Chapter 4
Sensitivity and Reliability Analysis of Engineering Structures: Sampling Based Methods

M. Oberguggenberger

Abstract This chapter intends to present an overview of Monte Carlo-type methods currently in use in the probabilistic analysis of large engineering structures. It starts with an introduction to the generation of multi-dimensional random quantities. Next, spatially distributed random properties, e.g., material or geometrical properties in continuum mechanics, are modeled as random fields. Approximations to random fields by means of Karhunen–Loève expansion and polynomial chaos expansion are introduced. These tools are employed to study the response of continuous structures with loads, material or geometrical properties given by random fields. The main focus is on sensitivity analysis of large engineering structures, where small Monte Carlo sample sizes are mandatory. The transition to reliability is undertaken by means of the concept of tolerance intervals. Further, current sampling methods for accurate reliability estimates are discussed, and practical applications are presented.

4.1   Introduction

Engineering structures are usually modelled as input–output maps: the response Y is a function $Y = g(X_1, \ldots, X_n)$ of input parameters $(X_1, \ldots, X_n)$ like material properties, geometry, boundary conditions, and driving forces (dynamic or distributed loads, noise). It has been acknowledged since a long time that both the structural model (given by the function g) and the input parameters are uncertain. Traditionally, uncertainties have been dealt with by employing safety factors. That is, the traditional codes would require that the load carrying capacity of the structure exceeds the design loads by a certain factor > 1, typically 1.35 for permanent loads (such as dead weight) and 1.5–2.0 for temporary loads.

M. Oberguggenberger (✉)
Unit of Engineering Mathematics, University of Innsbruck, Technikerstr. 13, A6020 Innsbruck, Austria
e-mail: Michael.Oberguggenberger@uibk.ac.at

This state of affairs is unsatisfactory in as much as no information about the actual distance to failure can be extracted. The desire for a more analytical description of the uncertainties led to the introduction of the probabilistic safety concept in civil engineering, initiated by the pioneering work of Freudenthal [17], Bolotin [6], and others in the 1950s. Starting with the 1980s and 1990s, the European engineering codes have been changed into probability based codes. By now, this is the standard in civil engineering (see, e.g., EN 1990:2002 [15])— interestingly, the civil engineering community has been far ahead of the other engineering fields in adopting the probabilistic point of view.

Under this point of view, every relevant parameter of the engineering model is a random variable. There is no absolute safety, but rather a probability of failure. As a consequence, more information than just the nominal parameter values must be entered in the model, namely a description of the statistical distribution of the input. Further, the response is no longer deterministic, but rather a random variable, whose distribution must be computed in order to describe the behavior of the structure as well as the probability that certain limits are exceeded (described by a limit state function).

In practical applications, the structure is usually represented by a finite element model. These models are generally large, computationally costly, and partially black boxes. Practically, Monte Carlo simulation is the only way to numerically compute the statistics of the system response. Thereby, an artificial sample of $X_1, \ldots, X_n$, a data matrix of size $N \times n$, is generated and N values of $y = g(x_1, \ldots, x_n)$ are calculated, producing a sample of size N of the response Y, which in turn can be evaluated statistically. This approach raises the computational cost dramatically, and so the need for cost-saving algorithms arises.

An adequate understanding of the uncertainties in an engineering task requires a number of actions, among them reflection about the choice of model and the failure mechanisms; assessing the variability of input and output variables and model parameters; sensitivity analysis (i.e., the determination of the relative influence of individual input parameters on the response); assessing the reliability of the structure. This involves a variety of activities to be performed, from laboratory experiments, data collection to model validation.

The reader is alerted that in the present contribution, only the comparatively narrow part of the numerical calculation of sensitivities and of reliability is addressed. As suggested in the title, the focus is on sampling based methods. In view of the need to employ as few model evaluations as possible, the choice of the sample becomes an important issue. This is dealt with under the heading *design of experiment* in Sect. 4.2. In due course, *metamodels* will be encountered there as well. Section 4.3 is devoted to the simulation of *random fields*, that is, spatially distributed random input. Section 4.4 starts off with *sensitivity analysis*, of interest in itself, but also the basis for model reduction. This becomes useful in Sect. 4.5, where *reliability analysis* is addressed. In Sect. 4.6, the concepts will be illustrated using a model from aerospace engineering, supplied by our industrial partner Intales GmbH Engineering Solutions.

The methods presented here have been developed, adapted, and implemented in a number of joint research projects with Intales GmbH Engineering Solutions.[1] Note that approaches to uncertainty analysis going beyond probability theory, such as interval analysis or the combination of both approaches in the form of random sets, are not addressed here. One instance of such a hybrid approach is in Chap. 3 of this volume. For further information the reader is referred to the recent surveys [4, 36].

4.2 Design of Experiment

In this section, the task of simulating the output $Y = g(X_1, \ldots, X_n)$ of an input–output function applied to random input $(X_1, \ldots, X_n)$ will be addressed. Direct Monte Carlo simulation consists in generating a sample $\mathbf{x}_1, \ldots, \mathbf{x}_N$ of the n-dimensional random variable $(X_1, \ldots, X_n)$, collected in an $N \times n$-matrix.[2] The sample has to be generated in such a way that the columns are statistically independent and each of them is distributed according to the distribution of the corresponding random variable. We are not going to detail this step—most scientific software packages come with a pseudorandom number generator that can produce high dimensional independent samples of most familiar statistical distributions [42] of sufficiently large size (the crucial question of accuracy will be addressed below). The term *design of experiment* refers to the choice of the sample so as to achieve certain desirable additional properties.

Subsequently, each sampled row $\mathbf{x}_j = (x_{ji}, \ldots, x_{jn})$ is sent through the input–output map to produce a sample $y_j = g(x_{j1}, \ldots, x_{jn})$, $j = 1, \ldots, N$ of the output Y.

The complete information about the statistical properties of the output Y is contained in its cumulative distribution function

$$F_Y(y) = P(Y \leq y) = P(g(X_1, \ldots, X_n) \leq y)$$

which in turn can be written as an expectation value, namely as

$$F_Y(y) = \mathrm{E}(h(Y)) = \mathrm{E}(h(g(X_1, \ldots, X_n)))$$

where h is the indicator function of the interval $(-\infty, y]$, i.e., $h(z) = 1$ for $z \leq y$ and 0 otherwise. Similarly, all statistical properties of the output Y can

---

[1]ICONA-project 2006–2008, supported by TransIT Innsbruck, ACOSTA-project 2008–2010, supported by The Austrian Research Promotion Agency, MDP-NE 2011–2013, supported by Astrium GmbH; main partners: Intales GmbH Engineering Solutions, Institute of Basic Sciences in Engineering Science and Institute of Mathematics, University of Innsbruck, Czech Technical University in Prague.

[2]We follow the common statistical practice that random variables are denoted by capital letters, while their realizations are denoted by small letters.

be formulated in terms of expectation values of functions of Y. For example, the moments of Y are obtained by choosing $h(z) = z^m$, $m = 1, 2, 3, \ldots$. The core of Monte Carlo simulation is that these expectation values can be approximated by the corresponding sample mean, that is,

$$\mathrm{E}\big(h(Y)\big) \approx \overline{h(Y)} = \frac{1}{N} \sum_{j=1}^{N} h(y_j) = \frac{1}{N} \sum_{j=1}^{N} h\big(g(x_{ji}, \ldots, x_{jn})\big).$$

By construction, $y_1, \ldots, y_N$ is an independent random sample, hence statistical sampling theory tells us that the variance of the estimator $\overline{h(Y)}$ is given by

$$\mathrm{V}\big(\overline{h(Y)}\big) = \frac{1}{N} \mathrm{V}\big(h(Y)\big) = \frac{C^2}{N}$$

where C^2 is the variance of $h(Y) = h\big(g(X_1, \ldots, X_n)\big)$, a fixed number depending only on h, g, and the given distribution of $(X_1, \ldots, X_n)$. Thus the mean error of a Monte Carlo estimate is of order $1/\sqrt{N}$. For methods to generate random samples leading to a numerical error approximately below prescribed bounds see [19].

We note in passing that replacing the pseudorandom numbers by quasirandom numbers, generated from the so-called low-discrepancy sequences, allows one to improve the mean square error to order $(\log N)^n/N$, but demonstrably *not* further [13, 34]. Rather than going into design of experiment based on quasirandom numbers, two sampling plans will be addressed which are of bigger importance in our setting.

Latin Hypercube Sampling The first issue is stratified sampling that is designed to avoid random clustering and produces sampled points with a balanced distribution over the parameter space. A prominent and easy-to-implement method of stratified sampling is *Latin hypercube sampling*. To obtain a sample of size N, the Latin hypercube sampling plan divides the range of each variable X_i into N disjoint subintervals of equal probability. First, N values of each variable X_i, $i = 1, \ldots, n$, belonging to the respective subintervals are randomly selected. Then the N values for X_1 are randomly paired without replacement with the N values for X_2. The resulting pairs are then randomly combined with the N values of X_3 and so on, until a set of N n-tuples is obtained. This set forms the Latin hypercube sample. The advantage of Latin hypercube sampling is that sampled points are evenly distributed through design space, thereby hitting also regions of low probability possibly important for the input–output map which might be missed by direct Monte Carlo simulation. A Latin hypercube estimate is not necessarily more accurate than a standard Monte Carlo estimate at given N, but it can be shown that the variance of a Latin hypercube estimator is asymptotically smaller than the variance of the direct Monte Carlo estimator, and possibly markedly smaller when the input–output map is partially monotonic [53].

Correlation Control The second issue is correlation control, which is an essential ingredient in Monte Carlo simulation with small sample sizes (say, around $N = 100$ or less). As the reader may easily verify, the rows of an independently sampled matrix $\mathbf{X} = (x_{ij}, i = 1, \ldots, N; j = 1, \ldots n)$ of independent random variables $X_1, \ldots, X_n$ may turn out to have correlation coefficients up to 20% in practice, when N is that small (this undesirable effect disappears for $N \approx 1{,}000$). Thanks to an empirical method due to Iman and Conover [21], it is possible to rearrange the entries of the sampled matrix in such a way that the new columns are nearly uncorrelated. In fact, the method allows one to construct a matrix $\mathbf{X}^*$ of any desired correlation structure $\mathbf{K}$. This is done as follows. The van der Waerden matrix $\mathbf{W}$ is defined by

$$\mathbf{W} = \begin{pmatrix} w_1^{(1)} & \ldots & w_1^{(n)} \\ \vdots & & \vdots \\ w_N^{(1)} & \ldots & w_N^{(n)} \end{pmatrix}$$

where each column consists of a random permutation of the van der Waerden scores

$$\Phi^{-1}\left(\frac{j}{N+1}\right), \qquad j = 1, \ldots, N.$$

Here Φ denotes the standard normal cumulative distribution function. Starting with the Cholesky factorizations

$$\mathbf{K} = \mathbf{P}\mathbf{P}^\mathsf{T}, \quad \boldsymbol{\rho}_W = \mathbf{Q}\mathbf{Q}^\mathsf{T},$$

with the correlation matrix $\boldsymbol{\rho}_W$ of $\mathbf{W}$, one can prove that

$$\mathbf{W}^* = \mathbf{W}\mathbf{Q}^{-\mathsf{T}}\mathbf{P}^\mathsf{T}$$

has the target correlation structure $\mathbf{K}$. Empirical investigations [21] showed that the rank correlation matrix of the resulting matrix $\mathbf{W}^*$ is nearly the same, i.e., $\rho_{W*} \approx \mathbf{R}_{W*}$. Therefore, rearranging the values in the columns of $\mathbf{X}$ corresponding to the rank order of the columns in $\mathbf{W}^*$ leads to a matrix $\mathbf{X}^*$ which approximately has the desired correlation structure:

$$\rho_{X*} = \rho_{W*} \approx \mathbf{R}_{W*} = \mathbf{K}.$$

Further improvement can usually be achieved by iteration of the procedure. It should be noted that the described method of correlation control does not destroy the Latin hypercube structure of a sample and thus can be directly combined with Latin hypercube sampling. The efficiency of correlation control in dependence on the number of input variables has been studied in [37].

In all simulations presented in this paper, Latin hypercube sampling and correlation control have been routinely implemented.

Bootstrap Resampling   The result of a Monte Carlo simulation is a single estimate $\overline{h(Y)}$ of one or more desired quantities $h(Y)$. One would like to be able to assess the accuracy of the estimate, i.e., the variance of the estimator, confidence intervals, etc., without additional calls of the expensive input–output map. A cost-saving method to achieve this is *bootstrap resampling* [14, 50]. The bootstrapping procedure consists in repeatedly drawing from the same sample with replacement to obtain new samples of the same size N. To obtain $B = 1,000$, say, bootstrap samples of size N, one proceeds as follows.

From the original data sample, e.g., $h(y_1), \ldots, h(y_N)$, of size N one randomly draws N-times, so that each realization has equal probability of being drawn. The results are combined to produce a bootstrap sample of size N (note that some entries of the bootstrap sample may be repetitions of realizations of the original data sample). This is repeated $B = 1,000$ times. The $B = 1,000$ bootstrap samples now are used to compute $B = 1,000$ realizations of $\overline{h(Y)}$, say, and the distribution of $\overline{h(Y)}$ can be estimated in this way. The reason why this works is that each bootstrap sample has a distribution which approximates the empirical distribution of the original sample.

In this way, the generation of a multitude of samples of the same distribution is mimicked and allows one to assess the variability of the individual sample estimators. For example, confidence intervals for $\overline{h(Y)}$ can be either obtained by computing the percentiles of the $B = 1,000$ estimates of $\overline{h(Y)}$, or approximated by a Student's t-distribution based on the empirical standard deviation of those values.

Metamodels   Also known as surrogate models or response surfaces, metamodels attempt to save computational cost by approximating the input–output function by a simpler (deterministic) function. Typically, such an approximation is based on evaluating the input–output function at a smaller number of design points and suitable extrapolation. Large size Monte Carlo simulation can then be performed with the metamodel with little computational cost. Metamodels obtained by linear regression with possibly nonlinear shape functions have the advantage that the powerful diagnostic methods of regression analysis can be used. For example, partial coefficients of determination admit to quantify the relative importance of input variables with respect to the variability of the output in nonparametric ways [28, 32, 39]. Other metamodels are based on radial basis functions, smoothing splines, or Kriging (i.e., variance minimizing piecewise linear extrapolation); see, e.g., [25, 45]. The accuracy of a metamodel crucially depends on the degree of smoothness of the input–output function. Metamodels cannot be used, e.g., to accurately describe nonlinear bifurcation as in buckling analysis. On the other hand, given a sufficiently smooth model, the accuracy of a metamodel can be controlled by sequential design of experiment, in which additional design points are added in regions of lower accuracy, optimizing both the global error and the space filling properties of the experimental design, see, e.g., [23].

Stochastic Response Surfaces If the input variables $(X_1, \ldots, X_n)$ are Gaussian or have been transformed into standard Gaussian variables, the input–output function can be seen as a function on standard Gaussian space and approximated by a response surface on that space. As an illustration, consider the univariate case of a single random variable X with distribution function $F(x)$. The transformed random variable $U = \Phi^{-1}(F(X))$ has a standard Gaussian distribution. This transformation reduces the input–output map to a function of the Gaussian variable U as well, by means of $Y(U) = g\big(F^{-1}(\Phi(U))\big)$. For example, if X has a normal distribution with mean μ and variance σ^2, the transformation is simply $X = F^{-1}(\Phi(U)) = \mu + \sigma U$.

Recall that the Hermite polynomials $h_n(u)$ form an orthonormal basis in the space of square integrable functions on the real line with respect to the Gaussian density $e^{-u^2/2} \, du / \sqrt{2\pi}$. The (normalized) Hermite polynomials are given by the recursion

$$h_{n+1}(u) = \frac{u}{\sqrt{n+1}} h_n(u) + \frac{n}{\sqrt{n(n+1)}} h_{n-1}(u)$$

with $h_0(u) = 1$, $h_1(u) = u$. Every function $Y(U)$ of a Gaussian variable U such that $Y(U)$ has finite second moments has a convergent Hermite expansion of the form

$$Y(U) = \sum_{k=0}^{\infty} c_k h_k(U).$$

The coefficients c_k can be obtained as the inner product $\mathrm{E}(Y(U)h_n(U))$; alternatively, collocation and regression can be used to numerically compute them. More precisely, choose finitely many points $\xi_1, \ldots, \xi_m$ in the domain of the input–output map g. Compute collocation points $u_j = \Phi^{-1}(F(\xi_j))$ on the real line. Record the outputs $y_j = g(\xi_j) = g\big(F^{-1}(\Phi(u_j))\big)$. Evaluate the coefficients $c_1, \ldots, c_M$ of a truncated Hermite expansion by linear regression on the data (u_j, y_j) with the Hermite functions h_k, $k = 1, \ldots, M$, as shape functions. This concludes the construction of a stochastic response surface $Y_M(U)$ for the input–output function, given as a truncated Hermite series. Monte Carlo simulation is now done at no cost by sampling a standard Gaussian variable U and evaluating $Y_M(U)$.

Note that this procedure requires only m evaluations of the costly input–output map on the points $\xi_1, \ldots, \xi_m$. The rest of the burden is put on the transformation $U = \Phi^{-1}(F(X))$, thus parametric studies with differently distributed X, for example, with varying mean and variance, can be easily undertaken. In the one-dimensional case a low value of M, say around ten, and twice the number of collocation points usually suffices. For an application of this method, see, e.g., [35].

As is well-known, the procedure can be generalized to multiple expansions of functions of infinitely many Gaussian variables, known as the polynomial chaos expansion; the reader is referred to e.g., [18, 29].

4.3 Random Fields

Material and geometrical properties (e.g., modulus of elasticity, thickness) of a structure may vary randomly from point to point. Such a behavior can be captured by means of *random fields*, that is, stochastic processes that assign a random variable $q(x)$ to every point x in a region in space. Usually, random fields are chosen so as to have continuous or even differentiable realizations, as opposed to random noise in stochastic mechanics. To define the field, the joint distributions of the values at any finite number of points $q(x_1), \ldots, q(x_k)$ should be specified. If the random field is stationary (i.e., the finite dimensional distributions are translation invariant) and Gaussian, it is completely specified by the mean value $\mu_q = \mathrm{E}(q(x))$ and the second moments, i.e., the covariance $\mathrm{COV}(q(x), q(y))$ for any two points x, y. Due to stationarity, the covariance depends only on the distance $\delta = |x - y|$ of the points and is of the form

$$\mathrm{COV}(q(x), q(y)) = C(x, y) = \sigma^2 c(\delta)$$

with the variance σ^2 and the autocorrelation function $c(\delta)$. A frequently used autocorrelation function is of the form

$$c(\delta) = \exp\left(-|\delta|/\ell\right), \tag{4.1}$$

where ℓ is the so-called correlation length. The indicator function of the interval $[-\ell, \ell]$ might be taken as a crude autocorrelation function with correlation equal to 1 for δ in the interval and 0 outside. The area under the curve (4.1) is the same as the area under this indicator function, whence the name correlation length. Other autocorrelation functions in use may be of Gaussian type, in higher dimensions also with anisotropic distance measure.

If measurement data are available, the autocorrelation function can be estimated from the empirical covariance matrix by arranging the values along the distance δ of the measured points and fitting a shape function as in (4.1), thereby estimating σ^2 and ℓ.

In order to simulate a random field, one discretizes the region under consideration with grid points x_i, $i = 1, \ldots, M$, measures the distance between the grid points x_i, and sets up a covariance matrix $\mathbf{C} = (C_{ij}, i, j = 1, \ldots, M)$ whose values are computed from (4.1) where δ is the distance between x_i and x_j. In case the random field is Gaussian, there are at least three methods to generate realizations of the field.

The first method is the standard simulation method for correlated Gaussian variables. It is based on the Cholesky factorization $\mathbf{C} = \mathbf{A}\mathbf{A}^\mathsf{T}$. If $\mathbf{Y}$ is an M-dimensional Gaussian random variable with mean zero and independent components (i.e., its covariance matrix $\mathrm{E}(\mathbf{Y}\mathbf{Y}^\mathsf{T}) = \mathbf{I}$, the identity matrix), then $\mathbf{X} = \mathbf{A}\mathbf{Y}$ is a mean-zero Gaussian random variable whose covariance matrix is $\mathbf{C}$. This follows from the simple identities

$$\mathrm{E}(\mathbf{X}\mathbf{X}^\mathsf{T}) = \mathrm{E}(\mathbf{A}\mathbf{Y}\mathbf{Y}^\mathsf{T}\mathbf{A}^\mathsf{T}) = \mathbf{A}\mathrm{E}(\mathbf{Y}\mathbf{Y}^\mathsf{T})\mathbf{A}^\mathsf{T} = \mathbf{A}\mathbf{I}\mathbf{A}^\mathsf{T} = \mathbf{C}.$$

Accordingly, starting from a realization of the M-dimensional standard Gaussian random variable $\mathbf{Y}$, the transformation $\mathbf{X} = \mu_q + \mathbf{AY}$ yields a realization of the desired random field $q(x)$ in the grid points, i.e., $X_i = q(x_i)$. This is repeated N times to obtain a Monte Carlo sample of the random field. It should be noted that this method works in any space dimension; it just requires enumerating the grid points and keeping track of their distance. The disadvantage of this method is that one cannot easily keep track of the error in terms of the number of grid points, that is, the accuracy of the autocovariance function of the simulated field.

The second method is advantageous in this respect. It is based on the *Karhunen–Loève expansion* of the field. In fact, the eigenvalue problem

$$\int C(x, y)\varphi_k(y)\,\mathrm{d}y = \lambda_k\varphi_k(x)$$

where $C(x, y)$ is the autocovariance function of the random field, has a sequence of positive eigenvalues λ_k and orthonormal eigenfunctions $\varphi_k(x)$ (orthonormality in mean square). Then

$$q(x) = \sum_{k=1}^{\infty} \sqrt{\lambda_k}\xi_k\varphi_k(x) \tag{4.2}$$

where the ξ_k are uncorrelated random variables with unit variance, see, e.g., [30]. If the process is Gaussian, the ξ_k are independent and distributed according to the standard normal distribution $\mathcal{N}(0, 1)$.

For the numerical simulation, the spatial region is again discretized by a grid and the φ_k are taken, e.g., piecewise constant on the grid elements. The eigenvalue problem becomes a matrix eigenvalue problem, and the series (4.2), with approximate eigenvalues and eigenfunctions, truncated after a finite number M of terms, can be used for Monte Carlo simulation of the field trajectories. Here the mean square error due to truncation after M terms is just the sum of the neglected eigenvalues; the discretization error can be estimated through the numerical integration error and its propagation through eigenvalue problems [5]. A further advantage of the method is that it can be directly based on a finite element discretization [46]. However, changing the field parameters, e.g., the correlation length, requires solving the eigenvalue problem with a different matrix anew, which can be costly.

This disadvantage is avoided in the third method, which is applicable in one space dimension. It is based on the observation that the autocorrelation function (4.1) coincides with the autocorrelation function of an Ornstein–Uhlenbeck process, namely the solution process of the Langevin stochastic differential equation

$$\mathrm{d}q(x) = -\tfrac{1}{\ell}q(x) + \sqrt{\tfrac{2}{\ell}}\sigma\,\mathrm{d}w(x), \quad q(0) \sim \mathcal{N}(0, \sigma^2), \tag{4.3}$$

where $w(x)$ denotes Wiener process on the real line, see e.g. [2]. Solutions of sufficient accuracy can be easily simulated from discrete white noise input by means of an explicit Euler scheme at little cost [26].

4.4 Sensitivity Analysis

Sensitivity analysis is a core ingredient in understanding the behavior of a structural model. It aims at determining the input parameters that have the largest influence on critical output. In addition, it can be used as a first step in reliability analysis or in optimizing structural properties.

Sensitivity analysis does not necessarily require knowledge of the probabilistic properties of the input and thus is a nonparametric method. If the input–output function is explicitly given and sufficiently smooth, one may use partial derivatives to assess the sensitivity, see, e.g., [40, 45]. In the context we envisage, the input–output function may be non-differentiable and a black box, in addition. For this reason we focus on derivative-free methods of sensitivity analysis, that is, on sampling based methods. The strategy is to produce a sample of the input data $(X_1, \ldots, X_n)$, to compute a sample of the output Y and to analyze the statistical input–output relations or the relations between different output quantities. For all variables, we take a uniform distribution centered around the nominal values μ_j with a spread of a certain equal percentage, say $\pm 15\%$. Equally scaled spread and uniform distributions are chosen to avoid distortion of the relative weights of the input variables. If information about the actual statistical distribution of one or the other input variable is known, this knowledge first does not enter in the sensitivity analysis, but may be considered in a second stage.

The computationally least expensive methods are correlation based, which will be described first. An explorative analysis usually starts with inspecting scatterplots of individual variables vs. output. To obtain a refined diagnosis, methods are needed that quantify the correlations, assess their significance, and possibly remove hidden influences of the co-variates on the correlation between a given input variable and the output variable. The simplest indicator is the Pearson correlation coefficient (CC). It detects linear relationships between input and output. To recall the definition, assume given a sample $x_{j1}, \ldots, x_{jn}$, and y_j, $j = 1, \ldots, N$ of the n-dimensional input and the corresponding output. Denote the mean values by $\overline{x}_i$ and $\overline{y}$, respectively. The empirical Pearson correlation coefficient of input number i with the output is defined as

$$r(x_i, y) = \frac{\sum_{j=1}^{N}(x_{ji} - \overline{x}_i)(y_j - \overline{y})}{\sqrt{\sum_{j=1}^{N}(x_{ji} - \overline{x}_i)^2 \sum_{j=1}^{N}(y_j - \overline{y})^2}}.$$

It turns out that the correlation coefficient does not isolate the effect of x_i on the output y, but is influenced by the co-variates (inputs with numbers $k \neq i$),

especially when they have a nonzero correlation with x_i. Regression based indices may be used to mitigate this effect.

A brief recall of linear regression is in order. The goal of linear regression analysis is to fit a linear model $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n$ to the data, that is, each value y_j is to be approximated by

$$y_j = \beta_0 + \beta_1 x_{j1} + \beta_2 x_{j2} + \ldots + \beta_n x_{jn} + \varepsilon_j, \quad j = 1, \ldots, N$$

with the errors ε_j. The estimated coefficients $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_n$ are obtained as the solution of the minimization problem

$$L(\beta_0, \beta_1, \ldots, \beta_n) = \sum_{j=1}^{N} \varepsilon_j^2 \rightarrow \min.$$

The values predicted by the model and the residuals are, respectively,

$$\hat{y}_j = \hat{\beta}_0 + \hat{\beta}_1 x_{j1} + \ldots + \hat{\beta}_n x_{jn}, \qquad e_j = y_j - \hat{y}_j,$$

$j = 1, \ldots, N$. If there is no linear relation between input and output, the best prediction is the mean value $\overline{y}$, in which case the residuals coincide with the measured data y_j, centered at the mean. The other extreme is that the data points y_j already lie on a hyperplane $y_j = \beta_0 + \beta_1 x_{j1} + \beta_2 x_{j2} + \ldots + \beta_n x_{jn}$, in which case the best prediction is simply the data point, $\hat{y}_j = y_j$, and the residuals are identically equal to zero.

It can be shown that the total square variability of y can be partitioned into two summands:

$$\sum_{j=1}^{N}(y_j - \overline{y})^2 = \sum_{j=1}^{N}(\hat{y}_j - \overline{y})^2 + \sum_{j=1}^{N}(y_j - \hat{y}_j)^2.$$

The coefficient of determination is defined as $R^2 = \sum_{j=1}^{N}(\hat{y}_j - \overline{y})^2 / \sum_{j=1}^{N}(y_j - \overline{y})^2$. By what has been said above about the residuals, it equals 1 if the data points already lie on a hyperplane and 0 when no linear relationship between input and output exists, and in general measures the explanatory power of the fitted regression model.

The regression coefficients as such cannot be used as indicators of the influence of the corresponding variables, because they are scale dependent. Rather, the *standardized regression coefficients* (SRC) can be used. These are the regression coefficients of the centered and normalized model, where the data x_{ji} are replaced by $(x_{ji} - \overline{x}_i) / \sqrt{\sum_{j=1}^{N}(x_{ji} - \overline{x}_i)^2}$, and similarly for the y_j. In case the n input columns $x_{j1}, \ldots, x_{jn}$, $j = 1, \ldots, N$ are uncorrelated, the SRCs coincide with the CCs. In general, the expression for the CCs has an additional summand which depends on the correlated co-variates.

A more effective removal of the influence of the co-variates is achieved through the *partial correlation coefficients* (PCCs). The partial correlation between x_i and y, given the set of co-variates $x_{\smallsetminus i} = \{x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n\}$ is defined as the correlation between the two residuals obtained by regressing x_i on $x_{\smallsetminus i}$ and y on $x_{\smallsetminus i}$, respectively. That is, one first constructs the two regression models

$$\hat{x}_i = \hat{\alpha}_0 + \sum_{k \neq i} \hat{\alpha}_k x_k, \qquad \hat{y} = \hat{\beta}_0 + \sum_{k \neq i} \hat{\beta}_k x_k,$$

obtaining the residuals $\mathbf{e}_i$ and $\mathbf{e}$ with components

$$e_{ji} = x_{ji} - \hat{x}_{ji}, \qquad e_j = y_j - \hat{y}_j,$$

$j = 1, \ldots, N$. By construction, the residuals e_{ji} and e_j are those parts of x_i and y that remain after subtraction of the predicted linear part depending on $x_{\smallsetminus i}$. Thus the PCC $\rho(\mathbf{e}_i, \mathbf{e})$ quantifies the linear relationship between x_i and y after removal of any part of the variation due to the linear influence of the co-variates $x_{\smallsetminus i}$.

The advantage of the PCCs is that they are more discriminating. In fact, if the input–output map is a truly linear function and the input parameters are uncorrelated, then the PCC of an input variable that enters with a non-zero coefficient is equal to plus or minus one. In reality, input–output maps are not ideally linear functions and so the effect is somewhat moderated. Still the PCCs are an accentuating measure of influence.

If the input–output function is decidedly nonlinear, but monotonic, sensitivities are better detected when one applies a rank transformation to the data. That is, the data $x_{j1}, \ldots, x_{jn}$, and y_j, $j = 1, \ldots, N$ are ordered and only their rank information is kept. This leads to the Spearman rank correlation coefficients (RCC), the standardized rank regression coefficients (SRRC), and the partial rank correlation coefficients (PRCC). Having the computed Monte Carlo sample $x_{j1}, \ldots, x_{jn}$, and y_j at hand, the calculation of the various coefficients produces no additional cost, thus it is recommended to evaluate all six of them to have a better overview. Finally, bootstrap resampling allows one to compute confidence intervals. If zero is outside the confidence interval, the corresponding coefficient can be considered to be significantly different from zero, and the corresponding input variable is classified as having a non-negligible influence on the output. The degree of influence can then be classified according to the magnitude of the six coefficients.

Alternative methods of sensitivity analysis are variance based. Pinching strategies consist in freezing individual variables at their central value and studying the change of variability in the output. If one produces a sample with X_i fixed to its nominal value μ_i, the reduction in variance says something about the influence of X_i on Y:

$$\frac{V(g(X_1, \ldots, X_{i-1}, \mu_i, X_{i+1}, \ldots, X_n))}{V(g(X_1, \ldots, X_n))}$$

In fact, this is the proportion of total variance *not* explained by X_i. As can already be seen from the formulation, this strategy is costly because for each pinched variable a new Monte Carlo simulation is required.

A more sophisticated method is the partition of variance according to the groups of variables, the Sobol' indices introduced in [51, 52]. It is based on an expansion of the input–output function into summands of increasing dimensionality.

For a survey of sampling based methods in sensitivity analysis, see [20].

4.5 Reliability Analysis

The central concept of reliability analysis is the *failure probability*. The system is considered in a failed state if a certain combination of the input parameters $(X_1, \ldots, X_n)$ and the output Y exceeds an admissible range. For the present purpose it is not necessary to distinguish into favorable (resistance increasing) and unfavorable (load or stress exerting) influences, as is done in the European civil engineering codes [15] with their partial safety factors, critical values and design values. Since Y is a function of the inputs $(X_1, \ldots, X_n)$, which subsume all random influences on the structure, failure can be described by a *limit state function* $\Phi(X_1, \ldots, X_n)$ of the input alone. Failure is usually defined by $\Phi(X_1, \ldots, X_n) < 0$, while $\Phi(X_1, \ldots, X_n) \geq 0$ signifies a safe state. The *failure region* is the subset of design space (the domain of the input parameters) resulting in violation of the limit state condition, i.e.,

$$\mathscr{F} = \{(x_1, \ldots, x_n) : \Phi(x_1, \ldots, x_n) < 0\}.$$

Then

$$p_f = P\big((X_1, \ldots, X_n) \in \mathscr{F}\big) = P\big(\Phi(X_1, \ldots, X_n) < 0\big)$$

is the failure probability; $R = 1 - p_f$ is the *reliability* of the structure. Occasionally it is useful to describe failure as a ratio of actual and admissible values, leading to a failure region of the form $\Psi(x_1, \ldots, x_n) > 1$. To determine the probability p_f, the types and parameters of the probability distributions of $(X_1, \ldots, X_n)$ are needed. As opposed to sensitivity analysis, this requires detailed information about the statistical properties of the input parameters, obtained from experiments or previous studies.

The acceptable value of the failure probability depends on the circumstances. The civil engineering codes require that the designed structure obtains an instantaneous probability of failure of $p_f = 10^{-6}$ and a long-term failure probability of $p_f = 10^{-5}$. To credibly estimate tail probabilities of such a small magnitude, *a lot* of information is needed. In addition, if time dependent reliability is to be assessed, failure rates and the additional parameters of the time-dependent reliability function are required. The problems arising from this concept of failure probability have

been discussed at many places, including the codes themselves [15, Annex C4(3)], see also [16, 36] and references therein. In technological development phases in aerospace engineering, a failure probability in the range of $p_f = 10^{-3}$ may be acceptable, especially if it is not used as an absolute measure, but as an objective function in optimization (reliability based optimization).

Having performed a Monte Carlo sensitivity analysis of the model output $Y = g(X_1, \ldots, X_n)$, the question comes to mind if one could not use the generated sample for getting reliability estimates of the structure. This is indeed the case, albeit at a possibly low accuracy due to the small sample size of the sensitivity analysis. There are two ways of exploiting the existing sample. One way is by means of tolerance intervals to estimate credible upper and lower bounds for the output Y; another way is by reweighting the generated sample of $(X_1, \ldots, X_n)$ so as to mimic input distributions other than the uniform distributions used in the sensitivity analysis.

Tolerance Intervals While confidence intervals give an estimate for the distribution parameters λ of a random quantity Y, a tolerance interval gives an estimate of the range of possible observations of Y. More precisely, one wants to compute an interval $[a, b]$ that contains a certain proportion p, say $p = 90\%$, of the population with a given confidence level $1 - \alpha$, say $1 - \alpha = 95\%$. A non-parametric approach based on order statistics is especially attractive, since it is applicable without knowledge of the type of statistical distribution of Y. In fact, given whatever sample of whatever random variable, one may estimate the proportion p of the population that lies within the sample maximum $Y_{\max}$ (the largest value in the sample) and the sample minimum $Y_{\min}$ with a given confidence $1 - \alpha$, depending only on the sample size N. In this situation, the interval $[a, b] = [Y_{\min}, Y_{\max}]$ is given and N is known. Thus depending on the desired confidence level, the proportion p lying within the boundaries $[a, b]$ can be computed.

The derivation of a one-sided non-parametric tolerance interval with upper boundary the sample maximum is particularly easy, using only combinatorics. In fact, a proportion p of the population lies in the interval $(-\infty, Y_{\max}]$ with confidence $1 - \alpha$ if the relation

$$p^N = \alpha$$

holds. This can be seen as follows. Denote by Q_p the p-th quantile of Y. This means that $P(Y \leq Q_p) = p$. On the other hand, the interval $(-\infty, Y_{\max}]$ contains at least the proportion p of the population if $Q_p \leq Y_{\max}$. Thus it is required that

$$P(Q_p \leq Y_{\max}) \geq 1 - \alpha.$$

But $P(Q_p \leq Y_{\max}) = 1 - P(Y_{\max} < Q_p)$. Observe that $Y_{\max} < Q_p$ if and only if each of the N independent realizations of Y in the sample is below Q_p, i.e., $Y_j < Q_p$ for $j = 1, \ldots, N$. By definition, the probability of the event $Y_j < Q_p$ is exactly p. Collecting terms, one arrives at $1 - P(Y_{\max} < Q_p) = 1 - p^N$, whence the assertion.

Table 4.1 Required sample size N for one-sided tolerance intervals at confidence $1 - \alpha$

$1 - \alpha$	$p = 0.90$	$p = 0.95$	$p = 0.99$
0.90	22	45	230
0.95	29	59	299
0.99	44	90	459

From there, universally valid estimates of the sample size N required so that a proportion p of values lies under the sample maximum at confidence $1 - \alpha$ can be established, see e.g. Table 4.1. The same formula applies to one-sided intervals of the form $[Y_{\min}, \infty)$. Tolerance intervals with various proportions and confidence levels are tabulated, e.g., in the ISO standard [22]. For the theory, see, e.g., [27].

Monte Carlo Reweighting As outlined in Sect. 4.2, the goal of Monte Carlo simulation is to estimate expectation values $E(h(Y)) = E(h(g(X_1, \ldots, X_n)))$ of functions of the model output. Suppose we have already generated a sample $(x_{ji}, \ldots, x_{jn})$ and computed the outputs y_j, $j = 1, \ldots, N$, where the sample has been generated according to a certain probability distribution of the input, say with probability density $f(x_1, \ldots, x_n)$. Is it possible to use the same sample to estimate $E(h(\tilde{Y})) = E(h(g(\tilde{X}_1, \ldots, \tilde{X}_n)))$ where the $\tilde{X}_j$ are random variables defined on the same range as the X_j, but with another probability distribution, say with probability density $\varphi(x_1, \ldots, x_n)$? To understand the positive answer, it is useful to write the expectation $E(h(\tilde{Y}))$ as an integral:

$$
\begin{aligned}
E(h(g(\tilde{X}_1, \ldots, \tilde{X}_n))) &= \int \cdots \int h(g(x_1, \ldots, x_n)) \varphi(x_1, \ldots, x_n) \, dx_1 \cdots dx_n \\
&= \int \cdots \int h(g(x_1, \ldots, x_n)) \frac{\varphi(x_1, \ldots, x_n)}{f(x_1, \ldots, x_n)} \\
&\quad \times f(x_1, \ldots, x_n) \, dx_1 \cdots dx_n \\
&= E\left(h(g(X_1, \ldots, X_n)) \frac{\varphi(X_1, \ldots, X_n)}{f(X_1, \ldots, X_n)} \right).
\end{aligned}
$$

This shows that the computation of the new expectation value can be accomplished by computing the old expectation value of the input–output function, multiplied by a weight—the quotient of the two densities. In terms of Monte Carlo simulation, one has to compute

$$
E(h(\tilde{Y})) \approx \frac{1}{N} \sum_{j=1}^{N} h(g(x_{ji}, \ldots, x_{jn})) \frac{\varphi(x_{ji}, \ldots, x_{jn})}{f(x_{ji}, \ldots, x_{jn})}.
$$

This causes no additional effort, because one can reuse the expensive computation of $h(g(x_{ji}, \ldots, x_{jn}))$. The accuracy of the method depends on the degree of similarity of the old and the new distribution.

A typical application could consist in reusing the sample from the sensitivity analysis, based on uniform distributions, and place truncated Gaussians on the intervals. This concludes the remarks about what can be extracted from the sensitivity analysis towards a reliability assessment.

Importance Sampling When estimating failure probabilities of low value, one cannot expect that the rather small sample sizes of sensitivity analysis suffice. In fact, a standard Monte Carlo estimate of the failure probability is of the form

$$p_f \approx \frac{1}{N} \sum_{j=1}^{N} \chi^{\mathscr{F}}(x_{j1}, \dots, x_{jn})$$

where $\chi^{\mathscr{F}}$ equals one if $(x_{j1}, \dots, x_{jn}) \in \mathscr{F}$ and zero otherwise. It thus can be seen as the mean value of an N-fold repetition of a zero-one experiment with success probability p_f. The variance of the Monte Carlo estimator of p_f is hence given by $p_f(1 - p_f)/N$, whence the mean estimation error is approximately equal to $\sqrt{p_f/N}$. This means that an accuracy of $\alpha \cdot 100\,\%$ requires a sample size of $N \approx 1/(\alpha p_f)$. Consequently, cost-saving methods need to be devised, two of which shall be discussed here.

The first one is *importance sampling*. As seen above, the probability of failure is estimated by counting the number of realizations of the input variables $(X_1, \dots, X_n)$ that fall into the failure region. Since this number is expected to be small compared to the total number of simulated points, the probability density $f(x_1, \dots, x_n)$ of the input variables will be small on $\mathscr{F}$. The idea of importance sampling is to generate a sample of another distribution, say with probability density $\varphi(x_1, \dots, x_n)$, which may be concentrated in the region $\mathscr{F}$. The idea is similar to Monte Carlo reweighting, but this time a different sample is produced to begin with. In fact,

$$p_f = \mathrm{E}\big(\chi^{\mathscr{F}}(X_1, \dots, X_n)\big) = \int \cdots \int \chi^{\mathscr{F}}(x_1, \dots, x_n) f(x_1, \dots, x_n)\, \mathrm{d}x_1 \cdots \mathrm{d}x_n$$

$$= \int \cdots \int \chi^{\mathscr{F}}(x_1, \dots, x_n) \frac{f(x_1, \dots, x_n)}{\varphi(x_1, \dots, x_n)}\, \varphi(x_1, \dots, x_n)\, \mathrm{d}x_1 \cdots \mathrm{d}x_n$$

$$= \mathrm{E}\Big(\chi^{\mathscr{F}}(\tilde{X}_1, \dots, \tilde{X}_n) \frac{f(\tilde{X}_1, \dots, \tilde{X}_n)}{\varphi(\tilde{X}_1, \dots, \tilde{X}_n)}\Big)$$

where the random variables $(\tilde{X}_1, \dots, \tilde{X}_n)$ have the probability density $\varphi(x_1, \dots, x_n)$. This leads to the following prescription. First, choose a probability density function $\varphi(x_1, \dots, x_n)$ concentrated in the failure region. Next, generate a random sample $(z_{ji}, \dots, z_{jn})$ according to the corresponding probability distribution. Finally, estimate the failure probability by

$$p_f \approx \frac{1}{N} \sum_{j=1}^{N} \chi^{\mathscr{F}}(z_{ji}, \dots, z_{jn}) \frac{f(z_{ji}, \dots, z_{jn})}{\varphi(z_{ji}, \dots, z_{jn})}.$$

Of course, this begs the question how to find a probability density $\varphi(x_1, \ldots, x_n)$ concentrated in the failure region. After all, the failure region unfolds itself only *after* Monte Carlo evaluation of the limit state function. Various proposals have been made in this respect, notably Bucher's adaptive sampling [7, 9]. This method starts with a pilot simulation with increased variance of the input variables (to rapidly produce a number of points in the failure region) and then uses certain shifted Gaussians as weight functions.

An interesting proposal has recently been made by Schwarz [49], developed for the situation where the input variables are supported in intervals (as, e.g., the uniform distributions used in sensitivity analysis or transformations thereof). There are two ingredients. First, one may expect that—in most cases—the failure region is concentrated in points near the boundaries of the intervals. Second, input variables with a larger influence on the output should have a larger weight. The starting point of the procedure is a sensitivity analysis with moderate sample size, from which the most important input variables and their correlation coefficients with the output are determined. Then a parametrized family of weight functions is placed on the input intervals, where the weight is just 1 for the unimportant parameters and is shifted more and more towards the boundaries of the input intervals, the larger the correlation.

Subset Simulation The subset simulation method was introduced by Au and Beck in [3]. The idea is to approximate the failure region $\mathscr{F}$ by a sequence of larger regions $\mathscr{F} = \mathscr{F}_m \subset \mathscr{F}_{m-1} \subset \ldots \subset \mathscr{F}_1 \subset \mathscr{F}_0$ and to compute the failure probability by a product of conditional probabilities

$$p_f = P(\mathscr{F}) = P(\mathscr{F}_m | \mathscr{F}_{m-1}) P(\mathscr{F}_{m-1} | \mathscr{F}_{m-2}) \ldots P(\mathscr{F}_1 | \mathscr{F}_0) P(\mathscr{F}_0)$$

where $\mathscr{F} = \mathscr{F}_m$ and $\mathscr{F}_0$ is the starting region. These conditional probabilities are appreciably larger than the failure probability and hence easier to simulate with smaller samples. In this case it is useful to describe the failure region by means of a ratio based limit state function $\mathscr{F} = \{(x_1, \ldots, x_n) : \Psi(x_1, \ldots, x_n) > 1\}$. The intermediate regions are chosen of the form $\mathscr{F}_i = \{(x_1, \ldots, x_n) : \Psi(x_1, \ldots, x_n) > \alpha_i\}$ with $0 < \alpha_0 < \alpha_1 < \ldots < \alpha_m = 1$. In fact, the choice of α_i is often made during the simulation such that $P(\mathscr{F}_i | \mathscr{F}_{i-1})$ has a fixed value, say p_0 between 0.1 and 0.3, and the regions $\mathscr{F}_i$ are constructed recursively. The conditional distribution $P(\cdot | \mathscr{F}_i)$ is just the original distribution, restricted to $\mathscr{F}_i$ and scaled by $P(\mathscr{F}_i)$. The latter probability is unknown a priori. This suggests to use the Metropolis–Hastings algorithm, a Markov chain Monte Carlo algorithm, which requires knowledge of the sampling distribution only up to a multiplicative factor (see below).

In the sequel, it is assumed that at each level i, a sample of size N is generated. The algorithm is initiated by generating a sample of the original distribution using standard Monte Carlo simulation. From this sample, the worst $p_0 \cdot 100\%$ realizations are declared to belong to $\mathscr{F}_0$. More precisely, the sample is ordered according to the Ψ-values. The threshold level α_0 is chosen as the $(1 - p_0)N$-th largest value among the Ψ-values attained in the sample, and $\mathscr{F}_0$ is defined to be the set

$\{(x_1, \ldots, x_n) : \Psi(x_1, \ldots, x_n) > \alpha_0\}$. Further, $P(\mathscr{F}_0) = p_0$. In the next step, one or more of the points in $\mathscr{F}_0$ is/are chosen as the initial point (root) of a Markov chain whose elements are distributed according to $P(\cdot|\mathscr{F}_0)$, this way generating a second sample of size N. Again, the worst $p_0 \cdot 100\,\%$ realizations are declared to belong to $\mathscr{F}_1$, and α_1 is chosen as the $(1 - p_0)N$-th largest value among the Ψ-values attained in the second sample, and so on. The simulation stops at the first level m at which the Ψ-values of the worst $p_0 \cdot 100\,\%$ bigger than 1. At this stage $P(\mathscr{F}_m|\mathscr{F}_{m-1})$ is estimated by M/N where M is the number of failed realizations in the last sample. Finally, p_f is estimated as $p_0^m \cdot M/N$.

Before discussing some details of the algorithm, a short introduction to Markov chain Monte Carlo simulation is in order. The ideas can be best explicated at the hand of the original Metropolis algorithm for simulating a one-dimensional distribution $\pi(x)$. The goal is to generate a realization $\xi_0, \xi_1, \xi_2, \ldots, \xi_N$ of a Markov chain whose stationary distribution is $\pi(x)$. Since the chain will converge to the stationary distribution as $N \to \infty$, the end-pieces $\xi_M, \ldots, \xi_N$ are approximately distributed according to π (for large M and N).

The algorithm proceeds as follows. Choose a transition kernel $q(x, y)$ *(proposal distribution)* such that $q(x, y) = q(y, x)$ for all x, y. (Often it is taken of the form $q(x, y) = p(x - y)$ where p is a nowhere vanishing probability density.) Choose an initial distribution $p^{(0)}(x)$.

- Sample a value ξ_0 from $p^{(0)}$.
- For $k = 1, \ldots, N$

  Sample a value η from the proposal distribution $q(\xi_{k-1}, \cdot)$.

  Compute the ratio $r = \dfrac{\pi(\eta)}{\pi(\xi_{k-1})}$.

  – If $r \geq 1$, the value η is accepted; set $\xi_k = \xi_{k-1}$.
  – If $r < 1$, the value η is accepted with probability r and rejected with probability $1 - r$.

  Draw a random number ζ from the uniform distribution on $[0, 1]$.

  – If $\zeta \leq r$, set $\xi_k = \eta$.
  – If $\zeta > r$, set $\xi_k = \xi_{k-1}$.

- $\xi_0, \xi_1, \ldots, \xi_N$ has $\pi(x)$ as limiting distribution as $N \to \infty$.

Observe that only the ratio r enters in the computation, so knowledge of $\pi(x)$ is only required up to a multiplicative factor. The Metropolis–Hastings algorithm is similar, but the proposal distribution is not required to be symmetric. More theory can be found in [42].

If the distribution $\pi(x_1, \ldots, x_n)$ is multidimensional, it is advantageous to change only one coordinate in each step in order to keep the number of rejected trials at a moderate level. In its application to subset simulation, the target distribution is $P(\cdot|\mathscr{F}_{i-1})$ in each level. Thus one has to check whether $\xi_k \in \mathscr{F}_{i-1}$ for acceptance, in addition. By construction, each root of the generated Markov chain is already distributed according to $P(\cdot|\mathscr{F}_{i-1})$. It can be shown [3] that the elements

of the whole chain at level i are not only asymptotically but also perfectly distributed according to $P(\cdot|\mathscr{F}_{i-1})$. As can be seen, there are a lot of screws that can be adjusted: choice of the proposal distribution, optimal acceptance/rejection rate, number and length of chains generated in level i—with the possibility of parallelization, choice of p_0, and so on. Based on many recommendations to be found in the literature, subset simulation has developed into an efficient method for estimating failure probabilities. A critical comparison of various simulation methods in reliability analysis can be found in [47, 48].

4.6 Application

This section is devoted to demonstrating the methods at work in a practical application from aerospace engineering, namely in a finite element model of the frontskirt of the ARIANE 5 launcher. The frontskirt is the part of the launcher that connects the tanks section with the payload section and also has to support the booster loads. It consists of a light weight shell structure reinforced by struts. The full finite element model is composed of shell elements and solid elements, altogether with two million degrees of freedom. The models have been supplied by Intales GmbH Engineering Solutions (see Footnote 1). Models of varying complexity and material properties with up to 130 input and a similar number of output parameters have been analyzed.

For the sake of presentation, we shall focus on a smaller finite element model keeping the global structure with about ninety thousand degrees of freedom. Figure 4.1 depicts the model schematically; it is composed of two hemispheres and three cylinders, one of which is made up of composite material. The shadings indicate thickness variations of the tank skin, described by a random field. Booster loads are introduced at two opposite locations in the upper cylinder (not shown in Fig. 4.1). A selection of seventeen input parameters (all loads characterizing various flight scenarios) will be considered; their meaning is described in Table 4.2. As a representative output we start with the load proportionality factor (LPF), a decisive variable indicating buckling failure. It is defined as the limiting value in an incremental procedure in which the mechanical loads during a flight scenario are increased step by step until breakdown of the structure is reached. In the full model, the LPF is computed by means of a path following procedure that follows bifurcations until material failure occurs. In the simplified computations presented here, no distinction of bifurcation or material failure was made, so that the terminal value of the LPF was taken as that value at which the finite element program failed to converge. What concerns the computational effort, a single run of the input–output function computing the LPF in Abaqus takes around 1 h on a personal computer and 10 min on the supercomputer Leo-III of the University of Innsbruck.

Sensitivity Analysis As a basis for the sensitivity analysis, a sample of size $N = 100$ of the $n = 17$ input parameters was generated. Each parameter was taken
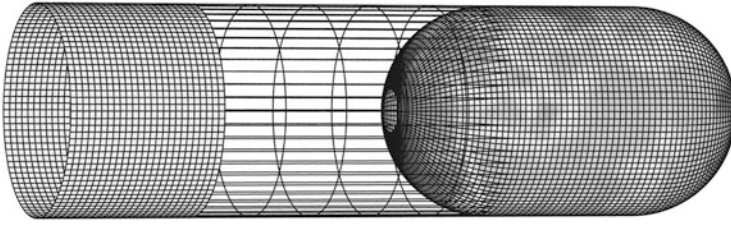
Fig. 4.1 Simplified finite element model of frontskirt; *shadings* show random field (thickness of tank skin) [12]

**Table 4.2** Description of input parameters no. 1–17 with their nominal values

i	Parameter X_i	Mean μ_i
1	Initial temperature	293 K
2	Step1 thermal loading cylinder1	450 K
3	Step1 thermal loading cylinder2	350 K
4	Step1 thermal loading cylinder3	150 K
5	Step1 thermal loading sphere1	150 K
6	Step1 thermal loading sphere2	110 K
7	Step2 hydrostatic pressure cylinder3	0.4 MPa
8	Step2 hydrostatic pressure sphere1	0.4 MPa
9	Step2 hydrostatic pressure sphere2	0.4 MPa
10	Step3 aerodynamic pressure	−0.05 MPa
11	Step4 booster loads y-direction node1	40,000 N
12	Step4 booster loads y-direction node2	20,000 N
13	Step4 booster loads z-direction node1	$3.e6$ N
14	Step4 booster loads z-direction node2	$1.e6$ N
15	Step4 mechanical loads x-direction	100 N
16	Step4 mechanical loads y-direction	50 N
17	Step4 mechanical loads z-direction	300 N

**Table 4.3** Sample statistics of simulated load proportionality factor

Mean	Minimum	Maximum	Standard deviation	Spread
3.5335	3.4468	3.6457	0.1989	±3 %

uniformly distributed around its nominal value listed in Table 4.2 with a spread of ±15 %. Latin hypercube sampling and correlation control was employed. The statistics of the computed LPF are listed in Table 4.3.

The scatterplot of Fig. 4.2 gives a first impression of the influence of the input variables on the output. It is quite clearly seen that input parameter no. 13 (booster loads) exerts a big influence, whereas the diagrams for the other parameters are less conclusive. To quantify the influences, the six different correlation indices CC, PCC, SRC, RCC, PRCC, SPRC were computed. (A complete list of the values has been published in [38].) As an example, the PRCCs of the 17 input parameters vs. the LPF are visualized in Fig. 4.3 (left).
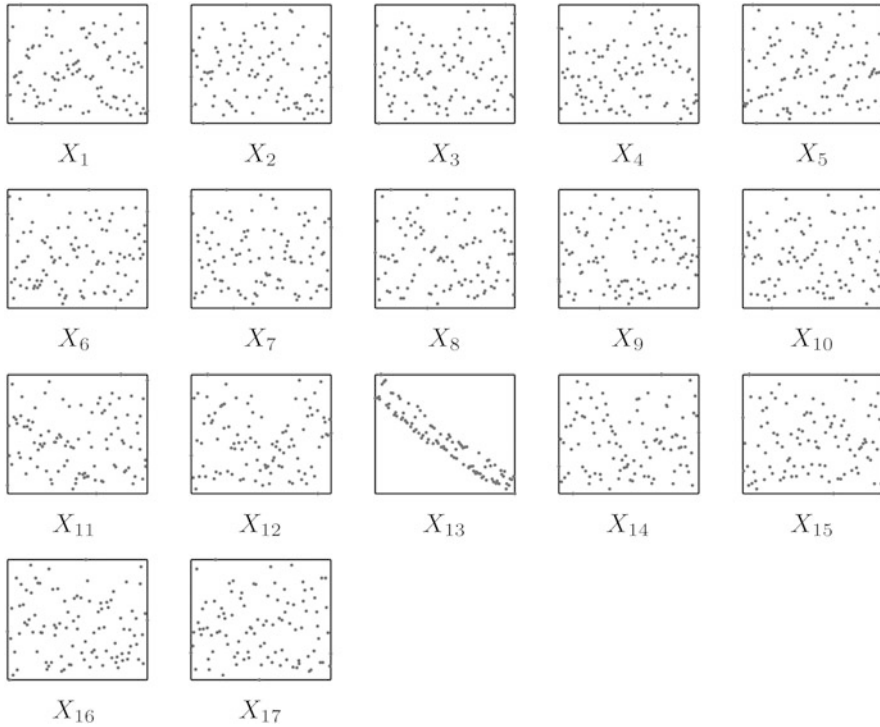
Fig. 4.2 *Scatterplots* of 17 input variables vs. output (LPF) [24, 38]
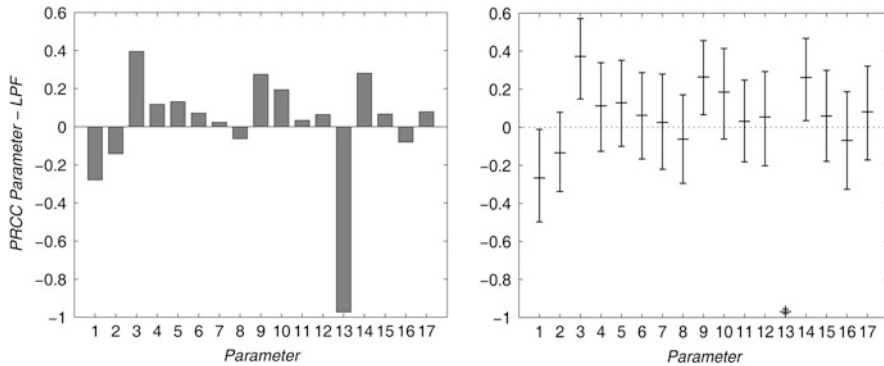
Fig. 4.3 Partial rank correlation coefficients (*left*) and 95 % bootstrap confidence intervals (*right*) [24, 38]

The resulting sensitivity indices induce a ranking of the input parameters according to their influence on the output. Note that the accuracy of the estimate for the correlation coefficients is in the range of $1/\sqrt{N} \cdot 100\,\% = 10\,\%$; this suffices to determine the ranking and confirms the observation that one can get along

Table 4.4 Ranks of the significant input parameters X_i according to the six measures of correlation input–output

| i	CC	PCC	SRC	RCC	PRCC	SRCC
1		3	3		4	4
3		2	2		2	2
9					5	5
13	1	1	1	1	1	1
14		4	4		3	3

with small sample sizes for an assertive sensitivity analysis. To check whether the computed correlation indices are significantly different from zero, bootstrap 95%-confidence intervals were computed (with bootstrap sample size $B = 5{,}000$). As a basis for an overall assessment of the ranking, only those sensitivity estimates with a resultant confidence interval not including 0 have been regarded as significant. As an example, bootstrap confidence intervals for the PRCCs are displayed in Fig. 4.3 (right). Accordingly, only the PRCCs of the parameters X_1, X_3, X_9, X_{13}, and X_{14} test to be nonzero. The ranks of those parameters that tested to be significant according to at least one of the six indices are listed in Table 4.4. The table gives a good impression of the sensitivity assessment—if a single scale is required, one might use the average ranks.

Coefficient of Determination As discussed in Sects. 4.2 and 4.4, metamodels can be used to further quantify the influence of selected parameters. One way to achieve this is to fit a linear regression model

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_n X_n$$

and then to compute the partial coefficients of determination. The sequential partial coefficient of determination of variable X_i is computed by first fitting the model $Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_n X_{i-1}$, then adjoining the variable X_i and recording the increase in the coefficient of determination R^2. Averaging all partial coefficients of determination which can be obtained by adding the variable X_i to all possible combinations of the already included variables leads to a non-parametric measure of the contribution of variable X_i to the *explanatory power* of the model. The procedure is explained, e.g., in [28, 39]. The result can be conveniently displayed in the form of a pie chart. As an example, a linear model for the *LPF* has been set up with input parameters nos. 1, 3, 4, 7, 9, 13, 14. It resulted in the assessment of the influences depicted in Fig. 4.4, taken from [43]. The eminent influence of parameter X_{13} is once more confirmed. In the figure, the label *Res* refers to the residual proportion which remains unexplained through the linear model.

Random Fields In order to investigate the effect of geometric and material imperfections, the thickness, modulus of elasticity and yield stress were disturbed by two-dimensional random fields with an autocovariance function of the form (4.1). The distance function on the cylinders was taken as the sum of the axial and radial distance, whereas on the spheres, it was taken as the sum of the latitude and altitude, multiplied by the radius. A routine extracting the distances from the finite
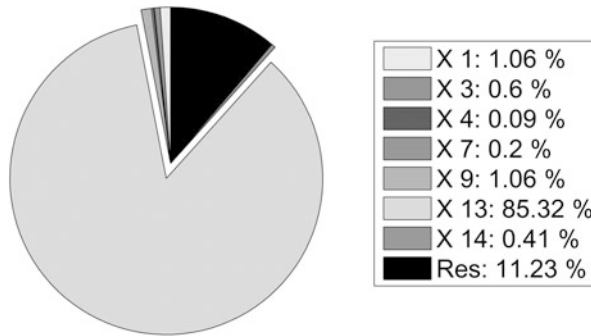
Fig. 4.4 Pie chart of relative explanatory power of LPF through various input variables [43]
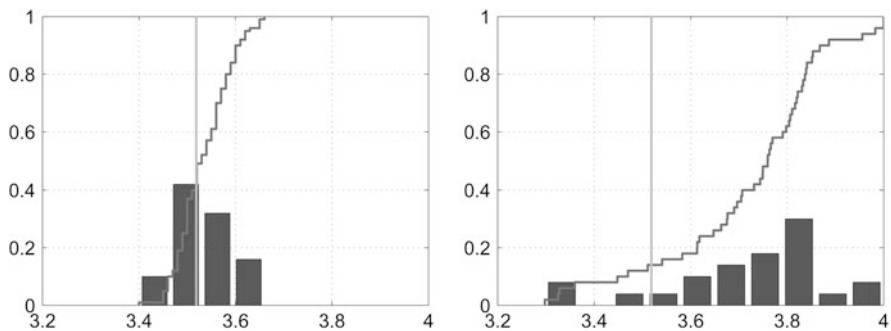
Fig. 4.5 Plots of histogram and cumulative distribution function of LPF produced by random loads, no imperfections (*left*) and with random field turned on (*right*). *Vertical line* indicates LPF corresponding to nominal input values [41]

element grid was implemented by [41]. The nominal values were 1 mm (thickness), 70,000 MPa (modulus of elasticity), 320 MPa (yield stress) for the first sphere, with similar values for the other components of the frontskirt. A coefficient of variation of 10 % was applied throughout. In a parametric study, the correlation lengths were varied between 60 mm (corresponding to the dimension of two elements of the grid) and 1,600 mm, with various combinations in the two respective directions. Random fields were generated by means of the Karhunen–Loève expansion.

In a first investigation, the different effects of the random imperfections and the 17 random loads on the LPF were studied. Figure 4.5 shows one of the results, with a sample size of $N = 100$ both for loads and realizations of the random fields. The correlation lengths were set to 188.5 mm in all angular directions (cylinders and spheres), while the correlation length in axial direction was taken 450 and 900 mm for the cylinders. In the left figure, the distribution of the LPF is shown with the random loads from the sensitivity analysis, but with material and geometric properties kept at their nominal value. In the right figure, the random field is turned on, in addition. One can see that the random field has a stabilizing effect (larger
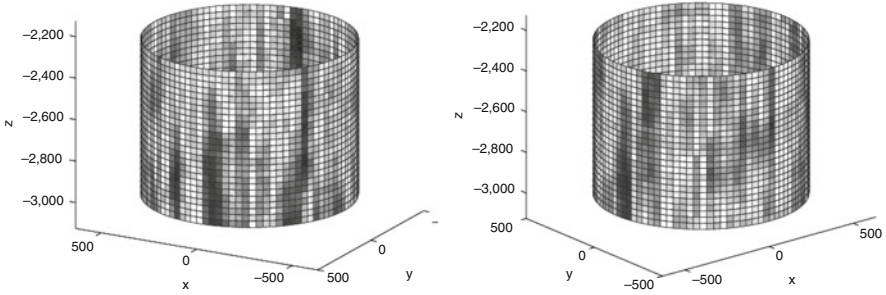
Fig. 4.6 Spatial distribution of influence (measured by CC) of element-wise yield stress (*left*) and thickness (*right*) on the LPF, based on random field simulation [41]

Table 4.5 Proportion p of LPFs lying above $LPF_{min} = 3.4468$ at confidence level $1 - \alpha$

$1 - \alpha$	90 %	95 %	99 %	99.9 %
p	97.7 %	97.1 %	95.5 %	93.3 %

values of LPF are attained), but at the same time increases the uncertainty of the outcome.

Interestingly, the application of random fields admits a structurally localized study of the correlations. After all, a realization of the random field induces a realization of the modelled quantity (e.g., thickness) in each element of the FE-grid. Thus one can do a standard sensitivity analysis with these variables. For example, cylinder no. 3 has 2,500 elements; the random field simulation produces $N = 100$ realizations of the 2,500 grid values of the thickness, elasticity modulus, yield stress. The spatial distribution of the influence on the LPF can thus be visualized. Figure 4.6 shows such a distribution in terms of the Pearson CC for the yield stress (left) and the thickness (right). In the same way, localized correlations of different output variables can be pictured.

Tolerance Intervals As a first step from sensitivity to reliability, tolerance intervals for the LPF can be established. Recall from Table 4.3 that the minimal sampled LPF was at $LPF_{min} = 3.4468$. Based on the formula $p^N = \alpha$ with $N = 100$, one can assess the proportions p of the possibly attainable LPF-values with a given confidence $1 - \alpha$. The results are summarized in Table 4.5.

Reliability Analysis In order to test various simulation methods for reliability, a benchmark study of the small launcher model was undertaken by [49], from where all results and tables in this paragraph are taken. For this study, an extended list of 35 input parameters was used. As a limit state function, a combination of allowable limits in the equivalent plastic strain (PEEQ), principal stress in the composite part (SP), and the absolute value of the smallest eigenvalue (EV) was employed:

$$\Psi(\mathbf{x}) = \max \left\{ \frac{PEEQ(\mathbf{x})}{0.07}, \frac{SP(\mathbf{x})}{180}, \frac{0.001}{EV(\mathbf{x})} \right\}$$

with failure defined by $\Psi(\mathbf{x}) > 1$, $\mathbf{x} = (x_1, \ldots, x_{35})$. The three criteria correspond to plastification in the metallic part, rupture in the composite part, and buckling of the structure.

A reference brute force Monte Carlo simulation of size $N = 5,000$ was undertaken on the supercomputer Leo-III, with three strands in parallel. As in the previous sensitivity analysis, the 35 input variables were taken uniformly distributed on an interval of spread $\pm 15\%$ around their nominal values. The resulting failure probability turned out to be $p_f = 0.0116$. A 95% bootstrap confidence interval was computed (bootstrap sample size $B = 5,000$) as $[0.0088, 0.0146]$. A sensitivity analysis with the sample revealed that only 10 of the 35 parameters had a significant influence on the failure criterion Ψ, measured at a 90% confidence level.

Next, an investigation was undertaken whether an estimate in the same range could be obtained with a smaller sample size by subset simulation or by importance sampling. Subset simulation was undertaken with $p_0 = 0.2$ (see Sect. 4.5). Since $p_0^3 = 0.008$ is already smaller than the intended failure probability, three levels $\mathscr{F}_0, \mathscr{F}_1, \mathscr{F}_2$ suffice for the subset simulation algorithm. Experiments were undertaken with sample size $N = 900$ and $N = 300$ for each level. Recall that 20% of the generated points 20% of the generated points of $\mathscr{F}_{i-1}$ are assigned to $\mathscr{F}_i$ in each step. Since these points can be reused when going from level $i - 1$ to level i, the total number of generated points in the three levels is $2,340$ and 780, respectively. Further, it was tested whether including all 35 input influential ones in the simulation changes the value of the failure probability. In addition, bootstrap confidence intervals for the failure probability were computed.

To keep results comparable, the importance sampling procedure was done with a sample size $N = 780$. The method described in Sect. 4.5 was employed, by which the weights are computed in dependence on the magnitude of the correlation coefficient of the respective input parameter with the Ψ-value. This required the actual simulation to be preceded by a sensitivity analysis. The sensitivity analysis was done with a sample of size 99, so that a sample size of 681 remained for the importance sampling part. Correlation control was employed when simulating the input data. It was tested whether weighting of all parameters or weighting only the parameters significant at the 90% level changes the outcome.

The joint results are recorded in Table 4.6. Here NR refers to the total number of realizations computed in the simulation, NV denotes the number of activated variables (subset simulation), respectively weighted variables (importance sampling); p_f is the failure probability and 95% BSL/BSU refers to the lower/upper bound of the 95% bootstrap confidence interval for p_f.

We conclude this section by reporting on a reweighting experiment. As a basis, the sample of size $N = 5,000$ of the reference Monte Carlo simulation was taken. The uniformly distributed input was replaced, using reweighting, by truncated Gaussians. The mean values of the Gaussian distributions were taken as the interval midpoints, the variance was computed from assumed coefficients of variation (between 7.5 and 15%), the truncation was effected at the interval endpoints. The change from uniform distributions to mid-pieces of Gaussians resulted in quite a

Table 4.6 Comparison of results of six different simulation procedures for the failure probability [49]

	Monte Carlo	Subset	Subset	Subset	Importance	Importance
NR	5,000	2,340	780	780	780	780
NV	35	35	35	10	35	10
p_f	0.0116	0.0130	0.0155	0.0120	0.0108	0.0124
95 % BSL	0.009	0.010	0.010	0.008	0.006	0.007
95 % BSU	0.015	0.016	0.022	0.017	0.019	0.019

change of the failure probability, namely to $p_f = 0.0019$ with a 95 % bootstrap confidence interval $[0.001, 0.003]$.

4.7 Conclusion

The purpose of this chapter was twofold. On the one hand, it served to describe current core methods of Monte Carlo simulation, from design of experiment, random fields, metamodelling to concepts of sensitivity and reliability analysis. On the other hand, the chapter demonstrated the implementation of those methods in joint research projects with Intales GmbH Engineering Solutions over the past years.

A number of themes have deliberately not been addressed in order to keep the presentation concise. These include simulation of correlated input using copulas [33], also implemented in the mentioned projects [44], Bayesian methods of reliability analysis [31], Bayesian estimates of the distribution of the failure probability [49, 54], and optimization for finding worst case parameter combinations [12]. Further, the discussion of asymptotic sampling [8], though implemented in our toolbox [49], was omitted because its presentation would have required to go into some details about the safety index and FORM (the first order reliability method).

References

1. Aichinger, C.: Monte Carlo methods in iterative solvers. Diploma thesis, University of Innsbruck, Austria (2010)
2. Arnold, L.: Stochastic Differential Equations: Theory and Applications. Wiley, New York (1974)
3. Au, S.-K., Beck, J.L.: Estimation of small failure probabilities in high dimensions by subset simulation. Probab. Eng. Mech. **16**, 263–277 (2001)

4. Beer, M., Ferson, S., Kreinovich, V.: Imprecise probabilities in engineering analysis. Mech. Syst. Signal Process. **37**, 4–29 (2013)
5. Bhatia, R.: Perturbation Bounds for Matrix Eigenvalues. Longman, Harlow (1987)
6. Bolotin, V.V.: Statistical Methods in Structural Mechanics. Holden-Day, San Francisco (1969)
7. Bucher, C.: Adaptive sampling: an iterative fast Monte Carlo procedure. Struct. Saf. **5**, 119–126 (1988)
8. Bucher, C.: Asymptotic sampling for high-dimensional reliability analysis. Probab. Eng. Mech. **24**, 504–510 (2009)
9. Bucher, C.: Computational Analysis of Randomness in Structural Mechanics. CRC Press/Balkema, Leiden (2009)
10. De Groof, V., Oberguggenberger, M., Haller, H., Degenhardt, R., Kling, A.: Quantitative assessment of random field models in finite element buckling analyses of composite cylinders. In: Eberhardsteiner, J., Böhm, H.J., Rammerstorfer, F.G. (eds.) CD-ROM Proceedings of the 6th European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS 2012), Vienna University of Technology, Wien (2012)
11. De Groof, V., Oberguggenberger, M., Haller, H., Degenhardt, R., Kling, A.: A case study of random field models applied to thin-walled composite cylinders in finite element analysis. In: Deodatis, G., Ellingwood, B.R., Frangopol, D.M. (eds.) Safety, Reliability, Risk and Life-Cycle Performance of Structures and Infrastructures, p. 379. CRC Press/Balkema, Leiden (2013)
12. De Groof, V., Oberguggenberger, M., Prackwieser, M., Schwarz, M.: Reliability analysis of shell structures. In: Barden, M., Ostermann, A. (eds.) Scientific Computing @ uibk, pp. 39–42. Innsbruck University Press, Innsbruck (2013)
13. Dick, J., Pillichshammer, F.: Digital Nets and Sequences: Discrepancy Theory and Quasi-Monte Carlo Integration. Cambridge University Press, Cambridge (2010)
14. Efron, B., Tibshirani, R.J.: An Introduction to the Bootstrap. Chapman and Hall, New York (1993)
15. European Committee for Standardization: EN 1990:2002. Eurocode: Basis of Structural Design. CEN, Brussels (2002)
16. Fellin, W., Lessmann, H., Oberguggenberger, M., Vieider, R.: Analyzing Uncertainty in Civil Engineering. Springer, Berlin (2005)
17. Freudenthal, A.N.: Safety and the probability of structural failure. Trans. ASCE **121**, 1337–1397 (1956)
18. Ghanem, R.G., Spanos, P.D.: Stochastic Finite Elements: a Spectral Approach. Springer, New York (1991)
19. Graham, C., Talay, D.: Stochastic Simulation and Monte Carlo Methods: Mathematical Foundations Of Stochastic Simulations. Springer, Berlin (2013)
20. Helton, J.C., Johnson, J.D., Sallaberry, C.J., Storlie, C.B.: Survey of sampling-based methods for uncertainty and sensitivity analysis. Reliab. Eng. Syst. Saf. **91**, 1175–1209 (2006)
21. Iman, R.L., Conover, W.J.: A distribution-free approach to inducing rank correlation among input variables. Commun. Stat. Simul. Comput. **11**, 311–334 (1982)
22. International Standard: ISO 16269-6:2005. Statistical Interpretation of Data: Part 6: Determination of Statistical Tolerance Intervals. ISO, Geneva (2005)
23. Janouchová, E., Kučerová, A.: Competitive comparison of optimal designs of experiments for sampling-based sensitivity analysis. Comput. Struct. **124**, 47–60 (2013)
24. King, J.: Stochastic simulation methods in sensitivity analysis. Diploma thesis, University of Innsbruck, Austria (2007)
25. Kleijnen, J.P.C.: Design and Analysis of Simulation Experiments. Springer Science+Business Media LLC, New York (2008)
26. Kloeden, P.E., Platen, E.: Numerical Solution of Stochastic Differential Equations. Springer, Berlin (1992)
27. Krishnamoorthy, K., Mathew, T.: Statistical Tolerance Regions: Theory, Applications and Computation. Wiley, New Jersey (2009)
28. Kruskal, W.: Relative importance by averaging over orderings. Am. Stat. **41**, 6–10 (1987)

29. Le Maître, O., Knio, O.: Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics. Springer, New York (2010)
30. Loève, M.: Probability Theory, vol. II, 4th edn. Springer, New York (1978)
31. Martz, H.F., Waller, R.A.: Bayesian Reliability Analysis. Wiley, Chichester (1982)
32. Montgomery, D.C., Peck, E.A., Vining, G.G.: Introduction to Linear Regression Analysis, 5th edn. Wiley, New York (2012)
33. Nelsen, R.B.: An Introduction to Copulas, 2nd edn. Springer, New York (2006)
34. Niederreiter, H.: Random number generation and quasi-Monte Carlo methods. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (1992)
35. Oberguggenberger, M.: Analysis and computation with hybrid random set stochastic models. In: Deodatis, G., Ellingwood, B.R., Frangopol, D.M. (eds.) Safety, Reliability, Risk and Life-Cycle Performance of Structures and Infrastructures, p. 93. CRC Press/Balkema, Leiden (2013)
36. Oberguggenberger, M.: Combined methods in nondeterministic mechanics. In: Elishakoff, I., Soize, C. (eds.) Nondeterministic Mechanics, pp. 263–356. Springer, Wien (2013)
37. Oberguggenberger, M., Aichinger, C., Caillaud, B., Haller, H., Roth, H.: Simulation tools for assessing the reliability and the design of shell structures. CD-ROM. In: Conference Proceedings, 4th International Conference on "Supply on the Wings", Frankfurt. AIRTEC Frankfurt, Paper No. D13 (2009)
38. Oberguggenberger, M., King, J., Schmelzer, B.: Classical and imprecise probability methods for sensitivity analysis in engineering: a case study. Int. J. Approx. Reason. **50**, 680–693 (2009)
39. Oberguggenberger, M., Ostermann, A.: Analysis for Computer Scientists: Foundations, Methods, and Algorithms. Springer, London (2011)
40. Ostermann, A.: Sensitivity analysis. In: Fellin, W., Lessmann, H., Oberguggenberger, M., Vieider, R. (eds.) Analyzing Uncertainty in Civil Engineering, pp. 101–114. Springer, Berlin (2005)
41. Riedinger, K.: Simulation of random fields for sensitivity analysis. Diploma thesis, University of Innsbruck, Austria (2010)
42. Robert, C.P., Casella, G.: Monte Carlo Statistical Methods, 2nd edn. Springer, New York (2004)
43. Roth, H.: Partial coefficient of determination. Internal Report, University of Innsbruck, Austria (2010)
44. Roth, H.: Sensitivity analysis with correlated variables. Diploma thesis, University of Innsbruck, Austria (2010)
45. Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., Tarantola, S.: Global Sensitivity Analysis: The Primer. Wiley, Chichester (2008)
46. Schenk, C.A., Schuëller, G.I.: Uncertainty Assessment of Large Finite Element Systems. Springer, Berlin (2005)
47. Schuëller, G.I. (ed.): A benchmark study on reliability in high dimensions. Special Issue. Struct. Saf. **29**, 165–252 (2007)
48. Schuëller, G.I., Pradlwarter, H.J., Koutsourelakis, P.S.: A critical appraisal of reliability estimation procedures for high dimensions. Probab. Eng. Mech. **19**, 463–474 (2004)
49. Schwarz, M.: Monte Carlo based reliability analysis. Master thesis, University of Innsbruck, Austria (2013)
50. Shao, J., Tu, D.-S.: The Jackknife and Bootstrap. Springer, New York (1995)
51. Sobol, I.M.: Sensitivity analysis for nonlinear mathematical models. Math. Model. Comput. Experiment **1**, 407–414 (1993)
52. Sobol, I.M.: Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Math. Comput. Simul. **55**, 271–280 (2001)
53. Stein, M.: Large sample properties of simulations using Latin hypercube sampling. Technometrics **29**, 143–151 (1987)
54. Zuev, K.M., Beck, J.L., Au, S.-K., Katafygiotis, L.S.: Bayesian post-processor and other enhancements of Subset Simulation for estimating failure probabilities in high dimensions. Comput. Struct. **92–93**, 283–296 (2012)