiation from the reactor core. This embrittlement may be measured by an increase, denoted by ΔRT_{NDT}, of the reference transition temperature RT_{NDT} (Figure 6.30), which leads to a decrease of the fracture toughness K_{IC}.

Using fluence data[1] and the chemical composition of the 16MND5 steel (*i.e.* mass concentration of copper, phosphorus and nickel), ΔRT_{NDT} may be estimated from prediction formulæ. A curve yielding the toughness as a function of temperature is taken as the reference curve for all materials in the non irradiated conditions. This curve is a minimum envelope of the toughness results found in the context of the American HSST program (Heavy Section Steel Technology). To obtain the curve corresponding to one specific material, the initial ductile-brittle transition temperature (denoted by RT_{NDT}^{ini}) of the material is shifted to the origin of the foregoing curve.

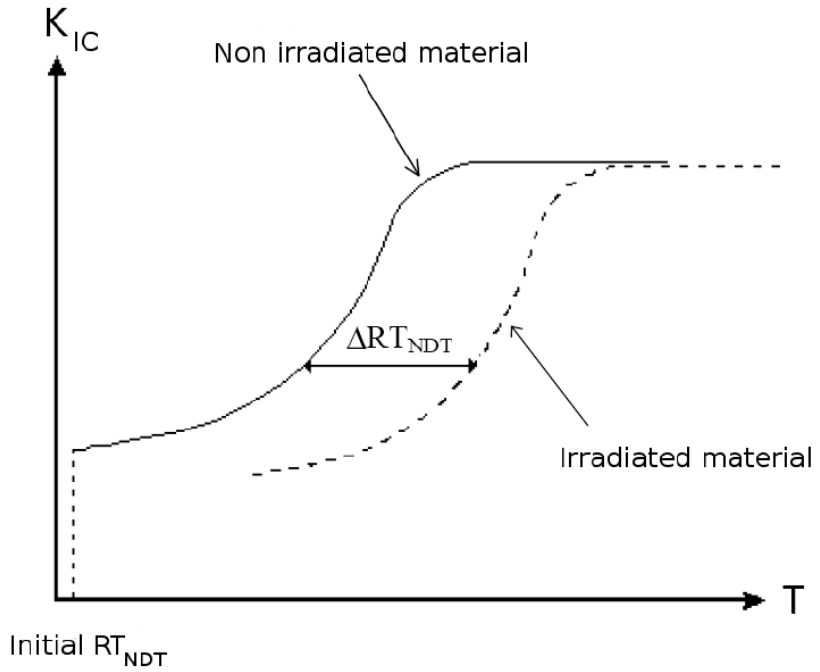


Figure 6.30: *RPV structural integrity - Correction of the toughness curve due to the fluence-induced embrittlement*

[1]Fluence is defined as the number of neutrons per cm^2 that have attained the vessel wall from the beginning of the reactor life.

3.3 Probabilistic analysis

3.3.1 Probabilistic model

The type and parameters of the random variables used in the analysis are listed in Table 6.11. Some parameters obtained from measurements on real RPV's have been omitted for the sake of confidentiality. Note that the toughness K_{IC} of the vessel base metal is assumed to follow a Gaussian distribution, which is quite unrealistic (a lognormal or a Gamma distribution would have been more appropriate to represent such a positive physical property). However, this probabilistic modelling will be adopted in this illustration since it corresponds to the statistical assumptions made in the French Standard RCC-M/ASME.

Table 6.11: *RPV structural integrity - Definition of the random variables*

Random variable	Type	Mean value	CV (%)
Crack height $h \equiv 2a$	Weibull	-	-
Crack aspect ratio $c \equiv a/b$	Lognormal	-	-
Cladding thickness e	Uniform	-	-
Mass concentration of copper Cu	Normal	0.086 %	12
Mass concentration of phosphorus P	Normal	0.0137 %	7
Mass concentration of nickel Ni	Normal	0.72 %	7
Fluence parameter Φ	Normal	μ_Φ †	-
Transition temperature offset ΔRT_{NDT}	Normal	$\mathcal{F}(Cu, P, Ni, \Phi)$	-
Initial transition temperature RT_{NDT}^{ini}	Normal	-	-
Toughness K_{IC}	Normal	$\mathcal{H}(T, RT_{NDT}^{ini}, \Delta RT_{NDT})$	15

† The mean value μ_Φ will be considered either as a parameter or as a random variable in the sequel

Besides, two variables have mean values that are functions of other random parameters, namely the transition temperature offset ΔRT_{NDT} and the base metal toughness K_{IC}. The mean value of random variable ΔRT_{NDT} is assumed to be a function $\mathcal{F}(Cu, P, Ni, \Phi)$ which corresponds to a prediction formula. Thus the randomness of ΔRT_{NDT} is related to the *intrinsic* variability of the transition temperature offset for fixed fluence parameter and chemical composition of the base metal. Similarly, the mean value of random variable K_{IC} is assumed to be a function $\mathcal{H}(T, RT_{NDT}^{ini}, \Delta RT_{NDT})$ available in the Standard RCC-M/ASME, where T is the temperature at point B in Figure 6.29. Thus the randomness of K_{IC} is related to the *intrinsic* variability of the toughness for fixed transition temperature.

It is interesting to recast the problem in terms of *independent* input random variables. To this end, one begins with rewriting random variable ΔRT_{NDT} as follows:

$$\Delta RT_{NDT} \;=\; \xi_1\,\sigma_1 \,+\, \mathcal{F}(Cu, P, Ni, \Phi) \qquad , \qquad \xi_1 \sim \mathcal{N}(0,1) \tag{6.23}$$

where σ_1 denotes the standard deviation of ΔRT_{NDT}. This allows one to recast the mean value of K_{IC} as follows: $\mu_{K_{IC}} \equiv \mathcal{H}(T, RT_{NDT}^{ini}, Cu, P, Ni, \Phi)$. Then the following mapping is used:

$$K_{IC} \;=\; \xi_2\,\sigma_2 \,+\, \mathcal{H}\left(T, RT_{NDT}^{ini}, \Delta RT_{NDT}(Cu, P, Ni, \Phi)\right) \qquad , \qquad \xi_2 \sim \mathcal{N}(0,1) \tag{6.24}$$

where σ_2 denotes the standard deviation of K_{IC}.

Besides, the temperature T which appears in the list of arguments of function $\mathcal{H}$ is obtained by solving the finite element problem. The latter takes $\boldsymbol{X}_{FE} \equiv \{h, c, e\}^{\mathsf{T}}$ as input parameters. Therefore the mean toughness may be cast formally as:

$$\mu_{K_{IC}} = \mathcal{H}\left(T(h, c, e), RT_{NDT}^{ini}, \Delta RT_{NDT}(Cu, P, Ni, \Phi)\right) \tag{6.25}$$

and the random variable K_{IC} as a function of random vector:

$$\boldsymbol{X} \equiv \{h, c, e, T_{NDT}^{ini}, Cu, P, Ni, \Phi, \xi_1, \xi_2\}^{\mathsf{T}} \tag{6.26}$$

whose components are independent. On the other hand, the plastic stress intensity factor K_β is also obtained by a finite element analysis, hence its dependence on random vector $\boldsymbol{X}_{FE}$. As a consequence, the safety margin may be eventually cast as a function of independent random variables as follows:

$$Y \;\equiv\; K_{IC}(\boldsymbol{X}) \;-\; K_\beta(\boldsymbol{X}_{FE}) \;=\; \mathcal{M}(\boldsymbol{X}) \tag{6.27}$$

The computational scheme of the calculation of the safety margin is sketched in Figure 6.31 for the sake of clarity.

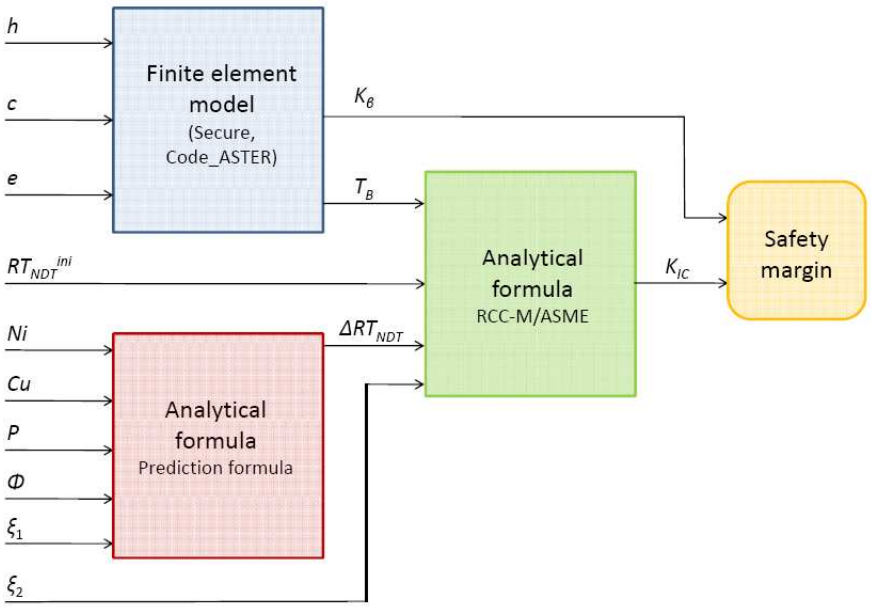


Figure 6.31: *RPV structural integrity - Computational scheme of the calculation of the safety margin*

3.3.2 Global sensitivity analysis

The evolution in time of the global sensitivity of the random variable $Y \equiv \mathcal{M}_{\mu_\Phi}(\boldsymbol{X})$ to each input parameter $\{X_i, i = 1, \ldots, M\}$ and interactions thereof is considered. To this end, a parametric study is carried out varying the mean value μ_Φ of the fluence parameter Φ. In this study, parameter μ_Φ takes successively the values $\{5, 6.25, 7.5, 8.75, 10\} \times 10^{19}$ n.cm^{-2}. This corresponds to values of the age of the powerplant equal to 20, 30, 40, 50 and 60 years. The related safety margins are then denoted by $\mathcal{M}_{\mu_\Phi}(\boldsymbol{X})$. In this case, $\boldsymbol{X}$ is of size 10.

Each model response $\mathcal{M}_{\mu_\Phi}(\boldsymbol{X})$ is approximated by a generalized PC representation. The PC basis is thus made both of normalized Legendre (for the uniform random variables) and Hermite (for the other random variables) polynomials. An hyperbolic truncation scheme is adopted using a $q = 0.4$-norm. Sparse PC expansions are determined using the adaptive LAR procedure. The target accuracy ε_{tgt} is set equal to 0.01. Lastly, the PC coefficients are computed using a sequential experimental design made of Sobol' quasi-random numbers. The initial size N_{ini} and the number of additional points when detecting overfitting N_{add} are both set equal to 100. The properties of the various PC approximations are gathered in Table 6.12. It appears that the various PC approximations could be determined at a maximum computational cost of 600 model evaluations. All these metamodels contain a low number of terms, with an index of sparsity IS_2 approximately ranging from 10% to 30%.

Table 6.12: *RPV structural integrity - Properties of the LAR-based sparse polynomial chaos expansions of the safety margin at several ages of the powerplant. The target accuracy is* $\varepsilon = 0.01$.

Age (in years)	20	30	40	50	60
PC degree	11	11	13	11	11
Number of PC terms	73	42	114	32	50
Index of sparsity IS_1	0.5%	0.5%	9×10^{-4}	0.5%	0.5%
Index of sparsity IS_2	32%	19%	23%	9%	15%
Number of model evaluations	200	400	600	300	300

Estimates of the Sobol' indices are obtained by elementary algebraic operations on the PC coefficients (Chapter 3, Section 5.2). The results are presented in Figure 6.32.

It appears that the most important variables are, in descending order, the intrinsic variability of toughness, the crack height and the intrinsic variability of the transition temperature offset. The effects of the cladding thickness, the mass concentration of phosphorus as well as all the interaction effects are negligible. No significant evolution in time of the values of the Sobol' indices is observed. It may be noticed though that the importance of the crack height keeps increasing in time up to a value greater than 30% at 60 years. The impact of all other variables decreases accordingly, except the one of the fluence parameter which remains rather constant.

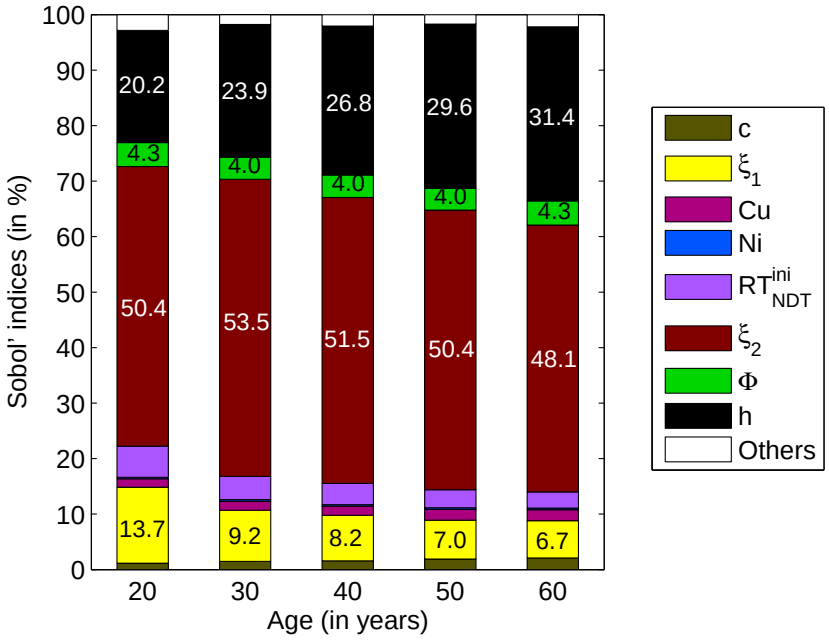


Figure 6.32: *RPV structural integrity - Evolution in time of the Sobol' indices*

3.3.3 Second moment analysis

The second moments of the safety margins are of interest. Estimates of the mean value and the standard deviation of the safety margins are obtained inexpensively by post-processing the PC coefficients. The two-standard deviation intervals $\mu_Y \pm 2\sigma_Y$ are depicted in Figure 6.33.

As expected, the mean value of the safety margin decreases as the irradiation fluence (*i.e.* the RPV age) increases. Such an evolution appears to be almost linear. On the other hand, the standard deviation does not vary much, with a coefficient of variation only increasing from 28% to 33%. This means that the nominal fluence μ_Φ has little impact on the dispersion of $\mathcal{M}_{\mu_\Phi}(\boldsymbol{X})$.

3.3.4 Distribution analysis

The distribution of the safety margins is now of interest. Characterizing the response PDF requires the derivation of more accurate PC approximations. This is achieved by running the LAR algorithm with a target error $\varepsilon_{tgt} = 0.005$. One considers the ages 20, 40 and 60 years of the vessel.

The safety margin function at 20 years appears to be much smoother than those at 40 and 60 years. Indeed, only $N = 400$ model evaluations are necessary to reach the target accuracy. It is also less sparse, with an index of sparsity IS_2 close to 50% instead of 20%. The construction of accurate approximations for the safety margin at 40 and 60 year requires a greater computational

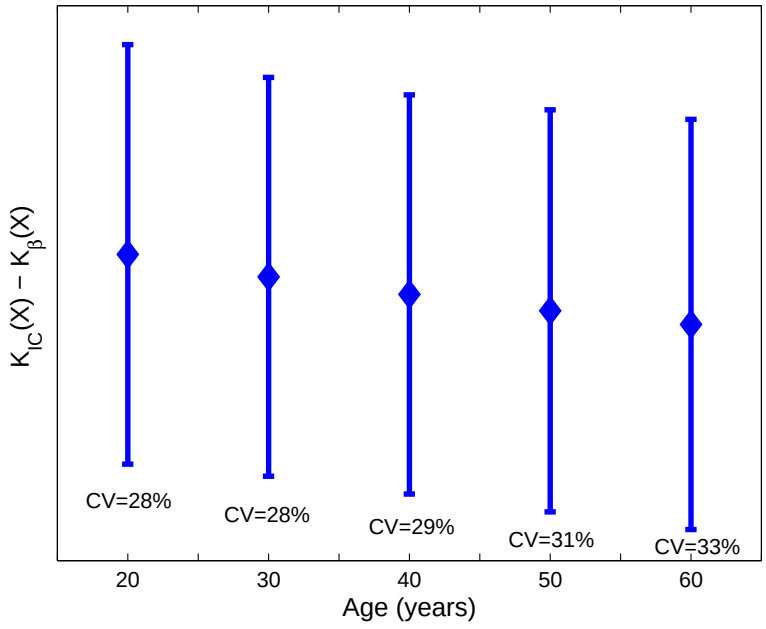


Figure 6.33: *RPV structural integrity - Evolution in time of the second moments of the safety margin (labels have been discarded from the y-axis for the sake of confidentiality). The error bars correspond to two-standard deviation intervals around the mean value.*

Table 6.13: *RPV structural integrity - Properties of the LAR-based sparse polynomial chaos expansions of the safety margin at several ages of the powerplant. The target accuracy is* $\varepsilon = 0.005$.

Age (in years)	20	40	60
PC degree	9	17	17
Number of terms	94	167	185
Index of sparsity IS_1	0.2%	10^{-4}	10^{-4}
Index of sparsity IS_2	53%	18%	20%
Number of model evaluations	400	1,100	1,400

effort, say $N = 1,100 - 1,400$.

The evolution in time of the probability density function (PDF) is represented in a linear (resp. logarithmic) scale in Figure 6.34 (resp. Figure 6.35). As expected, a shift to the left of the safety margin PDF is observed as the age of the powerplant increases.

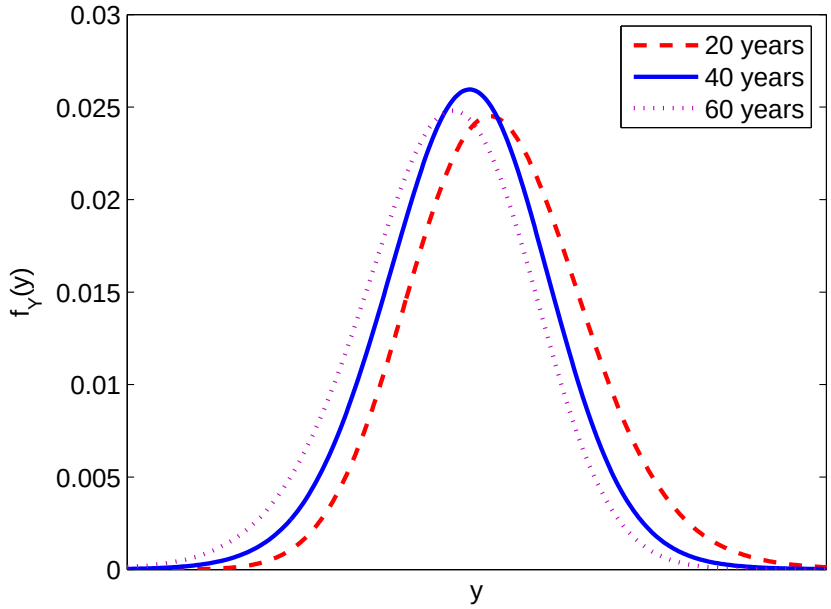


Figure 6.34: *RPV structural integrity - Evolution in time of the probability density function (labels have been discarded from the x-axis for the sake of confidentiality)*

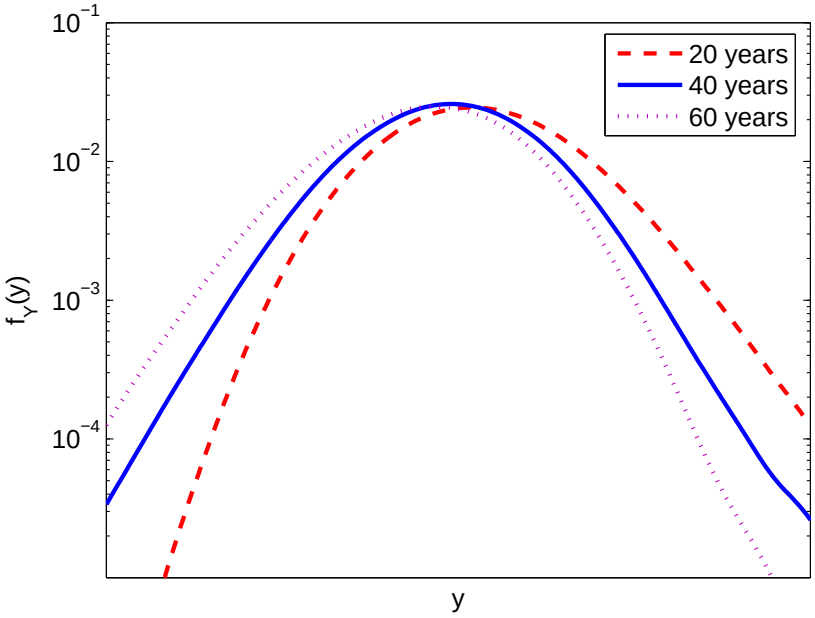


Figure 6.35: *RPV structural integrity - Evolution in time of the log-probability density function (labels have been discarded from the x-axis for the sake of confidentiality)*

3.3.5 Reliability analysis

The integrity of the pressure vessel is now assessed. The limit state function of the associated reliability problem is exactly the safety margin itself:

$$g(\boldsymbol{x}) \;=\; \mathcal{M}_{\mu_\Phi}(\boldsymbol{x}) \;\leq\; 0 \tag{6.28}$$

The probability of failure P_f is alternatively estimated by post-processing the accurate PC expansions used in the previous section. The results are depicted in Figure 6.36 in terms of generalized reliability indices $\widehat{\beta} \equiv -\Phi^{-1}(\widehat{P_f})$. As expected, the probability of failure increases in time. In particular, a sharp increase is observed from 30 years.

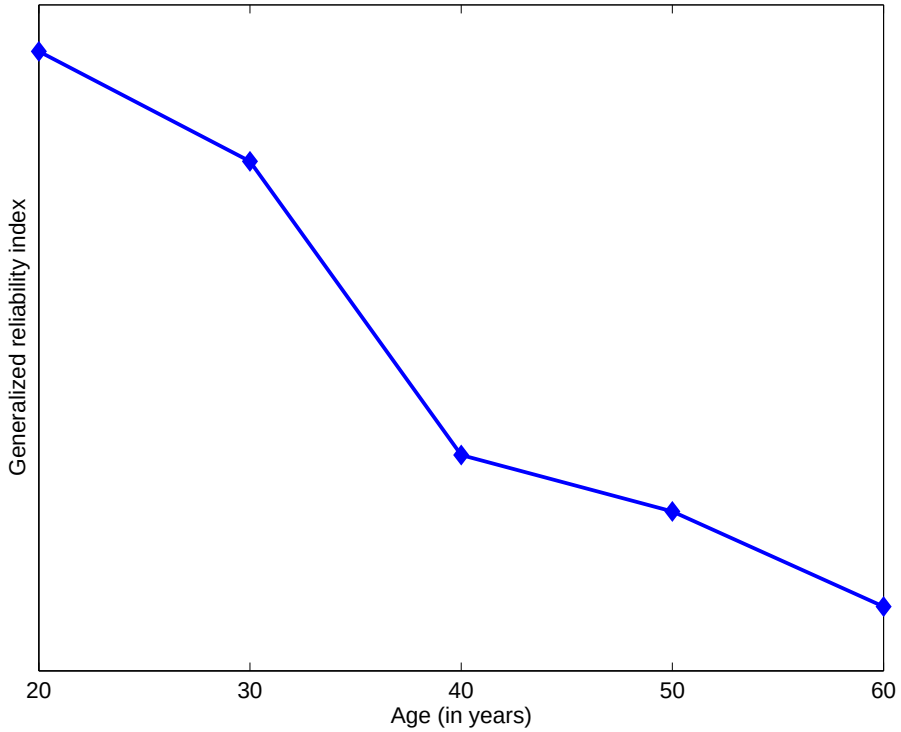


Figure 6.36: *RPV structural integrity - Evolution in time of the generalized reliability index (labels have been discarded from the y-axis for the sake of confidentiality)*

3.3.6 Considering the nominal fluence as an input parameter

Instead of carrying out a parameteric study with respect to the nominal fluence factor μ_Φ, the latter is now regarded as an additional input parameter of the model $\mathcal{M}$ for computing the safety margin. In this context, μ_Φ is modelled by a *random* variable uniformly distributed over $[5, 10]$. Hence one deals with the model $Y \equiv \mathcal{M}(\boldsymbol{X}')$, where $\boldsymbol{X}' \equiv \{\boldsymbol{X}^\mathsf{T}, \mu_\Phi\}^\mathsf{T}$. The number of input parameters is $M = 11$. The model response $\mathcal{M}(\boldsymbol{X}')$ is approximated by a sparse PC expansion $\widehat{Y} \equiv \widehat{\mathcal{M}}(\boldsymbol{X}')$ using the adaptive LAR procedure (uniform variable μ_Φ is associated with normalized Legendre polynomials). The target accuracy is first set equal to $\varepsilon = 0.01$.

The LAR-based metamodel $\widehat{\mathcal{M}}$ is obtained by means of $N = 600$ evaluations of the model $\mathcal{M}$. It allows the analyst to perform a parametric study on a fine grid of values of the mean fluence parameter μ_Φ. To this end, one has to derive statistics of a *conditional* PC approximation as shown in the following.

Conditional polynomial chaos approximation

Let us consider the following *conditional* PC approximation:

$$\widehat{\mathcal{M}}_\phi(\boldsymbol{X}) \;\equiv\; \left(\mathbb{E}\left[\widehat{\mathcal{M}}(\boldsymbol{X},\mu_\Phi) \;\middle|\; \mu_\Phi = \phi\right]\right) \;=\; \widehat{\mathcal{M}}(\boldsymbol{X},m) \tag{6.29}$$

This rewrites:

$$\widehat{\mathcal{M}}_\phi(\boldsymbol{X}) \;=\; \sum_{\{\boldsymbol{\alpha}_{-M},\alpha_M\}\in\mathcal{A}} a_{\boldsymbol{\alpha}_{-M},\alpha_M}\; \psi_{\boldsymbol{\alpha}_{-M}}(\boldsymbol{X})\; L_{\alpha_M}(u(\phi)) \tag{6.30}$$

where $\boldsymbol{\alpha}_{-M}$ is the index set containing the $(M-1)$ first components of $\boldsymbol{\alpha}$, $u(\phi)$ is the rescaled mean fluence parameter to $[-1,1]$ defined by:

$$u(\phi) \;\equiv\; 2\,\frac{\phi-5}{5} \;-\; 1 \tag{6.31}$$

and L_k denotes the normalized one-dimensional Legendre polynomial of degree k.

Second moment analysis

The conditional mean of the safety margin reads:

$$\mu_{\widehat{Y}}(\phi) \;\equiv\; \mathbb{E}\left[\widehat{\mathcal{M}}_\phi(\boldsymbol{X})\right] \;=\; \sum_{\substack{\boldsymbol{\alpha}\in\mathcal{A}\\ \boldsymbol{\alpha}=\{\boldsymbol{\alpha}_{-M},\alpha_M\}}} a_{\boldsymbol{\alpha}_{-M},\alpha_M}\; L_{\alpha_M}(u(\phi))\,\mathbb{E}\left[\psi_{\boldsymbol{\alpha}_{-M}}(\boldsymbol{X})\right] \tag{6.32}$$

As the mathematical expectation $\mathbb{E}\left[\psi_{\boldsymbol{\alpha}_{-M}}(\boldsymbol{X})\right]$ is non zero if and only if $\boldsymbol{\alpha}_{-M} \neq \mathbf{0}_{M-1}$ ($\mathbf{0}_{M-1}$ is the index set containing $M-1$ zero elements), this rewrites:

$$\mu_{\widehat{Y}}(\phi) \;=\; \sum_{\substack{\boldsymbol{\alpha}\in\mathcal{A}\\ \boldsymbol{\alpha}=\{\mathbf{0}_{M-1},\alpha_M\}}} a_{\boldsymbol{\alpha}}\; L_{\alpha_M}(u(\phi)) \tag{6.33}$$

Moreover, the conditional variance reads:

$$\sigma^2_{\widehat{Y}}(\phi) \;\equiv\; \mathbb{V}\left[\widehat{\mathcal{M}}_\phi(\boldsymbol{X})\right] \;=\; \sum_{\substack{\boldsymbol{\alpha}\in\mathcal{A}\\ \boldsymbol{\alpha}=\{\boldsymbol{\alpha}_{-M},\alpha_M\}}} a^2_{\boldsymbol{\alpha}_{-M},\alpha_M}\; L^2_{\alpha_M}(u(\phi))\,\mathbb{V}\left[\psi_{\boldsymbol{\alpha}_{-M}}(\boldsymbol{X})\right] \tag{6.34}$$

The variance $\mathbb{V}\left[\psi_{\boldsymbol{\alpha}_{-M}}(\boldsymbol{X})\right]$ is equal to 1 if $\boldsymbol{\alpha}_{-M} \neq \mathbf{0}_{M-1}$ and 0 otherwise. Thus the above equation reduces to:

$$\sigma^2_{\widehat{Y}}(\phi) \;=\; \sum_{\substack{\boldsymbol{\alpha}\in\mathcal{A}\\ \boldsymbol{\alpha}_{-M}\neq\mathbf{0}_{M-1}}} a^2_{\boldsymbol{\alpha},\alpha_M}\; L^2_{\alpha_M}(u(\phi)) \tag{6.35}$$

The evolution in time of the mean value as well as the two-standard deviation intervals $\mu_{\widehat{Y}} \pm 2\sigma_{\widehat{Y}}$ of the safety margin is plotted in Figure 6.37. The fluence ϕ is transformed into time for representing the results as a function of the RPV age. It appears that the conditional moments perfectly agree with the moments computed in the parametric study. Note that the *total* number of model evaluation is 600, instead of $1,900$ when carrying out the parametric study.

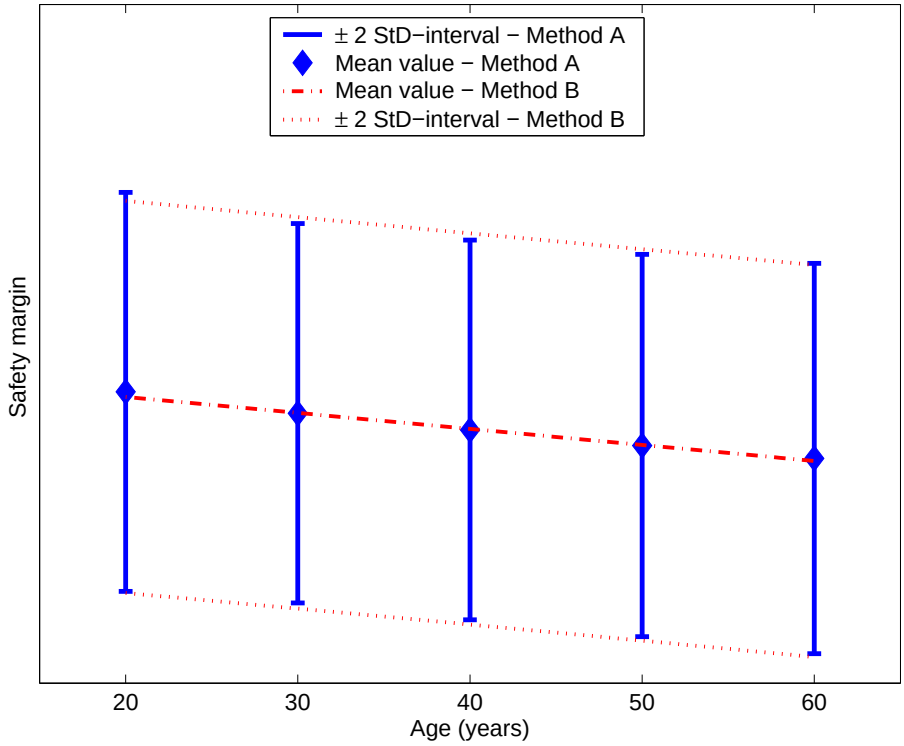


Figure 6.37: *RPV structural integrity - Evolution in time of the second moments of the safety margin (labels have been discarded from the y-axis for the sake of confidentiality). Method A is based on the parametric study varying the nominal fluence. Method B relies upon the derivation of the conditional response of the model that takes the mean fluence as a fake random variable.*

Reliability analysis

In order to estimate the probability of failure at several ages of the powerplant, the model response $Y \equiv \mathcal{M}(\boldsymbol{X})$ is approximated by a sparse approximation using the LAR procedure with a target value set equal to $\varepsilon_{tgt} = 0.005$. The required number or runs of the model is $N = 1,400$, instead of $2,900$ when carrying out a parametric study as in Section 3.3.5. The results are depicted in Figure 6.38 in terms of generalized reliability indices $\widehat{\beta} \equiv -\Phi^{-1}(\widehat{P_f})$, together with the estimates obtained in the parametric study.

The results provided by both methods agree quite well, with a maximal relative discrepancy of 7.5%. In particular, methods A and B yield very similar estimates for relatively low values of the probability of failure, *i.e.* when the age of the vessel attains 50 years. A larger discrepancy

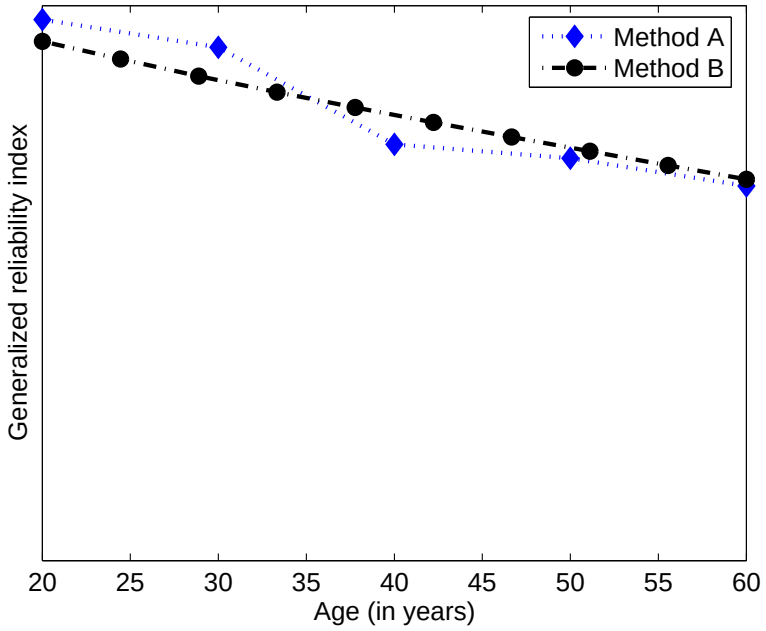


Figure 6.38: *RPV structural integrity - Evolution in time of the generalized reliability index (labels have been discarded from the y-axis for the sake of confidentiality). Method A is based on the parametric study varying the nominal fluence. Method B relies upon the derivation of the conditional response of the model that takes the fluence as an input variable.*

is observed though for smaller values of the age. This could be expected since the associated probabilities of failure are smaller, and then an accurate approximation of the lower tail of the limit state function PDF is required. Such an estimation may be very sensitive to changes in the approximation process.

3.4 Conclusion

The integrity of a reactor pressure vessel submitted to a thermal-mechanical transient loading has been investigated using the method of sparse polynomial chaos (PC) expansions. The safety margin of the structure, considered as the response of a model with random input parameters, has been approximated by a sparse metamodel using the adaptive LAR procedure. Then the second moments, the probablity density function and the sensitivity indices of the quantity of interest could be inexpensively computed from the PC coefficients.

First, the method has been applied for several ages of the nuclear powerplant, which corresponds to various degrees of embrittlement of the vessel due to irradiation. As an alternative, the age has been considered as an additional input variable, allowing a description of the evolution in time of the stochastic features of the safety margin by means of a *unique* PC approximation. Both approaches yielded identical results for second moment analysis. The second method

allows a description of the evolution in time of the stochastic response on a fine grid of fluence parameters (equivalently of ages of the powerplant), through the analytical computation of conditional properties.

Three important variables have been identified by the sensitivity analysis, namely the intrinsic randomness of the toughness and of the transition temperature offset, and the crack height. In contrast, the impact of some parameters revealed quite insignificant on the response variance, namely the cladding thickness, the crack aspect ratio and the parameters describing the chemical composition of the base metal. Those parameters may be fixed to their nominal values in further investigation.

4 Conclusion

The various academic examples as well as the nuclear engineering problem reported in this chapter have shown the potential of the methods described in Chapters 4,5. Indeed, the use of adaptive sparse polynomial chaos expansions has proven accurate and efficient for conducting uncertainty, sensitivity and reliability analysis. The stepwise regression and LAR methods allowed one to tackle problems with a high nominal dimension (*i.e.* up to 100 random variables) using a relatively small number of model evaluations, whereas it would have been unaffordable using the classical full PC expansions. The adaptive LAR algorithm globally revealed more efficient than stepwise regression in Examples #1-4. Furthermore, the former also appeared to be much less time-consuming when a high accuracy is prescribed ($\varepsilon_{tgt} = 10^{-5}$) for the analysis of the truss structure (Example #2).

On the other hand, an efficient method for carrying a parametric study at a reduced computational cost was experimented in the industrial example. It consists in taking the parameter under consideration as a uniform input random variable of the problem. Such an approach allowed a computational gain factor 3 (resp. 2) when conducting a moment (resp. reliability) analysis.

Chapter 7

Conclusion

1 Summary and results

The purpose of this work was to develop an efficient adaptive metamodelling method which allows inexpensive and straightforward post-processing for uncertainty propagation and sensitivity analysis. A special interest has been given to the mastering of the approximation error.

Firstly, several standard methods for uncertainty propagation, mostly based on intensive simulation of the model, were reviewed (Chapter 2). Then a general look was taken at metamodelling techniques. The discussion was limited to a brief presentation of two approaches, namely Gaussian Process modelling and Support Vector Regression. However, metamodelling is a wide topic that has led to an abundant literature, for instance the recent book by Forrester et al. (2008). Note that for many applications, metamodels are only built to substitute a complex model for a simple approximation that is very fast to evaluate. In this respect, they are not always well-suited to a stochastic framework, in which a particular focus should be put on zones of the domain of variation of the input parameters associated with a high probability.

This is why the spectral methods were introduced in Chapter 3. They consist in expanding *a priori* the random model response onto a suitable functional basis. A relevant choice is a basis that is orthogonal with respect to the probabilistic distribution of the input parameters. This makes possible an easy post-processing for deriving quantities of interest such as second moments and indices of sensitivity. Polynomial bases have been considered throughout this work. The associated representations are well-known as *polynomial chaos* (PC) *expansions* and have been intensively used in stochastic finite element analysis. The special case of an elliptic boundary value problem, which is encountered quite often in elastic mechanics, has been detailed. Although relying upon a sound mathematical foundation, the method often leads to a huge system of coupled equations. A recent approach known as Generalized Spectral Decomposition (GSD) aims at bypassing this issue by constructing iteratively an optimal expansion of the

solution, *i.e.* which captures its main stochastic features by means of a reduced number of terms. Thus the large system is replaced with a series of small problems. Note that various method for stochastic finite element analysis are based on an adaptation of the governing equations of the model.

Nonetheless, in many industrial applications the relationship between the input and the output parameters is complex, since for instance several physical phenomena may be coupled. One may thus prefer *non intrusive* methods, which are metamodel-like approaches in that they only require a set of evaluations of the existing deterministic model. A recent technique, known as stochastic collocation, consists in interpolating the model response at a set of suitably chosen observations. Polynomial interpolation is often performed, resulting in a representation of the solution in a Lagrange basis. An alternative scheme, closer to the original applications of the PC expansions, is based on the identification of the coefficients of the solution onto a fixed polynomial basis. Several methods for estimating the PC coefficients have been reviewed, namely simulation, numerical quadrature and regression. The latter has retained our attention since it is theoretically much more efficient than simulation, and the necessary number of model evaluations is supposed to be less sensitive to an increase of the number of input parameters than the one required by quadrature. However, even the computional cost associated with regression blows up when dealing with many input variables, say more than ten.

Several remedees to this "curse of dimensionality" were proposed in Chapter 4. First, the classical truncation scheme of the PC expansions (for allowing a practical implementation) has been criticized. Indeed, it tends to include a large number of terms related to higher-order interactions in the PC representation. However, according to the so-called *sparsity-of-effects principle*, such interactions are often negligible in many physical applications. Accounting for this principle, truncation schemes aimed at favoring the main effects and the low-order interactions have been designed. They result in PC expansions containing a small number of significant terms, which may be computed using few runs of the model.

Second, we have proposed the concept of *sparse* PC expansions for representing the model response. In this setup, many PC coefficients are zero. Of course, the few remaining non zero terms cannot be known in advance. Thus a stepwise regression procedure was devised in order to automatically detect the significant coefficients. A stopping criterion has been developed, based on an original robust and inexpensive error estimate, namely the *corrected leave-one-out error*. The algorithm eventually allows the analyst to build up an adaptive metamodel while monitoring the approximation error. Another objective was to minimize the number of runs to the deterministic model. In this purpose, an automatic enrichment of the computer experimental design has been set up, such that one only adds points (and possibly not too many of them) to the existing design when necessary. An heuristic criterion based on the numerical stability of the regression problems was retained.

Possible improvements of the adaptive procedure were examined from the viewpoint of *variable selection* (Chapter 5), which is a major problem in statistics. A simple and very efficient scheme has retained our attention, namely Least Angle Regression (LAR). Indeed, this method is well-suited to situations in which the number of PC terms (or predictors) is great compared to the number of computer experiments, and possibly significantly larger. LAR provides a set of more or less sparse PC approximations at the same cost as an ordinary least-square regression. We proposed an efficient modified cross-validation scheme (MCV) in order to select the best metamodel among the whole set of solutions. Similarly to the stepwise regression scheme discussed above, the combination of LAR and MCV was integrated into an iterative algorithm, which adapts both the candidate PC basis and the experimental design.

Finally, stepwise regression and adaptive LAR were applied to several academic problems in Chapter 6, namely one analytical function and finite element models of mechanical systems, involving $10 - 100$ input variables. Both schemes revealed quite efficient to derive statistical moments and sensitivity indices, providing accurate estimates using always less than 400 model evaluations. The methods provided satisfactory estimates of probabilities of failure as well. In this case, more accurate PC approximations had to be constructed, requiring up to $1,200$ model evaluations. Note that the computation of accurate classical full PC expansions was generally simply not affordable in the various examples due to the large number of input parameters. By and large, the LAR procedure generally overperformed stepwise regression in terms of trade-off between accuracy and number of runs to the model. Furthermore, it was shown in the second example that LAR is also more efficient in terms of computational processing time for fitting a sparse metamodel. This is why the LAR algorithm was chosen to be applied to an industrial problem, namely the integrity of the pressure vessel of a nuclear powerplant. The quantity of interest was the safety margin to brittle fracture of the vessel. The method allowed one to compute the second moments and the sensitivity indices using 600 runs to the model. It appeared that the variability in the safety margin was mainly explained by the intrinsic uncertainty in the toughness and the ductile to brittle transition temperature of the vessel metal, as well as the size of the possible cracks inside the structure. The evolution in time of the probability of the safety margin to be negative (*i.e.* the probability of failure) was also estimated.

2 Outlook

The two methods that have been devised, namely adaptive LAR and stepwise regression, fullfil the objectives which motivated this work. A nice feature of these schemes is that they can easily integrate several improvements at different levels.

Non polynomial chaos representations

We limited this work to chaos representations using polynomial bases. The reason is that polynomial spectral expansions have a whole branch of functional analysis behind them, see for instance Boyd (1989). In addition, the use of polynomial models has a long history in statistical science (Montgomery et al., 2001). Last but not least, as indicated in this document, *polynomial* chaos expansions have been extensively employed in stochastic finite element analysis from the pioneering work in Ghanem and Spanos (1991).

However, one may expect polynomial representations to be unsuited to problems featuring high non linearity or singularities. Other types of basis satisfying the appropriate orthogonality conditions could be considered. For instance, a chaos representation made of sinusoids, namely the *Fourier chaos expansion*, has been used in aerodynamics (Millman et al., 2005) in order to handle an oscillatory response featuring a bifurcation. Moreover, the so-called *multi-element generalized polynomial chaos method*, which consists in decomposing the domain of variation of the input variables, has been successfully applied to long-term integration problems featuring discontinuities (Wan and Karniadakis, 2005). Lastly, wavelet-based decompositions (Antoniadis, 1997; Le Maître et al., 2004b) may be used as well.

Computer experimental designs based on active learning

In order to fit the adaptive PC metamodels, *space-filling* designs have been employed so far, namely Quasi-Monte Carlo (QMC) and the proposed Nested Latin Hypercube Sampling (NLHS) scheme. Note that both sampling methods should be compared on a wide range of examples in order to draw general conclusions with regard to their efficiency. Adopting a statistical learning terminology as in Castro et al. (2005), QMC and NLHS may be labelled as *passive learning* schemes insofar as the locations of the sample points are chosen independently of the previous model observations.

In contrast, an *active learning* strategy consists in choosing the location of a new sampling point as a function of the already collected ones. Such an approach has received great interest in Gaussian Process (GP) modelling. Indeed, the ability to predict *a priori* the prediction error at *any* location in the domain of variation of the input parameters may lead to the following rule: add a new point at the location with highest prediction error (Santner et al., 2003). More sophisticated strategies have been devised. For instance, criteria for balancing *exploration* and *exploitation* of the GP metamodel are reviewed in Forrester et al. (2008, Chapter 3). Exploration consists in adding sampling points in zones with high uncertainty, whereas exploitation consists in adding points in relevant zones, such as the vicinity of a minimizer in optimization.

Such a trade-off would be particularly relevant in the framework of approximation by PC ex-

pansions. Indeed, a *global* approximation is sought, hence it is necessary to use sample points that properly reflect the distribution of the input parameters. This may be correctly achieved using Monte Carlo sampling (MCS), QMC or (N)LHS. In the same time, zones associated with a low probability should be also explored, such as the tails of the random model response when a reliability analysis is of interest. Thus developing appropriate criteria for managing the exploration-exploitation trade-off when building up a PC expansion will be beneficial.

Estimates of the approximation error in the tails of the response probability density function

In this work, only estimates of the approximation error in the $\mathcal{L}^2$-norm have been considered. The reason is that this norm is the natural one to assess the result of least-square regression calculations. Moreover, the related derivations are quite straightforward, with available analytical formulæ in order to evaluate the regression-based coefficients as well as the empirical and the leave-one-out error. However, the error in the $\mathcal{L}^2$-norm is little sensitive to the goodness of fit of the metamodel in the tails of the response PDF. Hence it may be unappropriate for reliability analysis. It will be then relevant to investigate error estimates which focus on a specified quantile or a probability of failure.

Adaptive polynomial chaos expansion based on a step-by-step discretization of random fields

The two methods developed in this work, namely adaptive LAR and stepwise, revealed efficient for producing sparse PC approximations of the model response at a low computational cost. The reduction of the number of terms in the PC expansion was particularly noticeable when considering examples featuring random fields. In order to achieve a high accuracy, the input random fields were discretized into a large number of input variables (say $38 - 100$), which correspond to the *nominal* dimensionality of the problem. However, it was shown that the model response only depended on few eigenmodes in the Karhunen-Loève expansion (say less than 5), hence a low *effective* dimensionality.

The principle of a PC decomposition onto an *anisotropic* basis (Chapter 4, Section 2.2.2) may help neglect some terms associated with insignificant variables. This promising approach should be suitably calibrated on various numerical examples. Nonetheless, it is our belief that the computational cost could be dramatically reduced by integrating an *iterative* discretization of the random fields into the proposed algorithms. The investigation of a suitable criterion for adding progressively the random variables into the metamodel will be of great interest.

Appendix A

Classical probability density functions

1 Gaussian distribution

The Gaussian distribution is usually used in statistics in order to model uncertain quantities when only their mean value and standard deviation are known. The support of a Gaussian random variable with mean μ and standard deviation σ is $\mathcal{D}_X = \mathbb{R}$ and its probability density function (PDF) reads:

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{A.1}$$

Accordingly, the cumulative distribution function (CDF) of $X \sim \mathcal{N}(\mu, \sigma)$ reads:

$$F_X(x) = \int_{-\infty}^{x} f_X(x)\, dx \tag{A.2}$$

A Gaussian random variable with mean $\mu = 0$ and $\sigma = 1$ is referred to as *standard*, and its PDF (resp.) is often denoted by φ (resp. Φ).

The skewness coefficient of a Gaussian random variable is $\delta = 0$ (symmetrical PDF) and its kurtosis coefficient is $\kappa = 3$. In other words, a Gaussian variable is completely characterized by its second moments.

2 Lognormal distribution

Lognormal distributions are commonly used to model positive quantities, such as material properties in mechanics. A lognormal random variable $X \sim \mathcal{LN}(\lambda, \zeta)$ is such that its logarithm is a

Gaussian random variable: $\log X \sim \mathcal{N}(\lambda, \zeta)$. In other words, denoting by ξ a standard normal variable, one gets:

$$X = \exp(\lambda + \zeta\,\xi) \tag{A.3}$$

The support of a lognormal distribution is $]0, +\infty[$ and its PDF reads:

$$f_X(x) = \frac{1}{\zeta\, x}\varphi\left(\frac{\log x - \lambda}{\zeta}\right) \tag{A.4}$$

Its CDF reads:

$$F_X(x) = \Phi\left(\frac{\log x - \lambda}{\zeta}\right) \tag{A.5}$$

Its mean value μ and standard deviation σ respectively read:

$$\begin{aligned} \mu &= \exp(\lambda + \zeta^2/2) \\ \sigma &= \mu\,\sqrt{e^{\zeta^2} - 1} \end{aligned} \tag{A.6}$$

Conversely, if the mean value and the standard deviation are known, the parameters of the distribution read:

$$\begin{aligned} \lambda &= \log \frac{\mu}{\sqrt{1 + CV^2}} \\ \zeta &= \sqrt{\log(CV^2 + 1)} \end{aligned} \tag{A.7}$$

where $CV = \sigma/\mu$ is the coefficient of variation of X.

3 Uniform distribution

Uniform distributions may be employed to model a parameter when only its *bounded* domain of variation is known. A uniform random variable $X \sim \mathcal{U}(a, b)$ is defined by the following PDF ($b > a$):

$$f_X(x) = 1/(b - a) \quad \text{for } x \in [a, b]\,, \quad 0 \text{ otherwise} \tag{A.8}$$

Its CDF reads:

$$F_X(x) = \frac{x - a}{b - a} \quad \text{for } x \in [a, b]\,, \quad 0 \text{ otherwise} \tag{A.9}$$

Its mean value and standard deviation read:

$$\begin{aligned} \mu &= \frac{a + b}{2} \\ \sigma &= \frac{b - a}{2\sqrt{3}} \end{aligned} \tag{A.10}$$

Its skewness coefficient is $\delta = 0$ and its kurtosis coefficient is $\kappa = 1.8$.

4 Gamma distribution

Just like the lognormal distributions, the Gamma distributions are useful to model parameters taking positive values. A random variable follows a Gamma distribution $\mathcal{G}(\kappa, \lambda)$ if its PDF reads:

$$f_X(x) = \frac{\lambda^k}{\Gamma(k)} x^{k-1} e^{-\lambda x} \quad \text{for } x \geq 0\,, \quad 0 \text{ otherwise} \tag{A.11}$$

where $\Gamma(k)$ is the Gamma function defined by:

$$\Gamma(k) = \int_0^\infty t^{k-1} e^{-t}\, dt \tag{A.12}$$

Its CDF reads:

$$F_X(x) = \gamma(k, \lambda x)/\Gamma(k) \quad \text{for } x \geq 0\,, \quad 0 \text{ otherwise} \tag{A.13}$$

where $\gamma(k, x)$ is the incomplete Gamma function defined by:

$$\gamma(k, x) = \int_0^x t^{k-1} e^{-t}\, dt \tag{A.14}$$

Its mean value and standard deviation read:

$$\begin{aligned} \mu &= k/\lambda \\ \sigma &= \sqrt{k}/\lambda \end{aligned} \tag{A.15}$$

Its skewness and kurtosis coefficients read:

$$\begin{aligned} \delta &= 2/\sqrt{k} \\ \kappa &= 3 + 6/k \end{aligned} \tag{A.16}$$

5 Beta distribution

A random variable follows a Beta distribution $\mathcal{B}(r, s, a, b)$ if its PDF reads:

$$f_X(x) = \frac{(x-a)^{r-1}(b-x)^{s-1}}{(b-a)^{r+s-1} B(r,s)} \quad \text{for } x \in [a, b]\,, \quad 0 \text{ otherwise} \tag{A.17}$$

where $B(r, s)$ is the Beta function defined by:

$$B(r,s) = \int_0^1 t^{r-1}(1-t)^{s-1}\, dt = \frac{\Gamma(r)\Gamma(s)}{\Gamma(r+s)} \tag{A.18}$$

Its CDF reads:

$$f_X(x) = \frac{1}{(b-a)^{r+s-1} B(r,s)} \int_a^x (t-a)^{r-1}(b-t)^{s-1}\, dt \quad \text{for } x \in [a, b]\,, \quad 0 \text{ otherwise} \tag{A.19}$$

Its mean value and standard deviation respectively read:

$$\begin{aligned} \mu &= a + (b-a)\frac{r}{r+s} \\ \sigma &= \frac{b-a}{r+s}\sqrt{\frac{r\,s}{r+s+1}} \end{aligned} \tag{A.20}$$

6 Weibull distribution

A random variable follows a Weibull distribution $\mathcal{W}(\alpha, \beta)$ if its CDF reads:

$$F_X(x) = 1 - \exp(-\left(\frac{x}{\alpha}\right)^{\beta}) \quad \text{for } x \geq 0\,, \quad 0 \text{ otherwise} \tag{A.21}$$

Its PDF reads:

$$f_X(x) = \frac{\beta}{\alpha}\left(\frac{x}{\alpha}\right)^{\beta-1} \exp(-\left(\frac{x}{\alpha}\right)^{\beta}) \quad \text{for } x \geq 0\,, \quad 0 \text{ otherwise} \tag{A.22}$$

Its mean value and standard deviation respectively read:

$$\begin{aligned} \mu &= \alpha\,\Gamma(1 + 1/\beta) \\ \sigma &= \alpha\left[\Gamma(1 + 2/\beta) - \Gamma(1 + 1/\beta)^2\right]^{1/2} \end{aligned} \tag{A.23}$$

Appendix B

Karhunen-Loève decomposition using a spectral representation of the autocorrelation function

1 Introduction

Let us consider a random field $H(\boldsymbol{x}, \omega)$, where $\boldsymbol{x}$ is a spatial variable in a *rectangular* domain $\mathcal{D} \subset \mathbb{R}^d$, $d \in \{1, 2, 3\}$ (*i.e.* $\mathcal{D} \in \prod_{j=1}^{d} [a_j, b_j]$) and ω is the elementary event of a probability space $(\Omega, \mathcal{F}, \mathcal{P})$. The *Karhunen-Loève expansion* of $H(\boldsymbol{x}, \omega)$ reads:

$$H(\boldsymbol{x}, \omega) \;=\; \mu(\boldsymbol{x}) \,+\, \sum_{i=1}^{+\infty} \sqrt{\lambda_i}\, \xi_i(\omega)\, \varphi_i(\boldsymbol{x}) \tag{B.1}$$

where $\mu(\boldsymbol{x})$ is the mean of the random field. $(\xi_i(\omega))_{i \in \mathbb{N}^*}$ is a sequence of uncorrelated, zero-mean and unit-variance random variables. $(\lambda_i)_{i \in \mathbb{N}^*}$ and $(\varphi_i(\boldsymbol{x}))_{i \in \mathbb{N}^*}$ are sequences that solve the following *Fredholm equations*:

$$\forall\, i \in \mathbb{N}^* \quad , \quad \int_{\mathcal{D}} C_H(\boldsymbol{x}, \boldsymbol{x}')\, \varphi_i(\boldsymbol{x}')\, d\boldsymbol{x}' \;=\; \lambda_i\, \varphi_i(\boldsymbol{x}) \tag{B.2}$$

where $C_H(\boldsymbol{x}, \boldsymbol{x}')$ denotes the autocorrelation function of the random field $H(\boldsymbol{x}, \omega)$. Although the problem in Eq.(B.2) admits a closed-form solution for particular choices of $C_H(\boldsymbol{x}, \boldsymbol{x}')$, it generally requires the implementation of a numerical solving scheme.

2 Spectral representation of the autocorrelation function

Let $\mathcal{H} = (h_i(\boldsymbol{x}))_{i\in\mathbb{N}}$ be a complete *orthonormal* basis of the Hilbert space $\mathcal{L}_2(\mathcal{D})$ of square-integrable functions over $\mathcal{D}$. The orthonormality property of the basis is given by:

$$\forall\, i,j \in \mathbb{N} \quad , \quad \langle h_i \,,\, h_j \rangle \;\equiv\; \int_{\mathcal{D}} h_i(\boldsymbol{x})\, h_j(\boldsymbol{x})\, d\boldsymbol{x} \;=\; \delta_{i,j} \tag{B.3}$$

where $\delta_{i,j} = 1$ if $i = j$ and 0 otherwise. Throughout this chapter one considers a basis $\mathcal{H}$ made of products of univariate normalized Legendre polynomials $L_j(x_j)$, $j = 1,\ldots,d$. Introducing a multi-index notation, the basis functions read:

$$\forall\, \boldsymbol{\alpha} \in \mathbb{N}^d \quad , \quad h_{\boldsymbol{\alpha}}(\boldsymbol{x}) \;=\; \prod_{j=1}^{d} L_{\alpha_j}(x_j) \tag{B.4}$$

As the domain $\mathcal{D}$ is assumed to be rectangular, one gets $\mathcal{L}_2(\mathcal{D}\times\mathcal{D}) \cong \mathcal{L}_2(\mathcal{D}) \otimes \mathcal{L}_2(\mathcal{D})$. Consequently, the tensor product $\mathcal{H}\otimes\mathcal{H}$ forms a complete orthonormal basis of $\mathcal{L}_2(\mathcal{D}^2)$. Assuming that the autocorrelation function of the random field under consideration is square-integrable, *i.e.* that $C_H(\boldsymbol{x},\boldsymbol{x}') \in \mathcal{L}_2(\mathcal{D}^2)$, it may be expanded as follows:

$$C_H(\boldsymbol{x},\boldsymbol{x}') \;=\; \sum_{\boldsymbol{\alpha}\in\mathbb{N}^d} \sum_{\boldsymbol{\beta}\in\mathbb{N}^d} c_{\boldsymbol{\alpha},\boldsymbol{\beta}}\, h_{\boldsymbol{\alpha}}(\boldsymbol{x})\, h_{\boldsymbol{\beta}}(\boldsymbol{x}') \tag{B.5}$$

Let us reorder the multi-indices as follows:

$$\mathbb{N}^d \;\equiv\; \{\boldsymbol{\alpha}\in\mathbb{N}^d\} \;\equiv\; \{\boldsymbol{\alpha}_i \,,\, i\in\mathbb{N}\} \tag{B.6}$$

Eq.(B.5) is equivalent to:

$$C_H(\boldsymbol{x},\boldsymbol{x}') \;=\; \sum_{i\in\mathbb{N}} \sum_{j\in\mathbb{N}} c_{\boldsymbol{\alpha}_i,\boldsymbol{\alpha}_j}\, h_{\boldsymbol{\alpha}_i}(\boldsymbol{x})\, h_{\boldsymbol{\alpha}_j}(\boldsymbol{x}') \tag{B.7}$$

Moreover, the functions $\varphi_i(\boldsymbol{x})$ in Eq.(B.2) may be recast in $\mathcal{H}$ as:

$$\forall\, i\in\mathbb{N}^* \quad , \quad \varphi_i(\boldsymbol{x}) \;=\; \sum_{j\in\mathbb{N}} d^i_{\boldsymbol{\alpha}_j}\, h_{\boldsymbol{\alpha}_j}(\boldsymbol{x}) \tag{B.8}$$

Introducing Eqs.(B.7),(B.8) into Eq.(B.2), it may be shown that the Fredholm equations reduce to the following *eigenvalue problem*:

$$\boldsymbol{C}\,\boldsymbol{D} \;=\; \boldsymbol{D}\,\boldsymbol{\Lambda} \tag{B.9}$$

where $\boldsymbol{C}$, $\boldsymbol{D}$ are infinite matrices which contain respectively the coefficients $c_{\boldsymbol{\alpha}_i,\boldsymbol{\alpha}_j}$ and $d^i_{\boldsymbol{\alpha}_j}$ in Eqs.(B.5),(B.8):

$$\forall\, i,j\in\mathbb{N} \quad , \quad \boldsymbol{C}_{ij} \;=\; c_{\boldsymbol{\alpha}_i,\boldsymbol{\alpha}_j} \tag{B.10}$$

$$\forall j\in\mathbb{N}\,,\, \forall\, k\in\mathbb{N}^* \quad , \quad \boldsymbol{D}_{jk} \;=\; d^k_{\boldsymbol{\alpha}_j} \tag{B.11}$$

and $\boldsymbol{\Lambda}$ is the infinite diagonal matrix that contains the eigenvalues λ_k:

$$\forall k,l\in\mathbb{N}^* \quad , \quad \boldsymbol{\Lambda}_{jl} \;=\; \delta_{k,l}\,\lambda_k \tag{B.12}$$

3 Numerical solving scheme

3.1 Finite dimensional eigenvalue problem

For computational purpose it is necessary to reduce the infinite eigenvalue problem in Eq.(B.9) to a standard *finite dimensional* eigenvalue problem. To this end the series in Eqs.(B.5),(B.8) may be truncated such that the retained basis multivariate polynomials have a total degree that does not exceed a given value. This leads to the approximations:

$$C_H(\boldsymbol{x},\boldsymbol{x}') \;\approx\; C_{H,p}(\boldsymbol{x},\boldsymbol{x}') \;=\; \sum_{|\boldsymbol{\alpha}|\leq p}\sum_{|\boldsymbol{\beta}|\leq p} c_{\boldsymbol{\alpha},\boldsymbol{\beta}}\; h_{\boldsymbol{\alpha}}(\boldsymbol{x})\; h_{\boldsymbol{\beta}}(\boldsymbol{x}') \tag{B.13}$$

$$\forall\, i\in\mathbb{N}^* \quad , \quad \varphi_i(\boldsymbol{x}) \;\approx\; \varphi_{i,p}(\boldsymbol{x}) \;=\; \sum_{|\boldsymbol{\alpha}|\leq p} d^i_{\boldsymbol{\alpha}}\; h_{\boldsymbol{\alpha}}(\boldsymbol{x}) \tag{B.14}$$

The number of terms in each summation is given by:

$$P \;=\; \binom{d+p}{p} \tag{B.15}$$

In order to obtain a well-posed eigenvalue problem, it is necessary to retain a number M of eigenfunctions $\varphi_i(\boldsymbol{x})$ not greater than P. One selects $M=P$ in the sequel. Thus the problem in Eq.(B.9) reduces to:

$$\boldsymbol{C}_P\;\boldsymbol{D}_P \;=\; \boldsymbol{D}_P\;\boldsymbol{\Lambda}_P \tag{B.16}$$

where the notation $\boldsymbol{M}_P$ represents the restriction of the infinite matrix $\boldsymbol{M}$ to its P first rows and columns.

Prior to solving the eigenvalue problem in Eq.(B.16), one has to select a suitable value for p (*i.e.* the size of the problem). Then the coefficients of the matrix $\boldsymbol{C}_P$ have to be computed. In the sequel, one proposes an iterative strategy which tackles progressively both issues in the same time.

3.2 Computation of the coefficients of the autocorrelation orthogonal series

Throughout this section, it is assumed that the truncation order p has been somehow selected for obtaining the approximation (B.13). Due to the orthonormality of the basis $\mathcal{H}\otimes\mathcal{H}$, upon premultiplying Eq.(B.13) by $C_H(\boldsymbol{x},\boldsymbol{x}')$ and integrating over $\mathcal{D}^2$ one gets:

$$\forall\, i,j\in\mathbb{N}\;/\;|\boldsymbol{\alpha}_i|,|\boldsymbol{\alpha}_j|\leq p \quad , \quad c_{\boldsymbol{\alpha}_i,\boldsymbol{\alpha}_j} \;=\; \int_{\mathcal{D}}\int_{\mathcal{D}} C_H(\boldsymbol{x},\boldsymbol{x}')\; h_{\boldsymbol{\alpha}_i}(\boldsymbol{x})\; h_{\boldsymbol{\alpha}_j}(\boldsymbol{x})\; d\boldsymbol{x}\; d\boldsymbol{x}' \tag{B.17}$$

The above multidimensional integral may be computed by *quadrature*, say:

$$\begin{aligned} &\forall\, k,l\in\mathbb{N}\;/\;|\boldsymbol{\alpha}_k|,|\boldsymbol{\alpha}_l|\leq p \\ c^n_{\boldsymbol{\alpha}_k,\boldsymbol{\alpha}_l} \;&=\; \sum_{i_1=1}^{n}\cdots\sum_{i_{2d}=1}^{n} w^{(i_1)}\cdots w^{(i_{2d})}\; C_H(\boldsymbol{x}^{(i)},\boldsymbol{x}'^{(i)})\; h_{\boldsymbol{\alpha}_k}(\boldsymbol{x}^{(i)})\; h_{\boldsymbol{\alpha}_l}(\boldsymbol{x}'^{(i)}) \end{aligned} \tag{B.18}$$

where the $\boldsymbol{x}^{(i)}$'s and the $w^{(i)}$'s denote respectively the quadrature *points* and *weights*. A Gauss-Legendre quadrature rule is used in this report. Approximating the autocorrelation function C_H by the truncated series in Eq.(B.13) is equivalent to assuming that C_H is close to a multivariate polynomial of partial degree p over the domain $\mathcal{D}$. Hence the maximal degree of the integrands such as in Eq.(B.17) is $q = 2p$. Then it is necessary to use $n = p$ quadrature points for each dimension of $\mathcal{D}^2$. As a consequence, $N = p^{2d}$ quadrature points have to be used in order to compute all the coefficients $c_{\boldsymbol{\alpha}_i,\boldsymbol{\alpha}_j}$.

The issue of selecting an appropriate value for p is addressed in the next section.

3.3 The issue of truncating the orthogonal series of the autocorrelation function

Upon selecting a given p and computing the coefficients $c_{\boldsymbol{\alpha}_i,\boldsymbol{\alpha}_j}$ according to Eq.(B.18) one gets the following approximation of C_H:

$$\widetilde{C}_{H,p}(\boldsymbol{x},\boldsymbol{x}') \;=\; \sum_{|\boldsymbol{\alpha}|\leq p}\sum_{|\boldsymbol{\beta}|\leq p} c^{(p)}_{\boldsymbol{\alpha},\boldsymbol{\beta}}\; h_{\boldsymbol{\alpha}}(\boldsymbol{x})\; h_{\boldsymbol{\beta}}(\boldsymbol{x}') \tag{B.19}$$

The error of approximation in the $\mathcal{L}_2$-norm is defined by:

$$\left\|C_H \;-\; \widetilde{C}_{H,p}\right\|^2 \;=\; \int_{\mathcal{D}}\int_{\mathcal{D}} \Big(C_H(\boldsymbol{x},\boldsymbol{x}') \;-\; \widetilde{C}_{H,p}(\boldsymbol{x},\boldsymbol{x}')\Big)^2 \; d\boldsymbol{x}\; d\boldsymbol{x}' \tag{B.20}$$

where $\|\cdot\|$ is the norm induced by the inner product $\langle\cdot,\cdot\rangle$, say:

$$\|f\| \;=\; \sqrt{\langle , f\rangle} \tag{B.21}$$

One rather considers the following relative approximation error in the sequel:

$$\varepsilon \;=\; \frac{\left\|C_H \;-\; \widetilde{C}_{H,p}\right\|^2}{\left\|C_H \;-\; \bar{C}_H\right\|^2} \tag{B.22}$$

where $\bar{C_H}$ is the spatial average of the autocorrelation function C_H, that is:

$$\bar{C}_H \;=\; \int_{\mathcal{D}}\int_{\mathcal{D}} C_H(\boldsymbol{x},\boldsymbol{x}')\; d\boldsymbol{x}\; d\boldsymbol{x}' \tag{B.23}$$

The error in Eq.(B.22) may be inexpensively estimated by Monte Carlo simulation as follows:

$$\varepsilon_p \;\approx\; \frac{\sum_{i=1}^{\mathcal{N}} \Big(C_H(\boldsymbol{x}^{(i)},\boldsymbol{x}'^{(i)}) - \widetilde{C}_{H,p}(\boldsymbol{x}^{(i)},\boldsymbol{x}'^{(i)})\Big)^2}{\sum_{i=1}^{\mathcal{N}} \left(C_H(\boldsymbol{x}^{(i)},\boldsymbol{x}'^{(i)}) - \bar{C}_H\right)^2} \tag{B.24}$$

where $\mathcal{N}$ is a large integer, *e.g.* $\mathcal{N} = 10^6$. It is now possible to devise an iterative scheme for selecting an optimal value of p:

1. Initialize $p \leftarrow 0$.

2. Set $p \leftarrow p+1$.

3. Compute the coefficients $c_{\boldsymbol{\alpha}_i,\boldsymbol{\alpha}_j}$ in Eq.(B.13) by quadrature according to Eq.(B.18).

4. Estimate the approximation error ε_p as shown in Eq.(B.24).

5. If ε_p is less than a target value ε_{tgt}, stop. Otherwise go back to Step 2.

Note that it is also possible to employ the iterative procedure outlined in Blatman and Sudret (2008b) in order to obtain a *sparse* spectral representation of C_H, *i.e.* in which only a small number of significant coefficients are retained.

Once the value of p has been selected one solves the standard eigenvalue problem in Eq.(B.16). The Karhunen-Loève approximation of the random field $H(\boldsymbol{x},\omega)$ is thus given by:

$$H(\boldsymbol{x},\omega) \;\approx\; \mu(\boldsymbol{x}) + \sum_{j=0}^{P-1} \left(\sum_{i=1}^{P} d^i_{\boldsymbol{\alpha}_j} \sqrt{\lambda_i}\, \xi_i(\omega) \right) h_{\boldsymbol{\alpha}_j}(\boldsymbol{x}) \tag{B.25}$$

If the random field $H(\boldsymbol{x},\omega)$ is *Gaussian*, it may be shown that $(\xi_i(\omega))_{1\leq i\leq P}$ forms a set of *independent* standard Gaussian random variables. This fully characterizes the decomposition in Eq.(B.25).

4 Comparison with other discretization schemes

4.1 Karhunen-Loève expansion using a Galerkin procedure

A Galerkin-type procedure has been suggested in Ghanem and Spanos (2003) in order to solve the integral eigenvalue problem in Eq.(B.2). It relies upon the choice of a complete basis $\mathcal{B} = (b_i(\boldsymbol{x}))_{i\in\mathbb{N}}$. Note that $\mathcal{B}$ is not necessarily orthonormal as $\mathcal{H}$ defined in Section 2. Similarly to Eq.(B.8) each eigenfunction $\varphi_i(\boldsymbol{x})$ may be expanded onto $\mathcal{B}$ as follows:

$$\forall\, i \in \mathbb{N}^* \quad , \quad \varphi_i(\boldsymbol{x}) \;=\; \sum_{j\in\mathbb{N}} \gamma^i_j\, b_j(\boldsymbol{x}) \tag{B.26}$$

The Galerkin procedure aims at obtaining the best approximation of φ_i when truncating the above series after the P-th term. As detailed in Sudret and Der Kiureghian (2000), this leads to the linear system:

$$\boldsymbol{C}\,\boldsymbol{\Gamma} \;=\; \boldsymbol{B}\,\boldsymbol{\Gamma}\,\boldsymbol{\Lambda} \tag{B.27}$$

where the various matrices are defined as follows:

$$\boldsymbol{C}_{ij} \;=\; \int_{\mathcal{D}}\int_{\mathcal{D}} C_H(\boldsymbol{x},\boldsymbol{x}')\, b_i(\boldsymbol{x})\, b_j(\boldsymbol{x})\, d\boldsymbol{x}\, d\boldsymbol{x}' \tag{B.28}$$

$$\boldsymbol{B}_{ij} \;=\; \int_{\mathcal{D}} b_i(\boldsymbol{x})\; b_j(\boldsymbol{x})\; d\boldsymbol{x} \tag{B.29}$$

$$\boldsymbol{\Gamma}_{ij} \;=\; \gamma_j^i \tag{B.30}$$

$$\boldsymbol{\Lambda}_{ij} \;=\; \delta_{i,j}\; \lambda_j \tag{B.31}$$

The solution scheme may be implemented using finite element-like shape functions as the basis $\mathcal{B}$ (Ghanem and Spanos, 2003). Other complete bases may also be chosen, such as an orthonormal basis. In this case, the matrix $\boldsymbol{B}$ is equal to the unit matrix $\boldsymbol{I}$, and Eq.(B.27) reduces to the standard eigenvalue problem:

$$\boldsymbol{C}\,\boldsymbol{\Gamma} \;=\; \boldsymbol{\Gamma}\,\boldsymbol{\Lambda} \tag{B.32}$$

which is strictly equivalent to Eq.(B.16). Note however that in the present setup the link between the size of the eigenvalue problem and the quality of approximation of the autocorrelation function is not explicit, in contrast to the formulation proposed in Section 2. Thus an adaptive selection of the number of eigenvalues is not straightforward when using a Galerkin formulation.

4.2 Orthogonal series expansion

The *orthogonal series expansion method* (OSE) (Zhang and Ellingwood, 1994) is based on the *a priori* representation of the random field $H(\boldsymbol{x},\omega)$ onto an orthonormal basis $\mathcal{H}$:

$$H(\boldsymbol{x},\omega) \;=\; \mu(\boldsymbol{x}) \;+\; \sum_{i=1}^{+\infty} \chi_i(\omega)\; h_i(\boldsymbol{x}) \tag{B.33}$$

where $\chi_i(\omega)$ are zero-mean random variables. If $H(\boldsymbol{x},\omega)$ is a *Gaussian* random field, it may be shown that $(\chi_i(\omega))_{i\in\mathbb{N}}$ is a sequence of zero-mean Gaussian random variables. Moreover, the correlation matrix $\boldsymbol{\Sigma}_\chi$ of these random variables may be cast as:

$$\forall\; i,j \in \mathbb{N}^* \quad , \quad (\boldsymbol{\Sigma}_\chi)_{ij} \;=\; \int_{\mathcal{D}}\int_{\mathcal{D}} C_H(\boldsymbol{x},\boldsymbol{x}')\; h_i(\boldsymbol{x})\; h_j(\boldsymbol{x}')\; d\boldsymbol{x}\; d\boldsymbol{x}' \tag{B.34}$$

which corresponds to the integrals denoted by $c_{\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2}$ in Eq.(B.18). As a consequence the correlation matrix $\boldsymbol{\Sigma}_\chi$ is equal to $\boldsymbol{C}$ in Eq.(B.9), and the number of terms P to be retained in Eq.(B.33) may be selected by monitoring the accuracy of the approximation of $C_H(\boldsymbol{x},\boldsymbol{x}')$ onto the truncated basis $\mathcal{H}_P \otimes \mathcal{H}_P$, as shown in Section 3.3.

The OSE method relies upon the transformation of the correlated Gaussian random variables $\boldsymbol{\chi} = (\chi_i)_{1\leq i\leq P}$ into an uncorrelated standard normal random vector $\boldsymbol{\xi}$ by means of a spectral decomposition of the covariance matrix $\boldsymbol{\Sigma}_\chi \equiv \boldsymbol{C}$:

$$\boldsymbol{C}\,\boldsymbol{D} \;=\; \boldsymbol{D}\,\boldsymbol{\Lambda} \tag{B.35}$$

which is nothing but the eigenvalue problem in Eq.(B.9). Truncating the series in Eq.(B.33) after P terms, this leads to the Karhunen-Loève approximation (B.25) of the random field $H(\boldsymbol{x},\omega)$.

5 Conclusion

A formulation of the Karhunen-Loève decomposition based on a spectral representation of the autocorrelation function has been proposed in the present report. It leads to a standard eigenvalue problem whose solving is straightforward. It appears that this spectral scheme is equivalent to the common Galerkin-based Karhunen-Loève decomposition when expanding the autocorrelation eigenfunctions onto a complete orthonormal basis. Furthermore, the proposed strategy is equivalent to the OSE method if the random field under consideration is Gaussian. The difference is that the approach proposed herein consists in expanding the covariance kernel $C(\boldsymbol{x}, \boldsymbol{x}')$, which is known, instead of the unknown eigenfunctions $\varphi_i(\boldsymbol{x})$ onto a suitable basis. Note that this scheme may provide a suitable framework for selecting adaptively the number of retained modes, *i.e.* the size of the eigenvalue problem to solve.

Appendix C

Efficient generation of index sets

1 Generation of a low rank index set

This section is focused on the generation of the following *low rank* index set:

$$\mathcal{A}^{M,p,j} = \{\boldsymbol{\alpha} \in \mathbb{N}^M \,:\, \|\boldsymbol{\alpha}\|_1 = p \,,\; \|\boldsymbol{\alpha}\|_0 = j\} \tag{C.1}$$

where:

$$\|\boldsymbol{\alpha}\|_1 \;\equiv\; \sum_{i=1}^{M} \alpha_i \qquad , \qquad \|\boldsymbol{\alpha}\|_0 \;\equiv\; \sum_{i=1}^{M} \mathbf{1}_{\{\alpha_i > 0\}} \tag{C.2}$$

A two-step strategy is proposed in this purpose.

First of all, one generates all integer j-tuples $\{\alpha_1, \cdots, \alpha_j\}$ such that $\alpha_1 \geq \cdots \geq \alpha_j \geq 1$ and $\alpha_1 + \cdots + \alpha_j = p$, assuming that $p \geq j \geq 2$. This is achieved using the so-called *Algorithm H* described in Knuth (2005a) and outlined below:

1. **Initialize.** Set $\alpha_1 \leftarrow p - j + 1$ and $\alpha_k \leftarrow 1$ for $1 < k \leq j$. Also set $\alpha_{j+1} \leftarrow -1$.

2. **Visit.** Visit the tuple $\alpha_1 \cdots \alpha_j$. Then go to Step 4 if $\alpha_2 \geq \alpha_1 - 1$.

3. **Tweak α_1 and α_2.** Set $\alpha_1 \leftarrow \alpha_1 - 1$, $\alpha_2 \leftarrow \alpha_2 + 1$, and return to Step 2.

4. **Find k.** Set $k \leftarrow 3$ and $s \leftarrow \alpha_1 + \alpha_2 - 1$. Then, if $\alpha_k \geq \alpha_1 - 1$, set $s \leftarrow s + \alpha_k$, $k \leftarrow k + 1$, and repeat until $\alpha_k < \alpha_1 - 1$. (Now $s = \alpha_1 + \cdots + \alpha_{k-1} - 1$.)

5. **Increase α_k.** Terminate if $k > j$. Otherwise set $x \leftarrow \alpha_k + 1$, $\alpha_k \leftarrow x$, $k \leftarrow k - 1$.

6. **Tweak $\alpha_1 \cdots \alpha_k$.** While $k > 1$, set $\alpha_k \leftarrow x$, $s \leftarrow s - x$ and $k \leftarrow k - 1$. Finally set $\alpha_1 \leftarrow s$ and return to Step 2.

For example, if $p = 4$ and $j = 2$ the procedure provides the couples $\{3, 1\}$ and $\{2, 2\}$.

It is then necessary to complement each couple with zeros on its left in order to obtain multi-indices with length M. For instance, if $M = 3$ one gets:

$$\begin{bmatrix} 0 & 3 & 1 \\ 0 & 2 & 2 \end{bmatrix} \tag{C.3}$$

In a second time, all the permutations of the sorted multi-indices are generated using the so-called *Algorithm L* (Knuth, 2005b), described below:

1. **Visit.** Visit the permutation $\alpha_1 \cdots \alpha_M$.

2. **Find** k**.** Set $k \leftarrow M - 1$. If $\alpha_k \geq \alpha_{k+1}$, decrease k by 1 repeatedly until $\alpha_k < \alpha_{k+1}$. Terminate the algorithm if $k = 0$.

3. **Increase** α_k**.** Set $l \leftarrow M$. If $\alpha_k \geq \alpha_l$, decrease l by 1 repeatedly until $\alpha_k < \alpha_l$. Then interchange $\alpha_k \leftrightarrow \alpha_l$.

4. **Reverse** $\alpha_{k+1} \cdots \alpha_M$**.** Set $r \leftarrow k + 1$ and $l \leftarrow M$. Then, if $r < l$, interchange $\alpha_r \leftrightarrow \alpha_l$, set $r \leftarrow r + 1$, $l \leftarrow l - 1$, and repeat until $r \geq l$. Return to Step 1.

In order to apply this algorithm to our problem, the elements of each M-tuple are first sorted as follows:

$$\begin{bmatrix} 0 & 1 & 3 \\ 0 & 2 & 2 \end{bmatrix} \tag{C.4}$$

The algorithm L thus yields the two following lists of multi-indices:

$$\begin{bmatrix} 0 & 2 & 2 \\ 2 & 0 & 2 \\ 2 & 2 & 0 \end{bmatrix} \tag{C.5}$$

and

$$\begin{bmatrix} 0 & 1 & 3 \\ 0 & 3 & 1 \\ 1 & 0 & 3 \\ 1 & 3 & 0 \\ 3 & 0 & 1 \\ 3 & 1 & 0 \end{bmatrix} \tag{C.6}$$

Thus all the elements of the set $\mathcal{A}^{M,p,j}$ are generated by concatenating the obtained lists of permutations.

2 Generation of the full polynomial chaos basis

The goal is now to compute all the multi-indices $\boldsymbol{\alpha}$ in the set $\mathcal{A}^{M,p}$. This is achieved by successively generating the subsets $\mathcal{A}^{M,k,j(k)}$, $k = 1, \ldots, p$ and $j(k) = 1, \ldots, \min(M, p)$ as shown in Section 1. A comparison between the computer processing time (CPT) required by the proposed strategy and the *ball sampling*-based scheme in Sudret and Der Kiureghian (2000) is carried out. Computations are performed on a Xeon(TM) CPU 2.80 GHz. The results are reported in Table C.1.

Table C.1: *Computer processing time required by the ball sampling scheme (method 1) and the alternative strategy devised herein (method 2)*

M	p	P	CPT 1 (")	CPT 2 (")
	3	286	0.0	0.0
10	4	1,001	0.1	0.0
	5	3,003	0.4	0.0
	6	8,008	6.9	0.2
	3	816	0.2	0.0
15	4	3,876	0.9	0.1
	5	15,504	86.9	3.0
	6	54,264	1407.3	75.3
	3	1,140	0.2	0.0
17	4	5,985	9.3	0.3
	5	26,334	357.8	21.2

When using the ball sampling scheme, the CPT increases extremely fast with both the number of input parameters and the degree of the PC expansion. The proposed strategy allows one to considerably moderate this algorithmic complexity.

3 Generation of an hyperbolic index set

Let us consider the following *hyperbolic* index set:

$$\mathcal{A}_q^{M,p} \equiv \{\boldsymbol{\alpha} \in \mathbb{N}^M : \|\boldsymbol{\alpha}\|_q \leq p\} \tag{C.7}$$

where

$$\|\boldsymbol{\alpha}\|_q = \left(\sum_{i=1}^{M} \alpha_i^q \right)^{1/q} \tag{C.8}$$

The index sets such as in Eq.(C.7) may be easily generated by slightly modifying the proposed method. Let us first define the index sets $\mathcal{A}_q^{M,p,j}$ associated to specific total degree k and interaction order j:

$$\mathcal{A}_q^{M,p,k,j} = \{\boldsymbol{\alpha} \in \mathbb{N}^M \, : \, \|\boldsymbol{\alpha}\|_1 = k \, , \, \|\boldsymbol{\alpha}\|_0 = j \, , \, \|\boldsymbol{\alpha}\|_q \leq p\} \tag{C.9}$$

These sets may be generated as follows:

- Apply Algorithm H to generate all integer j-tuples with total degree k.
- Complement each multi-index by adding $M - j$ zeros on its left. One gets a table with M columns.
- Remove from the table all those multi-indices whose q-norm is strictly greater than p.
- Apply Algorithm L to the table.

For the sake of illustration, let us consider the case $M = 3$, $p = 5$, $k = 4$ and $j = 2$. One selects the norm $q = 0.75$. By applying Algorithm H one gets the multi-indices $\{3, 1\}$ and $\{2, 2\}$ (Step 1). One complements those multi-indices with $M - j = 1$ zero on their left (Step 2), hence the table:

$$\begin{bmatrix} 0 & 3 & 1 \\ 0 & 2 & 2 \end{bmatrix} \tag{C.10}$$

The q-norm of each row of the table is computed (Step 3). The results are 4.87 and 5.04 for the first and second row, respectively. As $5.04 > p = 5$, the second row is deleted and the table reduces to:

$$\begin{bmatrix} 0 & 3 & 1 \end{bmatrix} \tag{C.11}$$

Algorithm L is now applied in order to generate all the permutations (Step 4):

$$\begin{bmatrix} 3 & 1 & 0 \\ 3 & 0 & 1 \\ 1 & 3 & 0 \\ 1 & 0 & 3 \\ 0 & 3 & 1 \\ 0 & 1 & 3 \end{bmatrix} \tag{C.12}$$

The whole set $\mathcal{A}_q^{M,p}$ is then obtained by repeating the procedure for all $k = 1, \ldots, p$ and $j = 1, \ldots, \min(k, M)$, since:

$$\mathcal{A}_q^{M,p} = \bigcup_{k=1,\ldots,p} \; \bigcup_{j=1,\ldots,\min(k,M)} \mathcal{A}_q^{M,p,k,j} \tag{C.13}$$

4 Conclusion

A method is proposed in order to generate efficiently the basis of a truncated polynomial chaos approximation. It relies upon two algorithms outlined in Knuth (2005a,b). It clearly overperforms the usual generation scheme based on a ball sampling scheme in terms of computational cost, especially when the total degree of the polynomial chaos and the number of input parameters of the model under consideration get large. The proposed strategy may be easily applied to polynomial chaos approximations which are truncated according to the sparsity-of-effect principle, *i.e.* by favoring the main effects and the simple interaction terms.

Appendix D

Leave-one-out cross validation

Let us consider the following polynomial chaos (PC) expansion of order p of the model response:

$$\widehat{\mathcal{M}}(\boldsymbol{X}) = \sum_{|\boldsymbol{\alpha}|\leq p} a_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{D.1}$$

Upon denoting by P the number of terms in the truncated series and introducing a reordering of the multi-indices $\boldsymbol{\alpha}$, one gets:

$$\widehat{\mathcal{M}}(\boldsymbol{X}) = \sum_{j=1}^{P} a_{\boldsymbol{\alpha}_j} \psi_{\boldsymbol{\alpha}_j}(\boldsymbol{X}) \tag{D.2}$$

Let us introduce the vector notation:

$$\boldsymbol{\psi}(\boldsymbol{X}) \;=\; (\psi_{\boldsymbol{\alpha}_1}(\boldsymbol{X}) \;\cdots\; \psi_{\boldsymbol{\alpha}_P}(\boldsymbol{X}))^{\mathsf{T}} \tag{D.3}$$

$$\boldsymbol{a} \;=\; (a_{\boldsymbol{\alpha}_1} \;\cdots\; a_{\boldsymbol{\alpha}_P})^{\mathsf{T}} \tag{D.4}$$

The PC representation in Eq.(D.2) rewrites:

$$\widehat{\mathcal{M}}(\boldsymbol{X}) \;=\; \boldsymbol{\psi}(\boldsymbol{X})^{\mathsf{T}}\, \boldsymbol{a} \tag{D.5}$$

Let $\mathcal{X} = \{\boldsymbol{x}^{(1)}, ..., \boldsymbol{x}^{(N)}\}^{\mathsf{T}}$ be an experimental design. Let $\mathcal{M}_{\mathcal{X}\setminus i}$ be the PC expansion whose coefficients $\widehat{\boldsymbol{a}}$ have been computed by regression from the experimental design $\mathcal{X}\setminus\{\boldsymbol{x}^{(i)}\} \equiv \mathcal{X}\setminus i$, *i.e.* when removing the i-th observation from the training set $\mathcal{X}$. The *predicted residual* is defined as the difference between the model evaluation at $\boldsymbol{x}^{(i)}$ and its prediction based on $\mathcal{M}_{\mathcal{X}\setminus i}$:

$$\Delta^{(i)} \;=\; \mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}_{\mathcal{X}\setminus i}(\boldsymbol{x}^{(i)}) \tag{D.6}$$

The generalization error is then estimated by the mean *predicted residual sum of squares* (PRESS), *i.e.* the following empirical mean-square predicted residual:

$$I^{*}_{\mathcal{X}}[\mathcal{M}_{\mathcal{X}}] \;=\; \frac{1}{N}\sum_{i=1}^{N} \left(\Delta^{(i)}\right)^2 \tag{D.7}$$

Let us denote by $\boldsymbol{\Psi}_i$ the *design matrix* related to the experimental design $\mathcal{X} \setminus i$:

$$\boldsymbol{\Psi}_i \;=\; \begin{pmatrix} \psi_{\boldsymbol{\alpha}_1}(\boldsymbol{x}^{(1)}) & \cdots & \psi_{\boldsymbol{\alpha}_P}(\boldsymbol{x}^{(1)}) \\ \vdots & \ddots & \vdots \\ \psi_{\boldsymbol{\alpha}_1}(\boldsymbol{x}^{(i-1)}) & \cdots & \psi_{\boldsymbol{\alpha}_P}(\boldsymbol{x}^{(i-1)}) \\ \psi_{\boldsymbol{\alpha}_1}(\boldsymbol{x}^{(i+1)}) & \cdots & \psi_{\boldsymbol{\alpha}_P}(\boldsymbol{x}^{(i+1)}) \\ \vdots & \ddots & \vdots \\ \psi_{\boldsymbol{\alpha}_1}(\boldsymbol{x}^{(N)}) & \cdots & \psi_{\boldsymbol{\alpha}_P}(\boldsymbol{x}^{(N)}) \end{pmatrix} \tag{D.8}$$

and by $\boldsymbol{M}_i = \boldsymbol{\Psi}_i^\mathsf{T}\boldsymbol{\Psi}_i$ the corresponding *information matrix*. Let $\mathcal{Y}_i$ be the associated response vector.

The coefficients of the PC expansion $\mathcal{M}_{\mathcal{X}\setminus i}$ thus read:

$$\widehat{\boldsymbol{a}_i} \;=\; \boldsymbol{M}_i^{-1}\boldsymbol{\Psi}_i^\mathsf{T}\mathcal{Y}_i \tag{D.9}$$

Moreover the prediction of the metamodel $\mathcal{M}_{\mathcal{X}\setminus i}$ at the design point $\boldsymbol{x}^{(i)}$ is given by:

$$\mathcal{M}_{\mathcal{X}\setminus i}(\boldsymbol{x}_i) \;=\; \boldsymbol{\psi}_i^\mathsf{T}\widehat{\boldsymbol{a}_i} \tag{D.10}$$

where

$$\boldsymbol{\psi}_i \;=\; \left(\psi_{\boldsymbol{\alpha}_1}(\boldsymbol{x}^{(i)}) \;\cdots\; \psi_{\boldsymbol{\alpha}_P}(\boldsymbol{x}^{(i)})\right)^\mathsf{T} \tag{D.11}$$

Hence the predicted residual:

$$\Delta^{(i)} \;=\; \mathcal{M}(\boldsymbol{x}^{(i)}) - \boldsymbol{\psi}_i^\mathsf{T}\widehat{\boldsymbol{a}_i} \tag{D.12}$$

which rewrites by using Eq.(D.9):

$$\Delta^{(i)} \;=\; \mathcal{M}(\boldsymbol{x}^{(i)}) - \boldsymbol{\psi}_i^\mathsf{T}\boldsymbol{M}_i^{-1}\boldsymbol{\Psi}_i^\mathsf{T}\mathcal{Y}_i \tag{D.13}$$

Let us now consider the vector $\boldsymbol{\Psi}^\mathsf{T}\mathcal{Y}$:

$$\boldsymbol{\Psi}^\mathsf{T}\mathcal{Y} \;=\; \begin{pmatrix} \psi_{\boldsymbol{\alpha}_1}(\boldsymbol{x}^{(1)}) & \cdots & \psi_{\boldsymbol{\alpha}_1}(\boldsymbol{x}^{(N)}) \\ \vdots & \ddots & \vdots \\ \psi_{\boldsymbol{\alpha}_P}(\boldsymbol{x}^{(1)}) & \cdots & \psi_{\boldsymbol{\alpha}_P}(\boldsymbol{x}^{(N)}) \end{pmatrix} \begin{pmatrix} \mathcal{M}(\boldsymbol{x}^{(1)}) \\ \vdots \\ \mathcal{M}(\boldsymbol{x}^{(N)}) \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{N} \psi_1(\boldsymbol{x}^{(i)})\mathcal{M}(\boldsymbol{x}^{(i)}) \\ \vdots \\ \sum_{i=1}^{N} \psi_P(\boldsymbol{x}^{(i)})\mathcal{M}(\boldsymbol{x}^{(i)}) \end{pmatrix} \tag{D.14}$$

Consequently, as $\boldsymbol{\Psi}_i$ and $\mathcal{Y}_i$ are both obtained by removing the i-th rows of their counterparts $\boldsymbol{\Psi}$ and $\mathcal{Y}$ for the full experimental design, one gets:

$$\boldsymbol{\Psi}_i^\mathsf{T}\mathcal{Y}_i \;=\; \begin{pmatrix} \sum_{j\in\{1,\ldots,N\}\setminus\{i\}} \psi_1(\boldsymbol{x}^{(j)})\mathcal{M}(\boldsymbol{x}^{(j)}) \\ \vdots \\ \sum_{j\in\{1,\ldots,N\}\setminus\{i\}} \psi_P(\boldsymbol{x}^{(j)})\mathcal{M}(\boldsymbol{x}^{(j)}) \end{pmatrix} \;=\; \boldsymbol{\Psi}^\mathsf{T}\mathcal{Y} - \mathcal{M}(\boldsymbol{x}^{(i)})\boldsymbol{\psi}_i \tag{D.15}$$

On the other hand, it can be shown that $\boldsymbol{M}_i^{-1}$ is related to its counterpart $\boldsymbol{M}^{-1}$ as follows:

$$\boldsymbol{M}_i^{-1} \;=\; \boldsymbol{M}^{-1} \;+\; \frac{\boldsymbol{M}^{-1}\boldsymbol{\psi}_i\boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1}}{1-\boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1}\boldsymbol{\psi}_i} \tag{D.16}$$

which leads to:

$$\begin{aligned}\boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}_i^{-1} &= \frac{\boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1} \;-\; \boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1}\boldsymbol{\psi}_i\boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1} \;+\; \boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1}\boldsymbol{\psi}_i\boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1}}{1-\boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1}\boldsymbol{\psi}_i} \\ &= \frac{\boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1}}{1-\boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1}\boldsymbol{\psi}_i}\end{aligned} \tag{D.17}$$

By substituting for Eqs.(D.15),(D.17) into the predicted residual (D.13), one gets:

$$\Delta^{(i)} \;=\; \frac{\mathcal{M}(\boldsymbol{x}^{(i)}) - \boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1}\boldsymbol{\Psi}\mathcal{Y}}{1-\boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1}\boldsymbol{\psi}_i} \tag{D.18}$$

The numerator in the previous equation rewrites:

$$\begin{aligned}\mathcal{M}(\boldsymbol{x}^{(i)}) \;-\; \boldsymbol{\psi}_i^{\mathsf{T}}\boldsymbol{M}^{-1}\boldsymbol{\Psi}\mathcal{Y} &= \mathcal{M}(\boldsymbol{x}^{(i)}) \;-\; \boldsymbol{\psi}_i^{\mathsf{T}}\widehat{\boldsymbol{a}} \\ &= \mathcal{M}(\boldsymbol{x}^{(i)}) \;-\; \widehat{\mathcal{M}}(\boldsymbol{x}^{(i)})\end{aligned} \tag{D.19}$$

where $\widehat{\boldsymbol{a}}$ is the vector of the PC coefficients that have been computed by regression from the experimental design $\mathcal{X}$. The first line holds because of the expression of the PC coefficients obtained by regression. The second line is justified by Eq.(D.5).

Let us now define the *projection matrix* by:

$$\boldsymbol{P} \;=\; \boldsymbol{I}_N \;-\; \boldsymbol{\Psi}\boldsymbol{M}^{-1}\boldsymbol{\Psi}^{\mathsf{T}} \tag{D.20}$$

where $\boldsymbol{I}_N$ denotes the N-dimensional unity matrix. It can be noticed that the denominator of the fraction in Eq.(D.13) is the i-th component of the diagonal of $\boldsymbol{P}$. Hence the predicted residual (D.18):

$$\Delta^{(i)} \;=\; \frac{\mathcal{M}(\boldsymbol{x}^{(i)}) \;-\; \widehat{\mathcal{M}}(\boldsymbol{x}^{(i)})}{h_i} \tag{D.21}$$

where h_i denotes the i-th diagonal term of $\boldsymbol{P}$. As the PRESS statistic is defined as the mean-square prediction error, it can thus be recast as:

$$I^*_{\mathcal{X}}[\mathcal{M}_{\mathcal{X}}] \;=\; \frac{1}{N}\sum_{i=1}^{N}\left(\frac{\mathcal{M}(\boldsymbol{x}^{(i)}) - \widehat{\mathcal{M}}(\boldsymbol{x}^{(i)})}{h_i}\right)^2 \tag{D.22}$$

Appendix E

Computation of the LAR descent direction and step

Let us consider the LAR algorithm at step $k > 1$. Let us denote by $\mathcal{A}^{(k)}$ the multi-index set corresponding to the k predictors that have entered the metamodel, and by $\mathcal{A}_c^{(k)} \equiv \mathcal{A} \setminus \mathcal{A}^{(k)}$ its complementary. Let us denote by $\mathcal{Y}^{(k)}$ the predictions based on the current metamodel at the points of the experimental design $\mathcal{X}$. Lastly, let us denote by $\hat{\boldsymbol{a}}^{(k)}$ the current estimates of the *active* PC coefficients.

One must include the predictor $\boldsymbol{\psi}_{\boldsymbol{\alpha}_i}$ most correlated with the current residual $\mathcal{Y} - \mathcal{Y}^{(k)}$, *i.e.* such that:

$$\boldsymbol{\alpha}_i \;=\; \arg \max_{\boldsymbol{\alpha} \in \mathcal{A}_c^{(k)}} \left| \boldsymbol{\Psi}_{\boldsymbol{\alpha}_i}^{\mathsf{T}} \left(\mathcal{Y} - \mathcal{Y}^{(k)} \right) \right| \tag{E.1}$$

The active multi-index set is updated as follows: $\mathcal{A}^{(k+1)} = \mathcal{A}^{(k)} \cup \{\boldsymbol{\alpha}_i\}$. Accordingly the values of the active coefficients are updated as follows: $\hat{\boldsymbol{a}}^{(k+1)} = \hat{\boldsymbol{a}}^{(k)} + \gamma^{(k)} \tilde{\boldsymbol{w}}^{(k)}$. $\tilde{\boldsymbol{w}}^{(k)}$ and $\gamma^{(k)}$ are referred to as the *descent direction* and *step*, respectively.

The descent direction is defined by the joint least-square coefficients of the active predictors on the current residual, that is:

$$\boldsymbol{w}^{(k)} \;\equiv\; \left(\boldsymbol{\Psi}_{\mathcal{A}^{(k)}}^{\mathsf{T}} \boldsymbol{\Psi}_{\mathcal{A}^{(k)}} \right)^{-1} \; \boldsymbol{\Psi}_{\mathcal{A}^{(k)}}^{\mathsf{T}} \; \left(\mathcal{Y} - \mathcal{Y}^{(k)} \right) \tag{E.2}$$

Now, as the active predictors are equally correlated with the current residual by construction of the LAR algorithm, the vector of empirical correlations satisfies:

$$\exists\, c > 0 \;,\; \boldsymbol{\Psi}_{\mathcal{A}^{(k)}}^{\mathsf{T}} \; \left(\mathcal{Y} - \mathcal{Y}^{(k)} \right) \;=\; c\, \boldsymbol{s} \tag{E.3}$$

where $\boldsymbol{s}$ is the vector of length $\mathrm{card}(\mathcal{A}^{(k)})$ that contains the correlation signs. Hence the descent direction rewrites:

$$\boldsymbol{w}^{(k)} \;\equiv\; \left(\boldsymbol{\Psi}_{\mathcal{A}^{(k)}}^{\mathsf{T}} \boldsymbol{\Psi}_{\mathcal{A}^{(k)}} \right)^{-1} \; c\, \boldsymbol{s} \tag{E.4}$$

One selects c in order to obtain a unit descent direction as follows:

$$c \;=\; \left[\boldsymbol{s}^{\mathsf{T}}\left(\boldsymbol{\Psi}_{\mathcal{A}^{(k)}}^{\mathsf{T}}\boldsymbol{\Psi}_{\mathcal{A}^{(k)}}\right)^{-1}\boldsymbol{s}\right]^{-1/2} \tag{E.5}$$

Using notation $\boldsymbol{u}^{(k)} \equiv \boldsymbol{\Psi}_{\mathcal{A}^{(k)}}\boldsymbol{w}^{(k)}$, the future vector of predictions may be cast as $\mathcal{Y}^{(k+1)} = \mathcal{Y}^{(k)} + \gamma^{(k)}\boldsymbol{u}^{(k)}$. The vector of correlation coefficients after adding the predictor $\boldsymbol{\psi}_i$ satisfies:

$$\begin{aligned}\boldsymbol{\Psi}_{\mathcal{A}^{(k)}}^{\mathsf{T}}\left(\mathcal{Y}-\mathcal{Y}^{(k+1)}\right) &= \boldsymbol{\Psi}_{\mathcal{A}^{(k)}}^{\mathsf{T}}\left(\mathcal{Y} - \mathcal{Y}^{(k)} - \gamma^{(k)}\boldsymbol{u}^{(k)}\right)\\ &= \boldsymbol{\Psi}_{\mathcal{A}^{(k)}}^{\mathsf{T}}\left(\mathcal{Y} - \mathcal{Y}^{(k)}\right) - \gamma^{(k)}\,\boldsymbol{\Psi}_{\mathcal{A}^{(k)}}^{\mathsf{T}}\,\boldsymbol{u}^{(k)}\end{aligned} \tag{E.6}$$

Using Eq.(E.3) this reduces to:

$$\boldsymbol{\Psi}_{\mathcal{A}^{(k)}}^{\mathsf{T}}\left(\mathcal{Y}-\mathcal{Y}^{(k+1)}\right) \;=\; \left(c - \gamma^{(k)}\right)\,\boldsymbol{s} \tag{E.7}$$

In other words each absolute correlation coefficient is equal to $c - \gamma^{(k)}$.

As the new predictor $\boldsymbol{\psi}_{\boldsymbol{\alpha}_i}$ will be the most correlated with the new residual, one gets:

$$\begin{aligned}c - \gamma^{(k)} &= \max_{\boldsymbol{\alpha}_i \in \mathcal{A}_c^{(k)}} \left|\boldsymbol{\psi}_{\boldsymbol{\alpha}_i}^{\mathsf{T}}\left(\mathcal{Y} - \mathcal{Y}^{(k+1)}\right)\right|\\ &= \max_{\boldsymbol{\alpha}_i \in \mathcal{A}_c^{(k)}} |\underbrace{\boldsymbol{\psi}_{\boldsymbol{\alpha}_i}^{\mathsf{T}}\left(\mathcal{Y} - \mathcal{Y}^{(k)}\right)}_{\equiv\, c_i} - \gamma^{(k)}\,\underbrace{\boldsymbol{\psi}_{\boldsymbol{\alpha}_i}^{\mathsf{T}}\,\boldsymbol{u}^{(k)}}_{\equiv\, d_i}|\end{aligned} \tag{E.8}$$

As a result the LAR descent step reads:

$$\gamma^{(k)} \;=\; \min_{\boldsymbol{\alpha}_i \in \mathcal{A}_c^{(k)}}{}^{+}\left(\frac{c - c_i}{1 - d_i}\,,\;\frac{c + c_i}{1 + d_i}\right) \tag{E.9}$$

where $\min^{+}$ indicates that the minimum is taken over only positive components within each choice of $\boldsymbol{\alpha}_i$. $\gamma^{(k)}$ is the smallest positive value such that a new predictor $\boldsymbol{\psi}_{\boldsymbol{\alpha}_i}$ enters the metamodel.

Bibliography

Abramowitz, M. and I. Stegun (1970). *Handbook of mathematical functions*. Dover Publications, Inc.

AFCEN (1997, January). *In service assessment code for mechanical components of PWR nuclear islands, RSE-M Code*. Paris.

Allen, D. (1971). The prediction sum of squares as a criterion for selecting prediction variables. Technical Report 23, Dept. of Statistics, University of Kentucky.

An, J. and A. Owen (2001). Quasi-regression. *J. Complexity 17*(4), 588–607.

Antoniadis, A. (1997). Wavelets in statistics: a review. *Journal of the Italian Statistical Association 6*, 97–144.

Antoniadis, A., J. Bigot, and T. Sapatinas (2001). Wavelet estimators in nonparametric regression: a comparative simulation study. *J. Stat. Software 6*(6), 1–83.

Archer, G., A. Saltelli, and I. Sobol' (1997). Sensitivity measures, ANOVA-like techniques and the use of bootstrap. *J. Stat. Comput. Simul. 58*, 99–120.

Au, S. and J. Beck (2001). Estimation of small failure probabilities in high dimensions by subset simulation. *Prob. Eng. Mech. 16*(4), 263–277.

Babuška, I. and P. Chatzipantelidis (2002). On solving elliptic stochastic partial differential equations. *Comput. Methods Appl. Mech. Engrg. 191*(37-38), 4093–4122.

Babuška, I., R. Tempone, and G. Zouraris (2005). Solving elliptic boundary value problems with uncertain coefficients by the finite element method: the stochastic formulation. *Comput. Methods Appl. Mech. Engrg. 194*, 1251–1294.

Babuška, I., F. Nobile, and R. Tempone (2007). A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM J. Num. Anal. 45*(3), 1005–1034.

Barthelmann, V., E. Novak, and K. Ritter (2000). High dimensional polynomial interpolation on sparse grids. *Adv. Comput. Math. 12*, 273–288.

Beres, D. and D. Hawkins (2001). Plackett-Burman techniques for sensitivity analysis of many-parametered models. *Ecol. Modell. 141*(1-3), 171–183.

Berveiller, M. (2005). *Eléments finis stochastiques : approches intrusive et non intrusive pour des analyses de fiabilité*. Ph. D. thesis, Université Blaise Pascal, Clermont-Ferrand.

Berveiller, M., B. Sudret, and M. Lemaire (2006). Stochastic finite elements: a non intrusive approach by regression. *Eur. J. Comput. Mech. 15*(1-3), 81–92.

Bieri, M. and C. Schwab (2009). Sparse high order FEM for elliptic sPDEs. *Comput. Methods Appl. Mech. Engrg 198*, 1149–1170.

Bjorck, Å. (1996). *Numerical methods for least squares problems*. SIAM Press, Philadelphia, PA.

Blatman, G. and B. Sudret (2008a). Adaptive sparse polynomial chaos expansions - Application to structural reliability. In *Proc. 4th Int. ASRANet Colloquium, Athenes, Greece*.

Blatman, G. and B. Sudret (2008b). Adaptive sparse polynomial chaos expansions using a regression approach. Technical Report H-T26-2008-00668-EN, EDF R&D.

Blatman, G. and B. Sudret (2008c). Développements par chaos polynomiaux creux et adaptatifs - Application à l'analyse de fiabilité. In *Proc. JN Fiab'08, Journées Nationales Fiabilité des Matériaux et des Structures, Nantes, France*.

Blatman, G. and B. Sudret (2008d). Sparse polynomial chaos expansions and adaptive stochastic finite elements using a regression approach. *Comptes Rendus Mécanique 336*(6), 518–523.

Blatman, G. and B. Sudret (2009a). Anisotropic parcimonious polynomial chaos expansions based on the sparsity-of-effects principle. In *Proc. ICOSSAR'09, Int Conf. on Structural Safety And Reliability, Osaka, Japan*.

Blatman, G. and B. Sudret (2009b). Efficient computation of Sobol' sensitivity indices using sparse polynomial chaos expansions. *Reliab. Eng. Sys. Safety*. submitted for publication.

Blatman, G. and B. Sudret (2009c). Eléments finis stochastiques non intrusifs à partir de développements par chaos polynomiaux creux et adaptatifs. In *Proc. 9ème Colloque National en Calcul de Structures, Giens, France*.

Blatman, G. and B. Sudret (2009d). Use of sparse polynomial chaos expansions in adaptive stochastic finite element analysis. *Prob. Eng. Mech.*. accepted for publication.

Blatman, G., B. Sudret, and M. Berveiller (2007). Quasi-random numbers in stochastic finite element analysis. *Mécanique & Industries 8*, 289–297.

Boyd, J. (1989). *Chebyshev & Fourier spectral methods*. Springer Verlag.

Breitung, K. (1984). Asymptotic approximation for multinormal integrals. *J. Eng. Mech. 110*(3), 357–366.

Brutman, L. (1978). On the Lebesgue function for polynomial interpolation. *SIAM J. Numer. Anal. 15*(4), 694–704.

Caflisch, R., W. Morokoff, and A. Owen (1997). Valuation of mortgage backed securities using Brownian bridges to reduce effective dimension. *J. Comput. Finance 1*, 27–46.

Candes, E. and T. Tao (2007). The Dantzig selector: statistical estimation when p is much larger than n. *Ann. Statist. 35*, 2313–2351.

Castro, R., R. Willett, and R. Nowak (2005). Faster rates in regression via active learning. In *Proceedings of NIPS*.

Chapelle, O., V. Vapnik, and Y. Bengio (2002). Model selection for small sample regression. *Machine Learning 48*(1), 9–23.

Chen, C., K.-L. Tsui, R. Barton, and M. Meckesheimer (2006). A review on design, modeling, and applications of computer experiments. *IIE Transactions 38*, 273–291.

Ching, J., J. Beck, and S. Au (2005). Hybrid subset simulation method for reliability estimation of dynamical systems subject to stochastic excitation. *Prob. Eng. Mech. 20*(3), 199–214.

Choi, S., R. Grandhi, R. Canfield, and C. Pettit (2004). Polynomial chaos expansion with Latin Hypercube sampling for estimating response variability. *AIAA Journal 45*, 1191–1198.

Chung, D., M. Gutiérrez, and R. de Borst (2005). Object-oriented stochastic finite element analysis of fibre metal laminates. *Comput. Methods Appl. Mech. Engrg. 194*, 1427–1446.

Clarke, S., J. Griebsch, and T. Simpson (2003). Analysis of support vector regression for approximation of complex engineering analyses. In *Proc. DETC'03, Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Chicago*.

Cukier, H., R. Levine, and K. Shuler (1978). Nonlinear sensitivity analysis of multiparameter model systems. *J. Comput. Phys. 26*, 1–42.

De Rocquigny, E. (2006a). La maîtrise des incertitudes dans un contexte industriel : 1^e partie – Une approche méthodologique globale basée sur des exemples. *J. Soc. Française Stat. 147*(3), 33–71.

De Rocquigny, E. (2006b). La maîtrise des incertitudes dans un contexte industriel : 2^e partie – Revue des méthodes de modélisation statistique, physique et numérique. *J. Soc. Française Stat. 147*(3), 73–106.

De Rocquigny, E., N. Devictor, and S. Tarantola (2008). *Uncertainty in industrial practice: a guide to quantitative uncertainty management.* Lavoisier, France.

Deb, M., I. Babuška, and J. Oden (2001). Solution of stochastic partial differential equations using Galerkin finite element techniques. *Comput. Methods Appl. Mech. Engrg. 190*(48), 6359–6372.

Deheeger, F. (2008). *Couplage mécano-fiabiliste : ^{2}SMART - méthodologie d'apprentissage stochastique en fiabilité.* Ph. D. thesis, Université Blaise Pascal, Clermont-Ferrand.

Dempster, A. P. (1968). A generalization of Bayesian inference. *J. Royal Stat. Soc., Series B* (30), 205–247.

Der Kiureghian, A. and T. Dakessian (1998). Multiple design points in first and second-order reliability. *Structural Safety 20*(1), 37–49.

Dessombz, O., F. Thouverez, J.-P. Laîné, and L. Jézéquel (2001). Analysis of mechanical systems using interval computations applied to finite element methods. *J. Sound Vib. 239*(5), 949–968.

Ditlevsen, O. and H. Madsen (1996). *Structural reliability methods.* J. Wiley and Sons, Chichester.

Efron, B., T. Hastie, I. Johnstone, and R. Tibshirani (2004). Least angle regression. *Annals of Statistics 32*, 407–499.

Efron, B., T. Hastie, and R. Tibshirani (2007). Discussion of "the Dantzig selector" by E. candes and T. Tao. *Ann. Statistics. 35*, 2358–2364.

Efron, B. and C. Stein (1981). The Jacknife estimate of variance. *Annals Statist. 9*(3), 586–596.

Efroymson, M. (1960). *Mathematical models for digital computers*, Volume 1, Chapter Multiple regression analysis, pp. 191–203. Wiley.

Elhay, S. and J. Kautsky (1992). Generalized Kronrod Patterson type embedded quadrature. *Aplikace Matematiky 37*, 81–103.

Fan, J. and R. Li (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of American Statistical Association 96*, 1348–1360.

Fan, J. and R. Li (2006). Sure independence screening for ultra-high dimensional feature space. Technical report, Dept. Operational Research and Financial Engineering, Princeton University.

Foo, J., X. Wan, and G. Karniadakis (2008). The multi-element probabilistic collocation method (ME-PCM): error analysis and applications. *J. Comput. Phys. 227*, 9572–9595.

Forrester, A., A. Sóbester, and A. Keane (2008). *Engineering Design via surrogate modelling.* Wiley.

Frauenfelder, P., C. Schwab, and R. Todor (2005). Finite elements for elliptic problems with stochastic coefficient. *Comput. Methods Appl. Mech. Engrg. 194*, 205–228.

Freudenthal, A. (1947). The safety of structures. *ASCE Trans. 112*, 125–180.

Furnival, G. and R. Wilson, Jr. (1974). Regression by leaps and bounds. *Technometrics 16*, 499–511.

Ganapathysubramanian, B. and N. Zabaras (2007). Sparse grid collocation schemes for stochastic natural convection problems. *J. Comput. Phys. 225*, 652–685.

Gasca, M. and T. Sauer (2000). Polynomial interpolation in several variables. *Adv. Comput. Math. 12*(4), 377–410.

Geisser, S. (1975). The predictive sample reuse method with applications. *J. Amer. Stat. Assoc. 70*, 320–328.

Gerstner, T. and M. Griebel (2003). Dimension-adaptive tensor-product quadrature. *Computing 71*, 65–87.

Ghanem, R. (1998). Probabilistic characterization of transport in heterogeneous media. *Comput. Methods Appl. Mech. Engrg. 158*, 199–220.

Ghanem, R. (1999a). Ingredients for a general purpose stochastic finite elements implementation. *Comput. Methods Appl. Mech. Engrg. 168*, 19–34.

Ghanem, R. (1999b). The nonlinear Gaussian spectrum of log-normal stochastic processes and variables. *J. Applied Mech. 66*, 964–973.

Ghanem, R. (1999c). Stochastic finite elements with multiple random non-Gaussian properties. *J. Eng. Mech. 125*(1), 26–40.

Ghanem, R. and V. Brzkala (1996). Stochastic finite element analysis of randomly layered media. *J. Eng. Mech. 122*(4), 361–369.

Ghanem, R. and D. Dham (1998). Stochastic finite element analysis for multiphase flow in heterogeneous porous media. *Transp. Porous Media 32*, 239–262.

Ghanem, R. and R. Kruger (1996). Numerical solution of spectral stochastic finite element systems. *Comput. Methods Appl. Mech. Engrg. 129*(3), 289–303.

Ghanem, R., G. Saad, and A. Doostan (2006). Efficient solution of stochastic systems: Application to the embankment dam problem. *Structural Safety 29*, 238–251.

Ghanem, R. and P. Spanos (1991). *Stochastic finite elements – A spectral approach*. Springer Verlag. (Reedited by Dover Publications, 2003).

Ghanem, R. and P. Spanos (2003). *Stochastic Finite Elements : A Spectral Appoach*. Courier Dover Publications.

Ghiocel, D. and R. Ghanem (2002). Stochastic finite element analysis of seismic soil-structure interaction. *J. Eng. Mech. 128*, 66–77.

Griebel, M. (2006). Sparse grids and related approximation schemes for higher dimensional problems. Technical report, Technical University of Bonn, Germany.

Grigoriu, M. (1998). Simulation of non-Gaussian translation processes. *J. Eng. Mech. 124*(2), 121–126.

Gunn, S. (1998). Support Vector Machines for classification and regression. Technical report, University of Southampton.

Harbitz, A. (1983). Efficient and accurate probability of failure calculation by use of the importance sampling technique. In *Proc. 4th Int. Conf. on Appl. of Statistics and Probability in Soil and Structural Engineering (ICASP4), Firenze, Italy*, pp. 825–836. Pitagora Editrice.

Hastie, T., J. Taylor, R. Tibshirani, and G. Walther (2007). Forward stagewise regression and the monotone Lasso. *Electronic J. Stat. 1*, 1–29.

Hastie, T., R. Tibshirani, and J. Friedman (2001). *The elements of statistical learning: Data mining, inference and prediction*. Springer, New York.

Helton, J., J. Johnson, C. Sallaberry, and C. Storlie (2006). Survey of sampling-based methods for uncertainty and sensitivity analysis. *Reliab. Eng. Sys. Safety 91*(10-11), 1175–1209.

Hesterberg, T., N. Choi, L. Meier, and C. Fraley (2008). Least angle and ℓ_1 penalized regression: A review. *Statistics Surveys 2*, 61–93.

Hoeffding, W. (1948). A class of statistics with asymptotically normal distributions. *Ann. Math. Stat. 19*, 293–325.

Hoerl, A. and R. Kennard (1970). Ridge regression: applications to nonorthogonal problems. *Technometrics 12*(1), 69–82.

Homma, T. and A. Saltelli (1996). Importance measures in global sensitivity analysis of non linear models. *Reliab. Eng. Sys. Safety 52*, 1–17.

Ishigami, T. and T. Homma (1990). An importance quantification technique in uncertainty analysis for computer models. In *Proc. ISUMA'90, First Int. Symp. Uncertain Mod. An.*, pp. 398–403. University of Maryland.

Isukapalli, S. S. (1999). *Uncertainty Analysis of Transport-Transformation Models*. Ph. D. thesis, The State University of New Jersey.

James, G., P. Radchenko, and J. Lv (2008). DASSO: Connections between the Dantzig selector and Lasso. *J. Royal Stat. Soc., Series B 71*(1).

Jin, R., W. Chen, and T. Simpson (2001). Comparative studies of metamodelling techniques under multiple modelling criteria. *Struct. Multidisc. Optim. 23*, 1–13.

Katafygiotis, L. and S. Cheung (2005). A two-stage subset simulation-based approach for calculating the reliability of inelastic structural systems subjected to Gaussian random excitations. *Comput. Methods Appl. Mech. Engrg. 194*(12-16), 1581–1595.

Keese, A. and H.-G. Matthies (2005). Hierarchical parallelisation for the solution of stochastic finite element equations. *Computers & Structures 83*, 1033–1047.

Kleijnen, J. (2004). An overview of the design and analysis of simulation experiments for sensitivity analysis. *Eur. J. Oper. Res. 164*, 287–300.

Klimke, W. (2006). *Uncertainty modeling using fuzzy arithmetic and sparse grids*. Ph. D. thesis, University of Stuttgart.

Knuth, D. (2005a). *The art of computer programming: Generating all combinations and partitions (Vol. 4, Fascicle 3)*. Addison-Wesley Professional.

Knuth, D. (2005b). *The art of computer programming: Generating all tuples and permutations (Vol. 4, Fascicle 2)*. Addison-Wesley Professional.

Koehler, J. and A. Owen (1996). Computer esperiments. *Handbook of Statistics 13*, 261–308.

Lagaros, N., G. Stefanou, and M. Papadrakakis (2005). An enhanced hybrid method for the simulation of highly skewed non-Gaussian stochastic fields. *Comput. Methods Appl. Mech. Engrg. 194*, 4824–4844.

Le Maître, O., O. Knio, H. Najm, and R. Ghanem (2004). Uncertainty propagation using Wiener-Haar expansions. *J. Comput. Phys. 197*, 28–57.

Le Maître, O., H. Najm, R. Ghanem, and O. Knio (2004). Multi-resolution analysis of Wiener-type uncertainty propagation. *J. Comput. Phys. 197*, 502–531.

Lebrun, R. and A. Dutfoy (2009). A generalization of the nataf transformation to distributions with elliptical copula. *Prob. Eng. Mech. 24*(2), 172–178.

Lee, S. and B. Kwak (2006). Response surface augmented moment method for efficient reliability analysis. *Structural safety 28*, 261–272.

Lemaire, M. (2005). *Fiabilité des structures – Couplage mécano-fiabiliste statique*. Hermès.

Lévi, R. (1949). Calculs probabilistes de la sécurité des constructions. *Annales des Ponts et Chaussées 26*.

Li, R. and R. Ghanem (1998). Adaptive polynomial chaos expansions applied to statistics of extremes in nonlinear random vibration. *Prob. Eng. Mech. 13*(2), 125–136.

Lin, G. and A. Tartakovsky (2009). An efficient, high-order probabilistic collocation method on sparse grids for three-dimensional flow and solute transport in randomly heterogeneous porous media. *Adv. Water Resour. 32*(5), 712–722.

Liu, P.-L. and A. Der Kiureghian (1991). Optimization algorithms for structural reliability. *Structural Safety 9*, 161–177.

Loève, M. (1977). *Probability theory*. Springer Verlag, New-York.

Lophaven, S., H. Nielsen, and J. Søndergaard (2002). DACE - A Matlab kriging toolbox. Technical report, Technical University of Denmark. Software available at www2.imm.dtu.dk/~hbn/dace.

Lucor, D. and G. Karniadakis (2004). Adaptive generalized polynomial chaos for nonlinear random oscillators. *SIAM J. SCi. Comput. 26*(2), 720–735.

Madigan, D. and G. Ridgeway (2004). Discussion of "least angle regression" by Efron et al. *Ann. Statistics 32*, 465–469.

Malliavin, P. (1997). *Stochastic Analysis*. Springer.

Mallows, C. (1973). Some comments on c_p. *Technometrics 15*, 661–675.

Marrel, A. (2008). *Mise en œuvre et utilisation du métamodèle processus gaussien pour l'analyse de sensibilité de modèles numériques : application à un code de transport hydrogéologique*. Ph. D. thesis, Institut National des Sciences Appliquées de Toulouse, France.

Marrel, A., B. Iooss, F. Van Dorpe, and E. Volkova (2007). An efficient methodology for modelling complex codes with Gaussian processes. *Comput. Stat. Data Anal 52*(10), 4731–4744.

Matheron, G. (1967). Kriging or polynomial interpolation procedures. *Canadian Inst. Mining Bull. 60*, 1041.

Matthies, H. and A. Keese (2005). Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations. *Comput. Methods Appl. Mech. Engrg. 194*, 1295–1331.

Mavris, D. and O. Bandte (1997). Comparison of two probabilistic techniques for the assessment of economic uncertainty. In *19th ISPA Conference, New Orleans, LA*.

Mayer, M. (1926). *Die Sicherheit der Bauwerke*. Springer Verlag.

McDonald, D., W. Grantham, and L. Wayne (2006). Global and local optimization using radial basis function response surface models. *Appl. Math. Model. 31*(10), 2095–2110.

McKay, M. (1995). Evaluating prediction uncertainty. Technical report, Los Alamos National Laboratory. NUREG/CR-6311.

McKay, M. D., R. J. Beckman, and W. J. Conover (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics 2*, 239–245.

Meinshausen, N., G. Rocha, and B. Yu (2007). A tale of three cousins: Lasso, l_2-Boosting, and Dantzig. *Annals Statistics 35*, 2373–2384.

Melchers, R.-E. (1999). *Structural reliability analysis and prediction*. John Wiley & Sons.

Miller, R. (1974). The Jacknife – A review. *Biometrika 61*, 1–15.

Millman, D., P. King, and P. Beran (2005). Airfoil pitch-and-plunge bifurcation behaviour with Fourier chaos expansions. *J. Aircraft 42*(2), 376–384.

Molinaro, A., R. Simon, and R. Pfeiffer (2005). Prediction error estimation: a comparison of resampling methods. *Bioinformatics 21*, 3301–3307.

Montgomery, D. (2004). *Design and analysis of experiments*. John Wiley and Sons, New York.

Montgomery, D., E. Peck, and G. Vining (2001). *Introduction to Linear Regression Analysis*. Wiley, New York.

Moore, R. (1979). *Methods and applications of interval analysis*. SIAM Press, Philadelphia, PA.

Morokoff, W. and R. Caflisch (1995). Quasi-Monte Carlo integration. *J. Comput. Phys. 122*, 218–230.

Morris, M. (1991). Factorial sampling plans for preliminary computational experiments. *Technometrics 33*(2), 161–174.

Nair, P. and A. Keane (2002). Stochastic reduced basis methods. *AIAA Journal 40*, 1653–1664.

Nataf, A. (1962). Détermination des distributions dont les marges sont données. *C. R. Acad. Sci. Paris 225*, 42–43.

Nelsen, R. (1999). *An introduction to copulas*, Volume 139 of *Lecture Notes in Statistics*. Springer.

Niederreiter, H. (1992). *Random number generation and quasi-Monte Carlo methods*. SIAM, Philadelphia, PA, USA.

Nobile, F., R. Tempone, and C. Webster (2006). A sparse grid stochastic collocation method for elliptic partial differential equations with random input data. Technical Report MOX Report 85, Politecnico di Milano.

Nouy, A. (2005). Technique de décomposition spectrale optimale pour la résolution d'équations aux dérivées partielles stochastiques. In *Proc. 17ème Congrès Français de Mécanique (CFM17), Troyes*, Number Paper #697.

Nouy, A. (2007a). A generalized spectral decomposition technique to solve stochastic partial differential equations. *Comput. Methods Appl. Mech. Engrg 196*(45-48), 4521–4537.

Nouy, A. (2007b). On the construction of generalized spectral bases for solving stochastic partial differential equations. *Mécanique et Industries 8*(3), 283–288.

Nouy, A. (2008). Generalized spectral decomposition method for solving stochastic finite element equations: invariant subspace problem and dedicated algorithms. *Comput. Methods Appl. Mech. Engrg 197*(51-52), 4718–4736.

Nouy, A. and O. Le Maître (2009). Generalized spectral decomposition for stochastic nonlinear problems. *J. Comput. Phys. 228*, 202–235.

Novak, E. and K. Ritter (1999). Simple cubature formulas with high polynomial exactness. *Constructive Approx. 15*, 499–522.

Oakley, J. and A. O'Hagan (2004). Probabilistic sensitivity analysis of complex models: a Bayesian approach. *J. Royal Stat. Soc., Series B 66*, 751–769.

O'Hagan, A. (2006). Bayesian analysis of computer code outputs: a tutorial. *Reliab. Eng. Sys. Safety 91*, 1290–1300.

Owen, A. (1992). A central limit theorem for Latin hypercube sampling. *J. Royal Stat. Soc., Series B 54*, 541–551.

Owen, A. (1998). Detecting near linearity in high dimensions. Technical report, Stanford University, Department of Statistics.

Pellissetti, M.-F. and R. Ghanem (2000). Iterative solution of systems of linear equations arising in the context of stochastic finite elements. *Adv. Eng. Soft. 31*, 607–616.

Phoon, K., S. Huang, and S. Quek (2002). Implementation of Karhunen-Loève expansion for simulation using a wavelet-Galerkin scheme. *Prob. Eng. Mech. 17*(3), 293–303.

PROSIR (2006). Probabilistic structural integrity of a PWR reactor pressure vessel. In *Round Robins on Probabilistic Approach for Structural Integrity of Reactor Pressure Vessel, OECD - NEA PROSIR Workshop, Lyon, France.*

Rabitz, H., F. Aliş, J. Shorter, and K. Shim (1999). Efficient input–output model representations. *Comput. Phys. Commun. 117*(1-2), 11–20.

Rao, S. and P. Sawyer (1995). Fuzzy finite element approach for the analysis of imprecisely defined systems. *AIAA Journal 33*(12), 2364–2370.

Runge, C. (1901). Über die Darstellung willkürlicher Functionen und die Interpolation zwischen äquidistanten Ordinaten. *Z. Angew. Math. Phys. 46*, 224–243.

Sachdeva, S., P. Nair, and A. Keane (2006). Hybridization of stochastic reduced basis methods with polynomial chaos expansions. *Prob. Eng. Mech. 21*(2), 182–192.

Sacks, J., W. Welch, T. Mitchell, and H. Wynn (1989). Design and analysis of computer experiments. *Stat. Sci. 4*, 409–435.

Saltelli, A., T. Andres, and T. Homma (1995). Sensitivity analysis of model output: performance of the iterated fractional factorial design method. *Comput. Stat. Data. Anal. 20*(4), 387–407.

Saltelli, A., K. Chan, and E. Scott (Eds.) (2000). *Sensitivity analysis*. J. Wiley & Sons.

Saltelli, A. and I. Sobol' (1995). About the use of rank transformation in sensitivity of model output. *Reliab. Eng. Sys. Safety 50*, 225–239.

Saltelli, A., S. Tarantola, and K. Chan (1999). A quantitative, model independent method for global sensitivity analysis of model output. *Technometrics 41*(1), 39–56.

Santner, T., B. Williams, and W. Notz (2003). *The design and analysis of computer experiments*. Springer, New York.

Saporta, G. (1990). *Probabilités, analyse des données et statistique*. Editions Technip.

Saporta, G. (2006). *Probabilités, analyse des données et statistique* (2nd ed.). Editions Technip.

Sathyanarayanamurthy, H. and R. Chinnam (2009). Metamodels for variable importance decomposition with applications to probabilistic engineering design. *Comput. Ind. Eng.*. In press.

Schoutens, W. (2000). *Stochastic processes and orthogonal polynomials*. Springer-Verlag, New York.

Seather, S. and M. Jones (1991). A reliable data-based bandwidth selection method for kernel density estimation. *J. Royal Stat. Soc. Series B 53*(3), 683–690.

Shafer, G. (1976). *A mathematical theory of evidence*. Princeton University Press.

Shinozuka, M. (1983). Basic analysis of structural safety. *J. Struct. Eng. 109*(3), 721–740.

Silverman, B. (1986). *Density estimation for statistics and data analysis.* Chapman and Hall, London.

Smith, S. (2006). Lebesgue constants in polynomial interpolation. *Annales Mathematicae et Informaticae 33*(109-123), 1787–5021.

Smola, A. and B. Schölkopf (2006). A tutorial on support vector regression. *Stat. Comput. 14*, 199–222.

Smolyak, S. (1963). Quadrature and interpolation formulas for tensor products of certain classes of functions. *Soviet. Math. Dokl. 4*, 240–243.

Sobol', I. (1993). Sensitivity estimates for nonlinear mathematical models. *Math. Modeling & Comp. Exp. 1*, 407–414.

Sobol', I. (2003). Theorems and examples on high dimensional model representation. *Reliab. Eng. Sys. Safety 79*, 187–193.

Sobol', I. and S. Kucherenko (2005). Global sensitivity indices for nonlinear mathematical models. Review. *Wilmott magazine 1*, 56–61.

Soize, C. and R. Ghanem (2004). Physical systems with random uncertainties: chaos representations with arbitrary probability measure. *SIAM J. Sci. Comput. 26*(2), 395–410.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *J. Royal Stat. Soc., Series B 36*, 111–147.

Sudret, B. (2007). *Uncertainty propagation and sensitivity analysis in mechanical models – Contributions to structural reliability and stochastic spectral methods.* Habilitation à diriger des recherches, Université Blaise Pascal, Clermont-Ferrand, France.

Sudret, B. (2008). Global sensitivity analysis using polynomial chaos expansions. *Reliab. Eng. Sys. Safety 93*, 964–979.

Sudret, B. and A. Der Kiureghian (2000). Stochastic finite elements and reliability: a state-of-the-art report. Technical Report UCB/SEMM-2000/08, University of California, Berkeley. 173 pages.

Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. *J. Royal Stat. Soc., Series B 58*, 267–288.

Todor, R.-A. and C. Schwab (2007). Convergence rates for sparse chaos approximations of elliptic problems with stochastic coefficients. *IMA J. Numer. Anal 27*, 232–261.

Vapnik, V. (1995). *The nature of statistical learning theory.* Springer-Verlag, New York.

Vapnik, V., S. Golowich, and A. Smola (1997). *Advances in Neural Information Processing Systems 9*, Chapter Support vector method for function approximation, regression estimation, and signal processing. MIT Press.

Vazquez, E. (2005). *Modélisation comportementale de systèmes non-linéaires multivariables par méthodes à noyaux reproduisants*. Ph. D. thesis, Université Paris XI.

Vazquez, E., G. Fleury, and E. Walter (2006). Kriging for inderect measurement, with application to flow measurement. *IEEE T. Instrum. Meas. 55*(1), 343–349.

Vazquez, E. and E. Walter (2003). Multi-output support vector regression. In *Proc. 13th IFAC Symposium on System Identification, Rotterdam*, pp. 1820–1825.

Wan, X. and G. Karniadakis (2006). Beyond Wiener-Askey expansions: handling arbitrary PDFs. *J. Sci. Comput. 27*, 455–464.

Wan, X. and G. E. Karniadakis (2005). An adaptive multi-element generalized polynomial chaos method for stochastic differential equations. *J. Comput. Phys. 209*, 617–642.

Wand, M. and M. Jones (1995). *Kernel smoothing*. Chapman and Hall.

Wang, G. (2003). Adaptive response surface method using inherited Latin Hypercube design points. *J. Mech. Design 125*, 210–220.

Wei, D. and S. Rahman (2007). Structural reliability analysis by univariate decomposition and numerical integration. *Prob. Eng. Mech. 22*, 27–38.

Welch, W., R. Buck, J. Sacks, H. Wynn, T. Mitchell, and M. Morris (1992). Screening, predicting, and computer experiments. *Technometrics 34*, 15–25.

Wiener, N. (1938). The homogeneous chaos. *Amer. J. Math. 60*, 897–936.

Witteveen, J., S. Sarkar, and H. Bijl (2007). Modeling physical uncertainties in dynamic stall induced fluid-structure interaction of turbine blades using arbitrary polynomial chaos. *Computers & Structures 85*, 866–878.

Xiu, D. (2009). Fast numerical methods for stochastic computations: a review. *Comm. Comput. Phys. 5*(2-4), 242–272.

Xiu, D. and J. Hesthaven (2005). High-order collocation methods for differential equations with random inputs. *SIAM J. Sci. Comput. 27*(3), 1118–1139.

Xiu, D. and G. Karniadakis (2002). The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput. 24*(2), 619–644.

Xiu, D., D. Lucor, C.-H. Su, and G. Karniadakis (2002). Stochastic modeling of flow-structure interactions using generalized polynomial chaos. *J. Fluid. Eng. 124*(1), 51–59.

Xiu, D., D. Lucor, C.-H. Su, and G. Karniadakis (2003). Performance evaluation of generalized polynomial chaos. In *Int. Conf. Comput. Science*, pp. 346–354.

Xu, C. and G. Gertner (2008). Uncertainty and sensitivity analysis for models with correlated parameters. *Reliab. Eng. Sys. Safety 93*, 1563–1573.

Zadeh, L. (1978). Fuzzy sets as the basis for a theory of possibility. *Fuzzy Sets and Systems 1*, 3–28.

Zhang, J. and B. Ellingwood (1994). Orthogonal series expansion of random fields in reliability analysis. *J. Eng. Mech. 120*(12), 2660–2677.

Zou, H. (2006). The adaptive Lasso and its oracle properties. *J. American Stat. Assoc. 101*, 1418–1429.

Zou, H., T. Hastie, and R. Tibshirani (2007). On the "degrees of freedom" of the Lasso. *Ann. Statistics 35*, 2173–2192.

Zou, H. and R. Li (2008). One-step sparse estimates in nonconcave penalized likelihood models. *Ann. Statistics*. to appear.