An optimized Line Sampling method for the estimation of the failure probability of nuclear passive systems

Enrico Zio, Nicola Pedroni

▶ **To cite this version:**

Enrico Zio, Nicola Pedroni. An optimized Line Sampling method for the estimation of the failure probability of nuclear passive systems. Reliability Engineering and System Safety, 2010, 95 (12), pp.1300-1313. 10.1016/j.ress.2010.06.007 . hal-00609180

HAL Id: hal-00609180

https://centralesupelec.hal.science/hal-00609180

Submitted on 26 Jul 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An optimized Line Sampling method for the estimation of the failure probability of nuclear passive systems

E. Zio and N. Pedroni
Energy Department, Polytechnic of Milan, Via Ponzio 34/3, 20133 Milan, Italy
Phone: +39-2-2399-6340; fax: +39-2-2399-6309
E-mail address: enrico.zio@polimi.it

Abstract

The quantitative reliability assessment of a thermal-hydraulic (T-H) passive safety system of a nuclear power plant can be obtained by i) Monte Carlo (MC) sampling the uncertainties of the system model and parameters, ii) computing, for each sample, the system response by a mechanistic T-H code and iii) comparing the system response with pre-established safety thresholds, which define the success or failure of the safety function. The computational effort involved can be prohibitive because of the large number of (typically long) T-H code simulations that must be performed (one for each sample) for the statistical estimation of the probability of success or failure.

In this work, Line Sampling (LS) is adopted for efficient MC sampling. In the LS method, an "important direction" pointing towards the failure domain of interest is determined and a number of conditional one-dimensional problems are solved along such direction; this allows for a significant reduction of the variance of the failure probability estimator, with respect, for example, to standard random sampling.

Two issues are still open with respect to LS: first, the method relies on the determination of the "important direction", which requires additional runs of the T-H code; second, although the method has been shown to improve the computational efficiency by reducing the variance of the failure probability estimator, no evidence has been given yet that accurate and precise failure probability estimates can be obtained with a number of samples reduced to below a few hundreds, which may be required in case of long-running models.

The work presented in this paper addresses the first issue by i) quantitatively comparing the efficiency of the methods proposed in the literature to determine the LS important direction; ii) employing Artificial Neural Network (ANN) regression models as fast-running surrogates of the original, long-running T-H code to reduce the computational cost associated to the determination of the LS "important direction" and iii) proposing a new technique for identifying the LS

"important direction", based on the Genetic Algorithm (GA) minimization of the variance of the LS failure probability estimator.
In addition, this work addresses the second issue by assessing the performance of the LS method in estimating small failure probabilities with a reduced (e.g., lower than one hundred) number of samples.
The issues are investigated within two case studies: the first one deals with the estimation of the failure probability of a nonlinear structural system subject to creep and fatigue damages [1], [2]; the second one regards a passive decay heat removal system in a Gas-cooled Fast Reactor (GFR) of literature [3].

Keywords: Functional failure probability, passive system, Line Sampling, important direction, variance minimization, Artificial Neural Network, Genetic Algorithm, long-running code, computational cost.

1 Introduction

Modern nuclear reactor concepts make use of passive safety features [4], which do not need external input (especially energy) to operate [5] and, thus, are expected to improve the safety of nuclear power plants because of simplicity and reduction of both human interactions and hardware failures [6]-[8].
However, the uncertainties involved in the modelling and functioning of passive systems are usually larger than for active systems. This is due to: i) the random nature of several of the physical phenomena involved in the functioning of the system (aleatory uncertainty); ii) the incomplete knowledge on the physics of some of these phenomena (epistemic uncertainty) [9].
Due to these uncertainties, the physical phenomena involved in the passive system functioning (e.g., natural circulation) might develop in such a way to lead the system to fail its function: actually, deviations in the natural forces and in the conditions of the underlying physical principles from the expected ones can impair the function of the system itself [10].
In this view, a passive system fails to perform its function when deviations from its expected behavior lead the load imposed on the system to exceed its capacity [11]. In the reliability analysis of such functional failure behavior, the passive system is modeled by a detailed, mechanistic T-H system code and the probability of failing to perform the required function is estimated based on a Monte Carlo (MC) sample of code runs which propagate the *epistemic* (state-of-knowledge) uncertainties in the model and numerical values of its parameters/variables [3], [4], [12]-[19].

In practice, the probability of functional failure of a passive system is very small (e.g., of the order of 10^{-4} or less), so that a large number of samples is necessary for acceptable estimation accuracy [20]. Given that the time required for each run of the detailed, mechanistic T-H system model code can be of the order of several hours [4], the MC simulation-based procedure typically requires considerable computational efforts.

To reduce the computational burden of MC simulation-based approaches to reliability and risk analysis, efficient sampling techniques like Importance Sampling (IS) [21], Stratified Sampling [22] and Latin Hypercube Sampling (LHS) [23] have been widely used [24].

In this paper, we consider an advanced simulation method called Line Sampling (LS), which has been recently introduced in structural reliability analysis [25]. *Lines*, instead of random *points*, are used to probe the failure domain [26]; an "important direction" pointing towards the failure domain of interest is first determined and a number of conditional, one-dimensional problems are then solved along such direction [26]. The approach has been shown capable of substantially improving computational efficiency in a wide range of reliability applications [2], [19], [25]-[29]. If the boundary profile of the failure domain of interest is not too irregular and the "important direction" is almost *perpendicular* to it, the variance of the failure probability estimator could ideally be reduced to zero [25].

Two main issues of the LS method are still under study for its practical application in reliability and risk analysis:

1. LS relies on the determination of the important direction, which requires additional runs of the T-H model, with an increase of the computational cost.
2. LS has been shown to significantly reduce the variance of the failure probability estimator, but this must be achieved with a small number of samples (and, thus, of T-H model evaluations; say, few tens or hundreds depending on the application), for practical cases in which the computer codes require several hours to run a single simulation [4].

The present paper addresses the first issue above by:

- comparing the efficiency of a number of methods proposed in the literature to identify the important direction;
- employing Artificial Neural Network (ANN) regression models [30] as fast-running surrogates of the long-running T-H code, to reduce the computational cost associated to the identification of the LS important direction;

- proposing a new technique to determine the LS important direction, based on the *minimization* of the variance of the LS failure probability estimator.

With respect to the second issue above, this paper aims at:

- assessing the performance of the LS method in the estimation of small failure probabilities (e.g., of the order of 10^{-4}) with a *reduced* number of samples (e.g., below 100).

The *novelties* with respect to previous work performed by the authors on these issues [31] are the following:

- Genetic Algorithms (GAs) are employed as optimization algorithms, whereas in the previous work an algorithm based on Sequential Quadratic Programming-SQP was used;
- the ANN regression models are here trained according to a *sequential*, *two-step* algorithm (based on error back-propagation) in order to increase the accuracy of the ANN model estimates in proximity of the failure domain of interest; on the contrary, in the previous work a simple *one-step* algorithm was used;
- the performance of the LS method in the estimation of small failure probabilities (e.g., of the order of 10^{-4}) is assessed with a *very small* number of samples drawn (of the order of 5–50); instead, in the previous work the performance of the LS method was assessed with a number of samples of the order of one-two hundreds;
- the following probabilistic simulation methods are compared in the estimation of small failure probabilities on the basis of a *very small* number of samples drawn: i) the optimized LS method proposed in this paper, ii) a combination of the optimized LS method and Latin Hypercube Sampling (LHS), also developed in this paper, iii) Importance Sampling (IS) [21] and iv) a combination of IS and LHS [32]; in the previous work, no comparison with other simulation methods was performed.

The investigations are carried out with regards to two case studies. The first one (not considered in the previous paper [31]) deals with the estimation of the failure probability of a nonlinear structural system subject to creep and fatigue damages [1], [2]: thanks to its simplicity, it is here used as a toy-problem to *extensively test* the proposed methods, with respect to both issues 1. and 2. above. The second one (considered also in the previous paper [31]) deals with the reliability analysis of a passive, natural convection-based decay heat removal system of a Gas-cooled Fast Reactor (GFR) [3]: on the basis of the investigations performed in the first case study, only issue 2. above is tackled in the second case study.

The remainder of the paper is organized as follows. In Section 2, the reliability analysis of T-H passive systems is framed in terms of the concepts of functional failure analysis. In Section 3, a general presentation of the LS procedure is provided. In Section 4, a detailed description of the techniques employed in this work to estimate the important direction for LS is given. In Sections 5 and 6, the structural case study of literature and the case study concerning the passive cooling of a GFR are respectively presented, together with the corresponding results. Finally, a critical discussion of the results obtained is proposed in Section 7 and some conclusions are drawn in the last Section.

2 Functional failure analysis of T-H passive systems

The basic steps of a functional failure analysis of a T-H passive system are [33]:

1. Detailed modeling of the system response by means of a deterministic, best-estimate (typically long-running) T-H code.
2. Identification of the parameters/variables, models and correlations (i.e., the inputs to the T-H code) which contribute to the *uncertainty* in the results (i.e., the outputs) of the best-estimate T-H calculations.
3. Propagation of the uncertainties through the deterministic, long-running T-H code in order to estimate the functional failure probability $P(F)$ of the passive system. Formally, let $\boldsymbol{x} = \{x_1, x_2, \ldots, x_j, \ldots, x_n\}$ be the vector of the relevant system uncertain parameters, $Y(\boldsymbol{x})$ be a scalar function indicating the performance of the T-H passive system (e.g., the fuel peak cladding temperature during an accidental transient) and α_Y a threshold value (imposed e.g. by the regulatory authorities) defining the criterion of loss of system functionality. For illustrating purposes, let us assume that the passive system fails if $Y(\boldsymbol{x}) > \alpha_Y$; equivalently, introducing a variable called Performance Function (PF) as $g_x(\boldsymbol{x}) = Y(\boldsymbol{x}) - \alpha_Y$, failure occurs if $g_x(\boldsymbol{x}) > 0$. The probability $P(F)$ of system functional failure can then be expressed by the multidimensional integral:

$$P(F) = \iint \ldots \int I_F(\boldsymbol{x}) q(\boldsymbol{x}) d\boldsymbol{x} \tag{1}$$

where $q(\cdot)$ is the joint Probability Density Function (PDF) representing the uncertainty in the parameters $\boldsymbol{x}$, F is the failure region (where $g_x(\cdot) > 0$) and $I_F(\cdot)$ is an indicator function such that $I_F(\boldsymbol{x}) = 1$, if $\boldsymbol{x} \in F$ and $I_F(\boldsymbol{x}) = 0$, otherwise.

The evaluation of integral (1) above entails *multiple* (e.g., many thousands) evaluations of the T-H code for different sampled combinations of system inputs; if the running time for each T-H code simulation takes several hours (which is often the case for T-H nuclear passive systems), the

associated computing cost is prohibitive. In this paper, the computational issue is addressed by resorting to the Line Sampling (LS) technique [25], whose main concepts are given in the following Section.

3 A synthetic illustration of the Line Sampling technique

Line Sampling (LS) is a simulation method for efficiently computing small failure probabilities. The underlying idea is to employ *lines* instead of random *points* in order to probe the failure domain of the system analyzed [25], [26].

In extreme synthesis, the computational steps of the algorithm are [26], [28]:

1. From the original multidimensional joint probability density function $q(\cdot):\Re^n \to [0,\infty)$, sample N_T vectors $\{\boldsymbol{x}^k : k=1,2,...,N_T\}$, with $\boldsymbol{x}^k = \{x_1^k, x_2^k, ..., x_j^k, ..., x_n^k\}$.
2. Transform the N_T sample vectors $\{\boldsymbol{x}^k : k=1,2,...,N_T\}$ defined in the original (i.e., physical) space into N_T samples $\{\boldsymbol{\theta}^k : k=1,2,...,N_T\}$ defined in the standard normal space; also the PFs $g_x(\cdot)$ defined in the physical space have to be transformed into $g_\theta(\cdot)$ in the standard normal space [34].
3. In the standard normal space, determine the *unit* important direction $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, ..., \alpha_j, ..., \alpha_n\}^T$ (hereafter also called "important unit vector" or "important direction") pointing towards the failure domain *F* of interest (see Section 4 below).
4. Reduce the problem of computing the high-dimensional failure probability integral (1) to a number of conditional one-dimensional problems, solved along the "important direction" $\boldsymbol{\alpha}$ in the standard normal space: in particular, estimate N_T conditional "one-dimensional" failure probabilities $\{\hat{P}(F)^{1\boldsymbol{D},k} : k=1,2,...,N_T\}$, corresponding to each one of the standard normal samples $\{\boldsymbol{\theta}^k : k=1,2,...,N_T\}$ obtained in step 2. Notice that $2 \cdot N_T$ or $3 \cdot N_T$ system performance analyses (i.e., runs of the T-H model code) have to be carried out to calculate *each* of the N_T conditional one-dimensional failure probability estimates $\{\hat{P}(F)^{1\boldsymbol{D},k} : k=1,2,...,N_T\}$ (see [26] and [28] for details).
5. Compute the unbiased estimator $\hat{P}(F)^{N_T}$ for the failure probability $P(F)$ and its variance $\sigma^2\left[\hat{P}(F)^{N_T}\right]$ as:

$$\hat{P}(F)^{N_T} = 1/N_T \cdot \sum_{k=1}^{N_T} \hat{P}(F)^{1\boldsymbol{D},k}, \tag{2}$$

$$\sigma^2\left[\hat{P}(F)^{N_T}\right] = 1/N_T(N_T-1) \cdot \sum_{k=1}^{N_T} \left(\hat{P}(F)^{1D,k} - \hat{P}(F)^{N_T}\right)^2. \tag{3}$$

The LS method here outlined can significantly reduce the variance (3) of the estimator (2) of the failure probability integral (1) [25]; however, its efficiency depends on the determination of the important direction $\boldsymbol{\alpha}$ (step 3. above): the following Section delves further into this issue.

4 Methods for the determination of the important direction $\boldsymbol{\alpha}$

In what follows, the methods used in this work to determine the LS important direction $\boldsymbol{\alpha}$ are presented in detail: in Section 4.1, the techniques proposed in the literature are critically reviewed; in Section 4.2, a new method based on the minimization of the variance of the LS failure probability estimator is proposed.

4.1 Literature methods

4.1.1 Normalized "center of mass" of the failure domain *F*

The important unit vector $\boldsymbol{\alpha}$ can be computed as the normalized "center of mass" of the failure domain F of interest [25]. A point $\boldsymbol{\theta}^0$ is taken in the failure domain F: this can be done by engineering judgment when possible. Subsequently, $\boldsymbol{\theta}^0$ is used as the initial point of a Markov chain which lies entirely in the failure domain F. For that purpose, a Metropolis-Hastings algorithm is employed to generate a sequence of N_s points $\left\{\boldsymbol{\theta}^u : u = 1, 2, ..., N_s\right\}$ lying in the failure domain F [35]. The unit vectors $\boldsymbol{\theta}^u \big/ \left\|\boldsymbol{\theta}^u\right\|_2$, u = 1, 2, …, N_s, are then averaged in order to obtain the LS important unit vector as $\boldsymbol{\alpha} = \frac{1}{N_s} \cdot \sum_{u=1}^{N_s} \boldsymbol{\theta}^u \big/ \left\|\boldsymbol{\theta}^u\right\|_2$ (Figure 1, top, left). This direction provides a good "map" of the important regions of the failure domain (at least as the sample size N_s is large); on the other hand, the procedure implies N_s additional system analyses by the T-H model, which may substantially increase the computational cost associated to the simulation method.

4.1.2 Direction of the design point in the standard normal space

A plausible selection of $\boldsymbol{\alpha}$ could be the direction of the "design point" in the standard normal space [27], [36]. According to a geometrical interpretation, the "design point" is defined as the point $\boldsymbol{\theta}^*$ on the limit state surface $g_\theta(\boldsymbol{\theta}) = 0$ in the standard normal space, which is closest to the origin (Figure 1, top, right). It can be computed by solving the following constrained nonlinear minimization problem:

$$\text{Find}\, \boldsymbol{\theta}^* : \left\|\boldsymbol{\theta}^*\right\|_2 = \min_{g_\theta(\boldsymbol{\theta})=0} \left\{\left\|\boldsymbol{\theta}\right\|_2\right\} \tag{4}$$

where $\|\cdot\|_2$ denotes the usual Euclidean measure of a vector.

Then, the unit important vector $\boldsymbol{\alpha}$ can be easily obtained by normalizing $\boldsymbol{\theta}^*$, i.e., $\boldsymbol{\alpha} = \boldsymbol{\theta}^* / \left\| \boldsymbol{\theta}^* \right\|_2$.

In this work, Genetic Algorithms (GAs) [37], [38] are used to solve the constrained nonlinear minimization problem (4). In extreme synthesis, the main properties of GAs are that the optimization search is conducted i) using a (possibly) large population of multiple solution points or candidates, ii) using operations inspired by the evolution of species, such as breeding and genetic mutation, iii) using probabilistic operations and iv) using information on the objective or search function and not on its derivatives. With regards to their performance, it is acknowledged that GAs take a more *global* view of the search space than many other optimization methods. The main advantages are i) fast convergence to near global optimum, ii) superior global searching capability in complicated search spaces and iii) applicability even when gradient information is not readily achievable [38]. A thorough descriptions of the GA computational flow is not reported here for brevity sake: for further details, the interested reader may refer to the cited references and the copious literature in the field.

Notice that checking the feasibility of a candidate solution $\boldsymbol{\theta}$ to (4) requires the evaluation of the PF $g_\theta(\cdot)$ at $\boldsymbol{\theta}$, which entails running the numerical T-H model code simulating the system. As a consequence, the computational cost associated with the calculation of the design point can be quite high, in particular if long-running numerical codes are used to simulate the response of the system to its uncertain input parameters [27], as it is the case in the functional failure analysis of T-H passive systems.

4.1.3 Gradient of the performance function in the standard normal space

In [26], the direction of $\boldsymbol{\alpha}$ is taken as the normalized gradient of the PF $g_\theta(\cdot)$ in the standard normal space. Since the unit vector $\boldsymbol{\alpha}$ points towards the failure domain *F*, it can be used to draw information about the relative importance of the uncertain parameters $\{\theta_j : j = 1, 2, ..., n\}$ with respect to the failure probability *P*(*F*): the more relevant an uncertain variable is in determining the failure of the system, the larger the corresponding component of the unit vector $\boldsymbol{\alpha}$ will be [26]. Such quantitative information is obtained from the gradient of the performance function $g_\theta(\boldsymbol{\theta})$ in the standard normal space, $\nabla g_\theta(\boldsymbol{\theta})$:

$$\nabla g_\theta(\boldsymbol{\theta}) = \left[\frac{\partial g_\theta(\boldsymbol{\theta})}{\partial \theta_1} \frac{\partial g_\theta(\boldsymbol{\theta})}{\partial \theta_2} \cdots \frac{\partial g_\theta(\boldsymbol{\theta})}{\partial \theta_j} \cdots \frac{\partial g_\theta(\boldsymbol{\theta})}{\partial \theta_n} \right]^T \tag{5}$$

The gradient (5) measures the relative importance of a particular uncertain variable with respect to the failure probability *P*(*F*): the larger the (absolute) value of a component of (5), the greater the

"impact" of the corresponding uncertain variable on the performance function $g_\theta(\boldsymbol{\theta})$ in the standard normal space. Thus, it is reasonable to identify the LS important direction with the direction of the gradient (5) and compute the corresponding unit vector $\boldsymbol{\alpha}$ as the *normalized* gradient of the performance function $g_\theta(\cdot)$ in the standard normal space, i.e. $\boldsymbol{\alpha} = \nabla g_\theta(\boldsymbol{\theta}) / \left\| \nabla g_\theta(\boldsymbol{\theta}) \right\|_2$ [26].

For clarity sake, Figure 1 bottom shows this procedure with reference to a two-dimensional problem: the important unit vector $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2\}$ associated to the two-dimensional performance function $g_\theta(\theta_1, \theta_2)$ is computed at a proper (selected) point $\boldsymbol{\theta}^0 = \left\{\theta_1^0, \theta_2^0\right\}^T$ (e.g., the nominal point of the system under analysis). Notice that since component $\alpha_1 = \left.\frac{\partial g_\theta(\boldsymbol{\theta})}{\partial \theta_1}\right|_{\boldsymbol{\theta}^0} \Big/ \left\| \nabla g_\theta(\boldsymbol{\theta})\big|_{\boldsymbol{\theta}^0} \right\|_2$ (Figure 1 bottom, left) is significantly larger than component $\alpha_2 = \left.\frac{\partial g_\theta(\boldsymbol{\theta})}{\partial \theta_2}\right|_{\boldsymbol{\theta}^0} \Big/ \left\| \nabla g_\theta(\boldsymbol{\theta})\big|_{\boldsymbol{\theta}^0} \right\|_2$ (Figure 1 bottom, right), uncertain variable θ_1 will be far more important than θ_2 in leading the system to failure.

Finally, notice that as the PF $g_\theta(\boldsymbol{\theta})$ is known only implicitly through the response of a numerical code, for a given vector $\boldsymbol{\theta} = \left\{\theta_1, \theta_2, ..., \theta_j, ..., \theta_n\right\}^T$ at least *n* system performance analyses are required to determine accurately the gradient (5) at a given point of the PF $g_\theta(\cdot)$, e.g., by numerical differentiation [39], [40].

Figure 1 here

All the techniques presented require additional runs of the T-H model code, with increase of the overall computational cost associated to the LS method. To improve on this issue, the substitution of the long-running T-H model code by a fast-running surrogate regression model is here investigated. The regression model is constructed on the basis of a *limited-size* set of data representing examples of the input/output nonlinear relationships underlying the original T-H code. Once built, the model can be used for performing, in an acceptable computational time, the evaluations of the system PF $g_\theta(\cdot)$ needed for an accurate estimation of the LS important direction $\boldsymbol{\alpha}$. In this work, a three-layered feed-forward Artificial Neural Network (ANN) regression model is considered. In extreme synthesis, ANNs are computing devices inspired by the function of the nerve cells in the brain [30]. They are composed of many parallel computing units (called neurons or nodes) interconnected by weighed connections (called synapses). Each of these computing units performs a few simple operations and communicates the results to its neighbouring units. From a mathematical viewpoint, ANNs consist of a set of nonlinear (e.g., sigmoidal) basis functions with

adaptable parameters that are adjusted by a process of *training* (on different input/output data examples), i.e., an iterative process of regression error minimization [41]. The details about ANN regression models are not reported here for brevity: for further details, the interested reader may refer to the cited references and the copious literature in the field.

The particular type of ANN employed in this paper is the classical three-layered feed-forward ANN; in order to improve the *accuracy* in the approximation of the system PF $g_\theta(\cdot)$ (needed for an accurate estimation of the LS important direction $\boldsymbol{\alpha}$), the employed ANN models are trained by a properly devised *sequential, two-step* algorithm based on error back-propagation. In extreme synthesis, a *first-step* ANN regression model is built using a set D'_{train} of input/output data examples of size N'_{train}; further, a *validation* data set D'_{val} (different from the training set) of size N_{val}' is used to monitor the accuracy of the first-step ANN model during the training procedure in order to avoid overfitting of the training data [41]. The resulting ANN model is used (instead of the original, long-running system model code) to provide an *approximation* to the *design point* of the problem (Section 4.1.2): this is meant to provide an approximate, rough indication of the real location of the failure domain F of interest. Subsequently, new training and validation data sets D''_{train} and D''_{val} of sizes N''_{train} and N''_{val}, respectively, are randomly generated *centred* on the approximate design point previously identified: a *second-step* (i.e., definitive) ANN model is then constructed on these newly generated training and validation data sets. This should result in an ANN regression model which is more accurate in proximity of the failure domain F of interest, thus providing reliable estimates of the system PF $g_\theta(\cdot)$ for the identification of the LS important direction $\boldsymbol{\alpha}$.

4.2 Minimization of the variance of the LS failure probability estimator

The *optimal* important direction $\boldsymbol{\alpha}^{opt}$ for Line Sampling can be defined as the one minimizing the variance $\sigma^2\left[\hat{P}(F)^{N_T}\right]$ (3) of the LS failure probability estimator $\hat{P}(F)^{N_T}$ (2). Notice that $\boldsymbol{\alpha}^{opt}$ can be expressed as the normalized version of a proper vector $\boldsymbol{\theta}^{opt}$ in the standard normal space, i.e., $\boldsymbol{\alpha}^{opt} = \boldsymbol{\theta}^{opt} \Big/ \left\|\boldsymbol{\theta}^{opt}\right\|_2$. Thus, in order to search for a physically meaningful important unit vector $\boldsymbol{\alpha}^{opt}$ (i.e., a vector that optimally points towards the failure domain F of interest), $\boldsymbol{\theta}^{opt}$ should belong to the failure domain F of interest, i.e. $\boldsymbol{\theta}^{opt} \in F$ or, equivalently, $g_\theta\left(\boldsymbol{\theta}^{opt}\right) > 0$.

In mathematical terms, the optimal LS important direction $\boldsymbol{\alpha}^{opt}$ is obtained by solving the following nonlinear constrained minimization problem:

$$\begin{aligned} &\text{Find } \boldsymbol{\alpha}^{opt} = \boldsymbol{\theta}^{opt} \Big/ \left\|\boldsymbol{\theta}^{opt}\right\|_2 : \ \sigma^2\left[\hat{P}(F)^{N_T}\right] = \min_{\boldsymbol{\alpha}=\boldsymbol{\theta}/\|\boldsymbol{\theta}\|_2} \left\{\sigma^2\left[\hat{P}(F)^{N_T}\right]\right\} \\ &\text{subject to } \boldsymbol{\theta} \in F \ (\text{i.e., } g_\theta(\boldsymbol{\theta}) > 0). \end{aligned} \tag{6}$$

The conceptual steps of the procedure for solving (6) are (Figure 2):

1. An optimization algorithm proposes a candidate solution $\boldsymbol{\alpha} = \boldsymbol{\theta} / \|\boldsymbol{\theta}\|_2$ to (6): as previously mentioned, in this work Genetic Algorithms (GAs) are employed.
2. The LS failure probability estimator $\hat{P}(F)^{N_T}$ (2) and the associated variance $\sigma^2\left[\hat{P}(F)^{N_T}\right]$ (3) are calculated using the unit vector $\boldsymbol{\alpha} = \boldsymbol{\theta} / \|\boldsymbol{\theta}\|_2$ proposed as important direction in step 1. above; notice that $2 \cdot N_T$ or $3 \cdot N_T$ system performance analyses (i.e., runs of the system model code) have to be carried out in this phase (see steps 4. and 5. in Section 3).
3. The variance $\sigma^2\left[\hat{P}(F)^{N_T}\right]$ obtained in step 2. above is the objective function to be minimized; it measures the quality of the candidate solution $\boldsymbol{\alpha} = \boldsymbol{\theta} / \|\boldsymbol{\theta}\|_2$ proposed by the optimization algorithm in step 1. above.
4. The feasibility of the proposed solution $\boldsymbol{\alpha} = \boldsymbol{\theta} / \|\boldsymbol{\theta}\|_2$ is checked by evaluating the system PF $g_{\theta}(\cdot)$ (i.e., by running the system model code) in correspondence of $\boldsymbol{\theta}$: if the proposed solution $\boldsymbol{\alpha} = \boldsymbol{\theta} / \|\boldsymbol{\theta}\|_2$ is not feasible (i.e., if $\boldsymbol{\theta} \notin F$ or, equivalently, $g_{\theta}(\boldsymbol{\theta}) \leq 0$), it is *penalized* by *increasing* the value of the corresponding objective function $\sigma^2\left[\hat{P}(F)^{N_T}\right]$ through an *additive* factor [37].
5. Steps 1. − 4. are repeated until a predefined stopping criterion is met and the optimization algorithm identifies the *optimal* unit vector $\boldsymbol{\alpha}^{opt} = \boldsymbol{\theta}^{opt} / \|\boldsymbol{\theta}^{opt}\|_2$.

Notice that i) the optimization search requires the iterative evaluation of hundreds or thousands of possible solutions $\boldsymbol{\alpha} = \boldsymbol{\theta} / \|\boldsymbol{\theta}\|_2$ to (6) and ii) $2 \cdot N_T$ or $3 \cdot N_T$ system performance analyses (i.e., runs of the system model code) have to be carried out to calculate the objective function $\sigma^2\left[\hat{P}(F)^{N_T}\right]$ for *each* proposed solution; as a consequence, the computational effort associated to this technique would be absolutely prohibitive with a system model code requiring hours or even minutes to run a single simulation. Hence, it is unavoidable, for practical applicability, to resort to a regression model (ANN-based, in this work) as a fast-running approximator of the original system model for performing the calculations in steps 2. and 4. above, to make the computational cost acceptable.

Figure 2 here

The characteristics of the methods described in Sections 4.1 and 4.2 are summarized in Table 1, with the specification of the computational tools employed for their implementation.

Table 1 here

5 Case study 1: structural system of literature

The first case study deals with a probabilistic model for the reliability analysis of creep and fatigue failure phenomena in structural materials: the model was first proposed in [1] and then employed also in [2].

According to the above mentioned references, the nonlinear Performance Function (PF) $g_x(\cdot)$ of a structural material subject to creep and fatigue damages can be expressed as

$$\begin{aligned} g_x(\boldsymbol{x}) &= g_x(x_1, x_2, x_3, x_4, x_5, x_6) \\ &= g_x(N_c, N_f, n_c, n_f, \theta_1, \theta_2) = 2 - e^{\theta_1 D_c} + \frac{e^{\theta_1} - 2}{e^{-\theta_2} - 1}\left(e^{-\theta_2 D_c} - 1\right) - D_f \end{aligned} \tag{7}$$

where $D_c = n_c/N_c$ and $D_f = n_f/N_f$ are the creep and fatigue damages, respectively, N_c and N_f are the creep and fatigue lives, respectively, n_c and n_f are the numbers of the creep and fatigue load cycles, respectively, and θ_1 and θ_2 are characteristic parameters of the structural material obtained from experimental data. The structural material is supposed to fail when its PF (7) becomes lower than or equal to 0, i.e., $g_x(\boldsymbol{x}) \leq 0$.

The shapes and parameters (i.e., mean μ and standard deviation σ) of the probability distribution functions associated to the uncertain variables $\{x_j: j = 1, 2, ..., 6\}$ of the probabilistic model (7) for creep and fatigue in structural materials are summarized in Table 2 [2].

The true (i.e., reference) probability $P(F)$ of the failure event $F = \{g_x(\boldsymbol{x}) \leq 0\}$ is $1.425 \cdot 10^{-4}$, obtained as an average of $S = 25000$ failure probability estimates $\hat{P}(F)_s^{N_T}$, $s = 1, 2, \ldots, S$, each one computed by standard MCS with $N_T = 500000$ samples.

Table 2 here

5.1 Application 1: comparison of the methods proposed in Section 4 for determining the important direction $\boldsymbol{\alpha}$ for Line Sampling

LS is here applied to the probabilistic model (7) described above for creep and fatigue in structural materials. In particular, in this Section a thorough comparison of the different methods proposed in Section 4 for determining the important direction $\boldsymbol{\alpha}$ for Line Sampling is carried out: in Section 5.1.1, the different experimental settings considered are described in details, together with the methods and models used, and the objectives; in Section 5.1.2, the quantitative indicators

introduced to compare the methods adopted are presented; finally, the results obtained in the different experimental settings of Section 5.1.1 are illustrated in Section 5.1.3.

5.1.1 Experimental settings

The simulations performed are intended to *compare* the efficiency of the different methods considered for the determination of the LS important direction $\boldsymbol{\alpha}$ (Section 4). In each LS simulation, the system performance function $g_\theta(\cdot)$ is evaluated by running the *original* system model code and the LS point estimates $\hat{P}(F)^{N_T}$ of the failure probability $P(F)$ are computed with a *large* number N_T (i.e., N_T = 10000) of samples (steps 4. and 5. of Section 3): this allows a reliable assessment of the effect of different important directions $\boldsymbol{\alpha}$ on the *accuracy* and *precision* of the obtained estimates $\hat{P}(F)^{N_T}$. In this case the use of a large number of samples N_T (i.e., N_T = 10000) is possible because the system performance function (7) is a *simple analytical* function which can be evaluated in a *negligible* computational time.

Three different experimental settings, namely settings 1, 2 and 3, are considered in this application. These settings differ by:

i) the method used for determining the important direction $\boldsymbol{\alpha}$ (Section 4);
ii) the model employed to evaluate the system performance function $g_\theta(\cdot)$ for the estimation of the important direction $\boldsymbol{\alpha}$;
iii) the number N_α of system performance evaluations used to determine $\boldsymbol{\alpha}$;
iv) the total number $N_{code,\alpha}$ of *actual* runs of the *original* system model code required by the *whole* process of determination of the important direction $\boldsymbol{\alpha}$.

The characteristics of the three settings are summarized in Table 3.

Table 3 here

In setting 1, the MCMC (labeled A, Section 4.1.1), design point (labeled B, Section 4.1.2) and gradient (labeled C, Section 4.1.3) methods are considered. A *large* number N_α of evaluations of the system performance function $g_\theta(\cdot)$ are carried out to determine $\boldsymbol{\alpha}$: in particular, N_α = 10000 is chosen to provide an accurate and reliable estimate for the important direction $\boldsymbol{\alpha}$. In this setting, the system performance function is evaluated by running the *original* system model "code" (i.e., the original system performance function $g_\theta(\cdot)$), so that $N_{code,\alpha} = N_\alpha$ = 10000.

In setting 2, N_α = 10000 evaluations of the system performance function $g_\theta(\cdot)$ are carried out to determine $\boldsymbol{\alpha}$, like in the previous setting 1. However, in this setting the system performance function

$g_\theta(\cdot)$ is evaluated by resorting to a *fast-running* ANN regression model approximating the original system performance function. The objective is to verify the possibility of reducing the computational cost associated to the LS method by using ANN regression models in place of the original system model. In particular, the ANN regression model is constructed on the basis of a *small* set of data representing examples of the input/output nonlinear relationships underlying the original system model; once built, the regression model is used to evaluate (in a negligible computational time) the system performance function $g_\theta(\cdot)$ for the determination of the important direction $\boldsymbol{\alpha}$ (steps 4. and 5. of Section 3).

A classical three-layered feed-forward ANN (trained by the *sequential, two-step* error back-propagation algorithm described at the end of Section 4.1) is here adopted: the number of inputs to the ANN regression model is 6 (i.e., the number of uncertain variables in Table 2 of Section 5), whereas the number of outputs is 1 (i.e., the value of the system performance function). The number of nodes in the hidden layer has been set equal to 5 by trial and error. The *first-step* ANN model is built using a set of input/output data examples of size $N_{train}' = 50$; further, a *validation* data set (different from the training set) of size $N_{val}' = 10$ is used to monitor the accuracy of the first-step ANN model during the training procedure, in order to avoid overfitting of the training data [41]. Subsequently, the *second-step* (i.e., definitive) ANN model is built using training and validation sets of sizes $N_{train}'' = 100$ and $N_{val}'' = 20$, respectively; finally, a *test* set of size $N_{test} = 10$, not used during the training and validation phases, is employed to provide a realistic measure of the second-step ANN model accuracy. Thus, the *total* number of system model *runs* performed to generate the two training sets, two validation sets and final test set is $N_{code,\alpha} = (N_{train}' + N_{val}' + N_{train}'' + N_{val}'' + N_{test}) = 50 + 10 + 100 + 20 + 10 = 190$.

Correspondingly, the total computational cost associated to the estimation of $\boldsymbol{\alpha}$ in setting 2 is much *lower* than that of setting 1, in spite of the same number N_α of system performance evaluations. Actually, when a single run of the system model code lasts several hours (which is often the case for passive safety systems) the total number $N_{code,\alpha}$ of simulations is the *critical* parameter which determines the overall computational cost associated to the method.

Further, in setting 2, the methods A, B and C are compared to the new one proposed in this paper, i.e., the one based on the minimization of the variance of the LS failure probability estimator (labeled D, Section 4.2).

The final setting 3 is similar to setting 1: methods A, B and C are used to determine $\boldsymbol{\alpha}$ and the original system model is run to evaluate the system performance function $g_\theta(\cdot)$; however, like in the previous setting 2, the number N_α of system performance evaluations (and, thus, the actual number

$N_{code,\alpha}$ of runs of the original system model) is $N_{code,\alpha} = (N_{train}' + N_{val}' + N_{train}'' + N_{val}'' + N_{test}) = 50 + 10 + 100 + 20 + 10 = 190$. Notice that in this setting, method D, based on the minimization of the variance of the LS failure probability estimator, is not employed for determining $\boldsymbol{\alpha}$ because the actual number of "allowed" code runs (i.e., $N_{code,\alpha} = 190$) is too small to provide *meaningful* results for this method.

5.1.2 Performance indicators

The experimental settings described in the previous Section 5.1.1 are compared in terms of two quantities: the percentage relative *error* ε between the LS failure probability estimate $\hat{P}(F)^{N_T}$ and the true (i.e., reference) value $P(F)$ of the failure probability of the system, and the percentage relative *width* w_{CI} of the 95% Confidence Interval (CI) of the LS failure probability estimator $\hat{P}(F)^{N_T}$. These indicators are defined in (8) and (9), respectively:

$$\varepsilon = \frac{\left|\hat{P}(F)^{N_T} - P(F)\right|}{P(F)} \cdot 100, \tag{8}$$

$$w_{CI} = \frac{U_{CI,\hat{P}(F)^{N_T}} - L_{CI,\hat{P}(F)^{N_T}}}{P(F)} \cdot 100, \tag{9}$$

where $U_{CI,\hat{P}(F)^{N_T}}$ and $L_{CI,\hat{P}(F)^{N_T}}$ are the upper and lower bounds of the 95% CI of the failure probability estimator $\hat{P}(F)^{N_T}$, respectively.

Obviously, the lower is the value of ε, the higher is the *accuracy* of the failure probability estimate $\hat{P}(F)^{N_T}$; instead, the lower is the value of w_{CI}, the higher the *precision* of the estimate.

5.1.3 Results

As previously mentioned, the example application has been set with the purpose of comparing different methods for determining the LS important direction $\boldsymbol{\alpha}$ (Section 4).

Figure 3 shows the values of the LS point estimates $\hat{P}(F)^{N_T}$ (dots) of the failure probability $P(F)$ obtained with $N_T = 10000$ samples in settings 1, 2 and 3 (Table 3); the corresponding 95% Confidence Intervals (CIs) are also reported (bars). Finally, the true (i.e., reference) value of the system failure probability $P(F)$ (i.e., $P(F) = 1.425 \cdot 10^{-4}$) is shown as a dashed line. Table 4 reports instead the values of the associated performance indicators ε and w_{CI} (Section 5.1.2).

Figure 3 here

Table 4 here

The results obtained in setting 1 show that method A (i.e., MCMC simulation) provides more *accurate* (i.e., the estimates are closer to the true values) and *precise* (i.e., the confidence intervals are narrower) estimates than methods B (i.e., design point) and C (i.e., gradient): the percentage errors ε are 0.421, 0.702 and 1.965, whereas the percentage 95% CI widths w_{CI} are 2.222, 2.282, and 7.323 for methods A, B and C, respectively. This can be explained by the fact that method A relies on a "map" approximating the failure domain F under analysis (given by the failure samples generated through a Markov chain) and thus it provides in principle the most realistic and reliable estimate for the LS important direction $\boldsymbol{\alpha}$.

Moreover, it is evident that method B (i.e., design point) performs consistently better than method C (i.e., gradient). Actually, although design points do not always represent the *most important* regions of the failure domain F, especially in high-dimensional spaces [27], they still provide an acceptable indication of the *real location* of the failure region F of interest. On the contrary, calculating $\boldsymbol{\alpha}$ through the normalized gradient of the performance function $g_\theta(\cdot)$ makes the values of the components of $\boldsymbol{\alpha}$ strongly dependent on the *point* where the *first-order*, *local* approximations of the performance function $g_\theta(\cdot)$ are carried out, and thus would relate inherently local (and possibly misleading) information: this effect is particularly critical for *nonlinear* systems like that of the case at hand.

In setting 2 the evaluation of the system performance function $g_\theta(\cdot)$ for the determination of $\boldsymbol{\alpha}$ is performed by replacing the original system model with an ANN (with $N_{train}' = 50$, $N_{val}' = 10$, $N_{train}'' = 100$, $N_{val}'' = 20$, $N_{test} = 10$ input/output examples employed in the first- and second-step training, first- and second-step validation and test phases, respectively). The number N_α of system performance evaluations is the same as in setting 1 (i.e., $N_\alpha = 10000$); however, the number $N_{code,\alpha}$ of *actual* runs of the original system model code is much lower: indeed, in setting 1 $N_{code,\alpha} = 10000$, whereas in setting 2 $N_{code,\alpha} = 190$: this means that the overall *computational effort* associated to setting 2 is *much lower* than that of setting 1.

It can be seen that the results obtained with methods A, B and C in setting 2 are *comparable* to those produced by the same methods in setting 1: the percentage errors ε are 0.421, 0.702 and 1.965 for methods A, B and C, respectively, in setting 1, and 0.211, 0.351 and 0.772 for methods A, B and C, respectively, in setting 2; the percentage 95% CI widths w_{CI} are 2.222, 2.282, and 7.323 for methods A, B and C, respectively, in setting 1, and 2.723, 2.516 and 7.199 for methods A, B and C, respectively, in setting 2.

Further, the proposed method D (Section 4.2) achieves *more accurate* and *precise* estimates than those of methods A, B and C in both settings 1 and 2: indeed, the percentage error ε and 95% CI width w_{CI} are 0.070 and 2.204, respectively; these improved results are due to the fact that the proposed technique is based on the definition of the *ideal* (i.e., optimal) important direction $\boldsymbol{\alpha}$ for LS (i.e., the one minimizing the variance of the LS failure probability estimator).
Finally, an important remark is in order with respect to the comparison between settings 1 and 2; the results produced in setting 2 are at least *comparable*, if not better, than those of setting 1; yet, they are obtained at a much *lower computational effort* thanks to the fast-running ANN approximation of the system performance function $g_\theta(\cdot)$.

A comparison can also be made between settings 2 and 3: actually, the number $N_{code,\alpha}$ of runs of the original system model code (and thus the associated *overall computational effort*) is the same for both settings (i.e., $N_{code,\alpha} = 190$). However, in setting 2 the *few* system model code runs are *directly* used to estimate $\boldsymbol{\alpha}$ (i.e., $N_{code,\alpha} = N_\alpha = 190$), whereas in setting 3 they are used to build an ANN regression model, which is in turn employed to estimate $\boldsymbol{\alpha}$ (i.e., $N_{code,\alpha} = 190 \neq N_\alpha = 10000$). It is evident that the methods A, B, C and D in setting 2 *outperform* the corresponding methods in setting 3: the percentage 95% CI widths w_{CI} are 2.723, 2.516, 7.199 and 2.204 for methods A, B, C and D in setting 2, respectively, whereas they are 6.697, 5.345 and 7.502 for methods A, B, and C in setting 3, respectively.
These findings bear an important practical implication: when a *low* number $N_{code,\alpha}$ of system model evaluations is *a priori imposed* due to computational time limitations (which is the case for long-running codes), superior results are obtained if the outcomes of the evaluations are employed to build a surrogate ANN regression model for determining the important direction $\boldsymbol{\alpha}$ instead of directly using them for estimating $\boldsymbol{\alpha}$.

Finally, let us compare settings 1 and 3. In both settings, the original system model is *directly* employed for estimating $\boldsymbol{\alpha}$: however, in setting 1 a *large* number of system model evaluations (i.e., $N_{code,\alpha} = N_\alpha = 10000$) are performed, whereas in setting 2 only a *small* number is used (i.e., $N_{code,\alpha} = N_\alpha = 190$). As expected, the precisions provided by methods A, B and C in setting 1 are significantly better than those produced by the same methods in setting 3: the percentage 95% CI widths w_{CI} are 2.222, 2.282, and 7.323 for methods A, B and C in setting 1, respectively, whereas they are 6.697, 5.345 and 7.502 for methods A, B, and C in setting 3.
In addition, it seems interesting to note that the difference between the performances of methods A, B and C is lower when N_α (= $N_{code,\alpha}$) is small (e.g., equal to 190) than when it is large (e.g., equal to

10000). This is due to the fact that the efficiency of methods A (based on MCMC simulation) and B (based on design point identification through optimization algorithms) strongly relies on the possibility of deeply exploring the uncertain parameter space within the failure region F of interest: if only a small number N_α (= $N_{code,\alpha}$) of system performance evaluations is available, such a deep search cannot be carried out, thus resulting in poor estimates of the important direction $\boldsymbol{\alpha}$. In such cases, even a simple procedure like method C (i.e., gradient estimation by straightforward numerical differentiation) may provide comparable results.

The conclusions on the accuracy and precision in the estimates provided by the important direction $\boldsymbol{\alpha}$ determined by method D (i.e., the one proposed in this paper, based on the minimization of the variance of the LS failure probability estimator) justify its adoption in the subsequent applications.

5.2 Application 2: failure probability estimation using an optimized Line Sampling method with small sample sizes

The objective of this application is verifying the possibility of obtaining accurate and precise estimates $\hat{P}(F)^{N_T}$ of *small* failure probabilities $P(F)$ (e.g., of the order of 10^{-4}) even reducing the number of system model evaluations to below one hundred, which may be mandatory in practical applications of computer codes requiring several hours to run a single simulation. Thus, in the present analysis the system performance function $g_\theta(\cdot)$ is evaluated by means of the original system model; however, the number N_T of samples drawn for the estimation of the system failure probability is much lower than in Application 1: indeed, sample sizes N_T ranging from 5 to 50 are employed (more precisely, N_T = 5, 10, 20, 30, 40 and 50).

In addition, the benefits coming from the use of an optimized Line Sampling method with very small sample sizes N_T is shown by means of a comparison between the estimation *accuracies* and *precisions* of the following simulation methods:

i) optimized Line Sampling (LS) (Sections 3 and 4.2);
ii) an original combination of optimized Line Sampling (LS) and Latin Hypercube Sampling (LHS) (hereafter referred to as LS + LHS);
iii) standard Importance Sampling (IS) [21];
iv) a combination of standard Importance Sampling (IS) and Latin Hypercube Sampling (LHS) (hereafter referred to as IS + LHS) [32].

Thorough descriptions of methods ii) – iv) above (i.e., LS + LHS, IS and IS + LHS) are not reported here for brevity: the interested reader may refer to the cited references for details.

In Section 5.2.1, the quantitative indicators used to compare methods i) – iv) above are presented; then, the results produced by all the methods considered are investigated in Section 5.2.2.

5.2.1 Performance indicators

In order to properly represent the randomness of the probabilistic simulation methods adopted and provide a statistically meaningful comparison between their performances in the estimation of the system failure probability $P(F)$, $S = 2000$ independent runs of each method have been carried out for each sample size N_T: this is required by the fact that in this application the sample sizes N_T are very small, such that they would produce poor statistics over a *single* simulation run. In each simulation $s = 1, 2, \ldots, S$, the percentage relative absolute error ε_s between the true (reference) value of the system failure probability $P(F)$ and the corresponding estimate $\hat{P}(F)_s^{N_T}$ is computed as follows:

$$\varepsilon_s = \frac{\left| P(F) - \hat{P}(F)_s^{N_T} \right|}{P(F)} \cdot 100 \text{ , } s = 1, 2, \ldots, S \tag{10}$$

The accuracies of the simulation method of interest in the estimation of $P(F)$ are then compared in terms of the mean percentage relative absolute error $\bar{\varepsilon}$ over $S = 2000$ runs:

$$\bar{\varepsilon} = \frac{1}{S} \cdot \sum_{s=1}^{S} \varepsilon_s \tag{11}$$

The quantity (11) provides a measure of the percentage relative absolute error in the estimation of the failure probability $P(F)$ made *on average in a single run* by the simulation method with N_T samples.

The failure probability estimates $\hat{P}(F)_s^{N_T}$, $s = 1, 2, \ldots, S$, are then used to build a *bootstrapped* 95% Confidence Interval (CI) for the failure probability estimator $\hat{P}(F)^{N_T}$, i.e.,

$$\left] L_{CI,\hat{P}(F)^{N_T}}, U_{CI,\hat{P}(F)^{N_T}} \right[\tag{12}$$

where $U_{CI,\hat{P}(F)^{N_T}}$ and $L_{CI,\hat{P}(F)^{N_T}}$ are the 2.5^{th} and 97.5^{th} percentiles, respectively, of the *bootstrapped empirical distribution* of the failure probability estimator $\hat{P}(F)^{N_T}$. The percentage relative *width* $\bar{w}_{CI}$ of the bootstrapped 95% Confidence Interval (CI) of the LS failure probability estimator $\hat{P}(F)^{N_T}$ is then computed as

$$\bar{w}_{CI} = \frac{U_{CI,\hat{P}(F)^{N_T}} - L_{CI,\hat{P}(F)^{N_T}}}{P(F)} \cdot 100 \tag{13}$$

5.2.2 Results

Table 5 reports the values of the performance indicators $\bar{\varepsilon}$ (11) and $\overline{w}_{CI}$ (13) obtained with N_T = 5, 10, 20, 30, 40 and 50 samples by the LS, LS + LHS, IS and IS + LHS methods in Application 2 of Case study 1.

Table 5 here

It is seen that:

- the optimized Line Sampling methods (i.e., both LS and LS + LHS) provide more accurate and precise failure probability estimates than the other methods (i.e., both IS and IS + LHS): for example with N_T = 5, the mean percentage errors $\bar{\varepsilon}$ are 16.305, 16.198, 75.041 and 65.771, whereas the percentage 95% CI widths $\overline{w}_{CI}$ are 98.535, 92.477, 390.881 and 319.972 for the LS, LS + LHS, IS and IS + LHS methods, respectively;
- the use of LHS in combination with the optimized LS method does not affect significantly the *accuracy* of the failure probability estimates in this application: for example with N_T = 5, the mean percentage errors $\bar{\varepsilon}$ are 16.305 and 16.198 for the LS and LS + LHS methods respectively; conversely, the combination of LS and LHS *increases* the *precision* of the failure probability estimates: for example with N_T = 5, the percentage 95% CI widths $\overline{w}_{CI}$ are 98.535 and 92.477 for the LS and LS + LHS methods, respectively (a 6% increase in the precision of the estimate);
- the use of LHS in combination with the IS method *significantly increases* both the *accuracy* and the *precision* of the failure probability estimates: for example with N_T = 5, the mean percentage errors $\bar{\varepsilon}$ are 75.041 and 65.771, whereas the percentage 95% CI widths $\overline{w}_{CI}$ are 390.881 and 319.972 for the IS and IS + LHS methods, respectively.

Summing up, the results obtained confirm the possibility of achieving accurate and precise estimates of small failure probabilities by an optimized LS with a very low number N_T of samples drawn in a *nonlinear* (but *monotonous*) case study.

6 Case study 2: thermal-hydraulic passive system

This case study concerns the natural convection cooling in a Gas-cooled Fast Reactor (GFR) under a post-Loss Of Coolant Accident (LOCA) condition; the reactor is a 600-MW GFR cooled by helium flowing through separate channels in a silicon carbide matrix core [3].

A GFR decay heat removal configuration is shown schematically in Figure 4; in the case of a LOCA, the long-term heat removal is ensured by natural circulation in a given number N_{loops} of identical and parallel loops; only one of the N_{loops} loops is reported for clarity of the picture: the flow path of the cooling helium gas is indicated by the black arrows. The loop has been divided into $N_{sections}$ = 18 sections for numerical calculation; technical details about the geometrical and structural properties of these sections are not reported here for brevity: the interested reader may refer to [3].

In the present analysis, the average core power to be removed is assumed to be 18.7 MW, equivalent to about 3% of full reactor power (600 MW): to guarantee natural circulation cooling at this power level, a pressure of 1650 kPa in the loops is required in nominal conditions. Finally, the secondary side of the heat exchanger (i.e., item 12 in Figure 4) is assumed to have a nominal wall temperature of 90 °C [3].

Figure 4 here

6.1 Uncertainties

Uncertainties affect the modeling of passive systems. There are unexpected events, e.g. the failure of a component or the variation of the geometrical dimensions and material properties, which are random in nature. This kind of uncertainty, often termed aleatory [42]-[44], is not considered in this work. Additionally, there is incomplete knowledge on the properties of the system and the conditions in which the passive phenomena develop (i.e., natural circulation). This kind of uncertainty, often termed epistemic, affects the model representation of the passive system behaviour, in terms of both (*model*) uncertainty in the hypotheses assumed and (*parameter*) uncertainty in the values of the parameters of the model [16], [22], [45].

Only epistemic uncertainties are considered in this work. Epistemic parameter uncertainties are associated to the reactor power level, the pressure in the loops after the LOCA and the cooler wall temperature; epistemic model uncertainties are associated to the correlations used to calculate the Nusselt numbers and friction factors in the forced, mixed and free convection regimes. The consideration of these uncertainties leads to the definition of a vector $\boldsymbol{x} = \{x_j : j = 1, 2, ..., 9\}$ of nine uncertain model inputs, assumed described by normal distributions of known means and standard deviations (Table 6, [3]).

Table 6 here

6.2 Failure criteria of the T-H passive system

The passive decay heat removal system of Figure 4 fails to provide its safety function when the temperature of the coolant helium leaving the core (item 4 in Figure 4) exceeds either 1200 °C in the hot channel or 850 °C in the average channel: these values are expected to limit the fuel temperature to levels which prevent excessive release of fission gases and high thermal stresses in the cooler (item 12 in Figure 4) and in the stainless steel cross ducts connecting the reactor vessel and the cooler (items from 6 to 11 in Figure 4) [3]. Denoting by $T_{out,core}^{hot}(\boldsymbol{x})$ and $T_{out,core}^{avg}(\boldsymbol{x})$ the coolant outlet temperatures in the hot and average channels, respectively, the system failure event F can be written as follows:

$$F = \left\{\boldsymbol{x} : T_{out,core}^{hot}(\boldsymbol{x}) > 1200\right\} \cup \left\{\boldsymbol{x} : T_{out,core}^{avg}(\boldsymbol{x}) > 850\right\}. \quad (14)$$

The probability $P(F)$ of this event is $3.332 \cdot 10^{-4}$, obtained by standard MCS with N_T = 500000 samples drawn.

6.3 Application

The objective of the application is the estimation of the *small* functional failure probability $P(F)$ (i.e., $P(F) = 3.332 \cdot 10^{-4}$) of the T-H passive system described in Section 6 by means of LS with a very small number N_T of samples; more precisely, values of N_T = 5, 10, 20, 30, 40 and 50 are considered.

Justified by the results obtained in the previous case study, method D of Section 4.2 (i.e., the one based on the minimization of the variance of the LS failure probability estimator) is employed to estimate the important direction $\boldsymbol{\alpha}$ for LS. The ANN regression model used to this purpose is the classical three-layered feed-forward ANN: the number of inputs to the ANN regression model is equal to 9 (i.e., the number of uncertain inputs in Table 6 of Section 6.1), whereas the number of outputs is equal to 2 (i.e., the number of system variables of interest, the hot- and average-channel coolant outlet temperatures, as reported in Section 6.2). The number of nodes in the hidden layer has been set equal to 4 by trial and error. The ANN model is built using the sequential, two-step training algorithm described in Section 4.1: training sets of sizes N_{train}' = 50 and N_{train}'' = 70, validation sets of sizes N_{val}' = 10 and N_{val}'' = 10 and a test set of size N_{test} = 10 have been generated to train, validate and test the ANN model; thus, the *total* number of T-H code *runs* performed to generate the training, validation and test sets in this case is $N_{code,\alpha} = (N_{train}' + N_{val}' + N_{train}'' + N_{val}'' + N_{test}) = 50 + 10 + 70 + 10 + 10 = 150$.

The accuracies and precisions of the optimized LS, LS + LHS, IS and IS + LHS methods are also compared on the basis of the performance indicators $\bar{\varepsilon}$ (11) and $\overline{w}_{CI}$ (13) computed on S = 10 runs with N_T = 5, 10, 20, 30, 40 and 50 samples each. Table 7 reports the values obtained for the performance indicators $\bar{\varepsilon}$ (11) and $\overline{w}_{CI}$ (13).

Table 7 here

It is seen that:

- the optimized Line Sampling methods (i.e., both LS and LS + LHS) provide more accurate and precise functional failure probability estimates than the other methods considered (i.e., both IS and IS + LHS): for example with N_T = 5, the mean percentage errors $\bar{\varepsilon}$ are 16.045, 15.156, 84.801 and 38.671, whereas the percentage 95% CI widths $\overline{w}_{CI}$ are 84.175, 67.387, 386.026 and 212.079 for the LS, LS + LHS, IS and IS + LHS methods, respectively.
- the use of LHS in combination with the optimized LS method in this case *slightly increases* the *accuracy* of the functional failure probability estimates: for example with N_T = 10, the mean percentage errors $\bar{\varepsilon}$ are 12.547 and 7.378 for the LS and LS + LHS methods, respectively; moreover, the combination of LS and LHS in this case *strongly increases* the *precision* of the failure probability estimates: for example with N_T = 10, the percentage 95% CI widths $\overline{w}_{CI}$ are 84.175 and 67.387 for the LS and LS + LHS methods, respectively (a 20% increase in the precision of the estimate).
- the use of LHS in combination with the IS method *significantly increases* both the *accuracy* and the *precision* of the functional failure probability estimates: for example with N_T = 5, the mean percentage errors $\bar{\varepsilon}$ are 84.801 and 38.671, whereas the percentage 95% CI widths $\overline{w}_{CI}$ are 386.026 and 212.079 for the IS and IS + LHS methods, respectively;
- by way of example, the 95% CI associated to a standard MCS-based estimate of $P(F)$ with N_T = 100 is [0, 0.0296] and the corresponding percentage 95% CI width w_{CI} is 8793.8: this value is about *two orders of magnitude* larger (and conversely the precision is about two orders of magnitude lower) than that produced by LS with N_T = 5 samples: in other words, the precision of the optimized LS method is two order of magnitude larger than that of standard MCS even using a number of samples 20 times lower.

In summary, the results obtained confirm the previous finding regarding the possibility of achieving accurate and precise estimates of small failure probabilities by an optimized LS method with a very low number N_T of samples drawn; however, a much stronger conclusion can be drawn from this case study, regarding the actual feasibility of application of the method to the *realistic*, *nonlinear* and *non-monotonous* cases of practical interest in the reliability analysis of passive systems.

7 Discussion

In this paper, the Line Sampling (LS) method has been considered for improving the efficiency of Monte Carlo sampling in the estimation of the functional failure probability of a T-H passive system. A system designed to provide the safety function of natural convection cooling in a Gas-cooled Fast Reactor (GFR) after a Loss of Coolant Accident (LOCA) has been taken as reference case study.

Two relevant issues for the practical application of the LS method have been addressed:

1. the determination of the important direction for LS;
2. the reduction of the overall computational cost associated to the LS method in the estimation of the small functional failure probabilities characteristic of passive systems.

Concerning the first issue, the main contributions of the work presented and its related findings are (Case study 1):

- from a critical comparison of the methods currently available in the literature for the estimation of the LS important direction, it turns out that:
  - the technique based on Markov Chain Monte Carlo (MCMC) simulation produces *more accurate* and *precise* failure probability estimates than those provided by the design point and gradient methods;
  - the technique based on the identification of the design point performs *better* than the one based on gradient estimation.
- an Artificial Neural Network (ANN) regression model has been built using a *sequential*, *two-step* training algorithm on a *reduced-size* set of examples of the input/output nonlinear relationships underlying the original system model code; then, the ANN model has been used as a fast-running surrogate of the original system model code in the determination of the LS important direction:
  - the accuracy and precision of the estimates provided by the ANN-based method have been shown to be *comparable* to those produced by running the original system code: however, they have been obtained at a much *lower* computational effort;

    - conversely, when a *low* number of system model code simulations needs to be *a priori imposed* due to computational time limitations (which is the case of the long-running system model codes, typical of nuclear safety), the accuracy and precision of the failure probability estimates provided by the ANN-based method have been shown to be *consistently higher* than those produced by running the original system model code.
- a new technique has been proposed based on the minimization of the variance of the LS failure probability estimator; since the proposed method relies on the definition of the *optimal* LS important direction, it produces more accurate and precise failure probability estimates than those provided by all the techniques of literature, as clearly shown by the numerical results obtained.

Concerning the second issue, the main contributions of the work presented and the related findings are (Case studies 1 and 2):

- the performance of the LS method has been assessed in the estimation of a small failure probability (i.e., of the order of 10^{-4}) with a *reduced* number of samples drawn (i.e., ranging from 5 to 50). The results have demonstrated that accurate and precise estimates can be obtained even reducing the number of samples to below one hundred and even in *realistic*, *nonlinear* and *non-monotonous* case studies;
- the optimized Line Sampling method (i.e., both LS and the combination of LS and LHS) provide more accurate and precise failure probability estimates than both the IS and the combination of IS and LHS methods;
- the use of LHS in combination with the optimized LS method *slightly increases* the *accuracy* of the failure probability estimates and *strongly increases* the *precision* of the failure probability estimates;
- the use of LHS in combination with the IS method *significantly increases* both the *accuracy* and the *precision* of the failure probability estimates.

8 Conclusions

The findings of the work presented (summarized in the previous Section 7) suggest the adoption of the following methodology for the accurate and precise estimation of the (typically small) functional failure probability of T-H passive systems (modelled by *long-running*, *nonlinear* and *non-monotonous* T-H codes):

1. build an Artificial Neural Network (ANN) regression model using a *sequential*, *two-step* training algorithm on a *reduced* (e.g., around one hundred) number of examples of the input/output nonlinear relationships underlying the original system model code;
2. use the ANN model as a fast-running surrogate of the original system model code in the determination of the LS important direction; for this purpose, the technique proposed in this paper (based on the *minimization* of the variance of the LS failure probability estimator by means of Genetic Algorithms) is strongly suggested: since it relies on the definition of the *optimal* LS important direction, it produces more accurate and precise failure probability estimates than those provided by all the techniques of literature;
3. estimate the functional failure probability of the T-H passive system by means of Line Sampling with a *small* number of samples (e.g., few tens); the accuracy and precision of the estimates can be enhanced by combining Line Sampling with Latin Hypercube Sampling.

The outstanding performance of the *optimized* Line Sampling method presented in this paper in the estimation of very small failure probabilities makes it a rather attractive tool for passive system functional failure analyses and possibly one worth considering for extended adoption in full scale PRA applications, provided that the numerous possible accident scenarios and outcomes can be handled computationally in an efficient way.

References

[1] Mao H, Mahadevan S. Reliability analysis of creep–fatigue failure. Int. J. Fatigue 2000; 22: 789-797.

[2] Lu Z, Song S, Yue Z, Wang J. Reliability sensitivity method by Line Sampling. Structural Safety 2008; 30: 517-532.

[3] Pagani L, Apostolakis GE, Hejzlar P. The impact of uncertainties on the performance of passive systems. Nuclear Technology 2005; 149: 129-140.

[4] Fong CJ, Apostolakis GE, Langewish DR, Hejzlar P, Todreas NE, Driscoll MJ. Reliability analysis of a passive cooling system using a response surface with an application to the flexible conversion ratio reactor. Nuclear Engineering and Design 2009; 239:2660-2671.

[5] IAEA. Safety related terms for advanced nuclear plant. IAEA TECDOC-626, 1991.

[6] Nayak AK, Gartia MR., Antony A, Vinod G, Sinha RK. Passive system reliability analysis using the APSRA methodology. Nuclear Engineering and Design 2008; 238: 1430-1440.

[7] Nayak AK, Jain V, Gartia MR, Srivastava A, Prasad H, Anthony A, Gaikwad AJ, Bhatia SK, Sinha RK. Reliability assessment of passive isolation condenser system using APSRA methodology. Annals of Nuclear Energy 2008; 35: 2270-2279.

[8] Nayak AK, Jain V, Gartia MR, Prasad H, Anthony A, Bhatia SK, Sinha RK. Reliability assessment of passive isolation condenser system of AHWR using APSRA methodology. Reliability Engineering and System Safety 2009; 94: 1064-1075.

[9] Apostolakis GE. The concept of probability in safety assessment of technological systems. Science 1990; 250: 1359-1793.

[10] Burgazzi L. Reliability evaluation of passive systems through functional reliability assessment. Nuclear Technology 2003; 144: 145-151.

[11] Burgazzi L. Addressing the uncertainties related to passive system reliability. Progress in Nuclear Energy 2007; 49: 93-102.

[12] Bassi C, Marquès M. Reliability assessment of 2400 MWth gas-cooled fast reactor natural circulation decay heat removal in pressurized situations. Science and Technology of Nuclear Installations, Special Issue “Natural Circulation in Nuclear Reactor Systems”, Hindawi Publishing Corporation, Paper 87376, 2008.

[13] Mackay FJ, Apostolakis GE, Hejzlar P. Incorporating reliability analysis into the design of passive cooling systems with an application to a gas-cooled reactor. Nuclear Engineering and Design 2008; 238(1): 217-228.

[14] Mathews TS, Ramakrishnan M, Parthasarathy U, John Arul A, Senthil Kumar C. Functional reliability analysis of safety grade decay heat removal system of Indian 500 MWe PFBR. Nuclear Engineering and Design 2008; 238(9): 2369-2376.

[15] Mathews TS, Arul AJ, Parthasarathy U, Kumar CS, Ramakrishnan M, Subbaiah KV. Integration of functional reliability analysis with hardware reliability: An application to safety grade decay heat removal system of Indian 500 MWe PFBR. Annals of Nuclear Energy 2009; 36: 481-492.

[16] Patalano G, Apostolakis GE, Hejzlar P. Risk-informed design changes in a passive decay heat removal system. Nuclear Technology 2008; 163: 191-208.

[17] Arul AJ, Iyer NK, Velusamy K. Adjoint operator approach to functional reliability analysis of passive fluid dynamical systems. Reliability Engineering and System Safety 2009; 94: 1917-1926.

[18] Zio E, Pedroni N. Estimation of the functional failure probability of a thermal-hydraulic passive systems by means of Subset Simulation. Nuclear Engineering and Design 2009; 239: 580-599.

[19] Zio E, Pedroni N. Functional failure analysis of a thermal-hydraulic passive system by means of Line Sampling. Reliability Engineering and System Safety 2009; 94(11): 1764-1781.

[20] Schueller GI. On the treatment of uncertainties in structural mechanics and analysis. Computers and Structures 2007; 85: 235-243.

[21] Au SK, Beck JL. Importance sampling in high dimensions. Structural Safety 2003; 25(2): 139-163.

[22] Cacuci DG, Ionescu-Bujor M. A comparative review of sensitivity and uncertainty analysis of large scale systems – II: Statistical methods. Nuclear Science and Engineering 2004; 147: 204-217.

[23] Helton JC, Davis FJ. Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. Reliability Engineering and System Safety 2003; 81: 23-69.

[24] Helton JC, Sallaberry C. Computational implementation of sampling-based approaches to the calculation of expected dose in performance assessments for the proposed high-level radioactive waste repository at Yucca Mountain, Nevada. Reliability Engineering and System Safety 2009; 94: 699-721.

[25] Koutsourelakis PS, Pradlwarter HJ, Schueller GI. Reliability of structures in high dimensions, Part I: algorithms and application. Probabilistic Engineering Mechanics 2004; 19: 409-417.

[26] Pradlwarter HJ, Pellissetti MF, Schenk CA, Schueller GI, Kreis A, Fransen S, Calvi A, Klein M. Realistic and efficient reliability estimation for aerospace structures. Computer Methods in Applied Mechanics and Engineering 2005; 194: 1597-1617.

[27] Schueller GI, Pradlwarter HJ, Koutsourelakis PS. A critical appraisal of reliability estimation procedures for high dimensions. Probabilistic Engineering Mechanics 2004; 19: 463-474.

[28] Pradlwarter HJ, Schueller GI, Koutsourelakis PS, Charmpis DC. Application of line sampling simulation method to reliability benchmark problems. Structural Safety 2007; 29: 208-221.

[29] Schueller GI, Pradlwarter HJ. Benchmark study on reliability estimation in higher dimensions of structural systems – An overview. Structural Safety 2007; 29(3): 167-182.

[30] Bishop CM. Neural Networks for pattern recognition. Oxford University Press; 1995.

[31] Zio E, Pedroni N. Nuclear passive system reliability assessment by an optimized Line Sampling method. Submitted to IEEE Transactions on Reliability, 2010.

[32] Olsson A, Sabdberg G, Dahlblom O. On Latin hypercube sampling for structural reliability analysis. Structural Safety 2003; 25: 47-68.

[33] Marquès M, Pignatel JF, Saignes P, D' Auria F, Burgazzi L, Müller C, Bolado-Lavin R, Kirchsteiger C, La Lumia V, Ivanov I. Methodology for the reliability evaluation of a passive system and its integration into a probabilistic safety assessment. Nuclear Engineering and Design 2005; 235: 2612-2631.

[34] Huang B, Du X. A robust design method using variable transformation and Gauss-Hermite integration. International Journal for Numerical Methods in Engineering 2006; 66: 1841-1858.

[35] Metropolis N, Rosenbluth AW, Rosenbluth MN, Taller AH. Equations of state calculations by fast computing machines. Journal of Chemical Physics 1953; 21(6): 1087-1092.

[36] Valdebenito MA, Pradlwarter HJ, Schueller GI. The role of the design point for calculating failure probabilities in view of dimensionality and structural nonlinearities. Structural Safety 2010; 32(2): 101-111.

[37] Konak A, Coit DW, Smith AE. Multi-objective optimization using genetic algorithms: A tutorial. Reliability Engineering and System Safety 2006; 91(9): 992-1007.

[38] Marseguerra M, Zio E, Martorell S. Basics of genetic algorithms optimization for RAMS applications. Reliability Engineering and System Safety 2006; 91(9): 977-991.

[39] Ahammed M, Malchers M. Gradient and parameter sensitivity estimation for systems evaluated using Monte Carlo analysis. Reliability Engineering and System Safety 2006; 91: 594-601.

[40] Fu M. Stochastic gradient estimation. In: Henderson SG, Nelson BL, editors. Handbook on Operation Research and Management Science: Simulation. Elsevier; 2006.

[41] Rumelhart DE, Hinton GE, Williams RJ. Learning internal representations by error back-propagation. In: Rumelhart DE, McClelland JL, editors. Parallel distributed processing: exploration in the microstructure of cognition (vol. 1). Cambridge (MA): MIT Press; 1986.

[42] NUREG-1150, 1990. Severe accident risk: an assessment for five US nuclear power plants, US Nuclear Regulatory Commission.

[43] Helton JC 1998. Uncertainty and sensitivity analysis results obtained in the 1996 performance assessment for the waste isolation power plant, SAND98-0365, Sandia National Laboratories.

[44] USNRC 2002. An approach for using probabilistic risk assessment in risk-informed decisions on plant-specific changes to the licensing basis. NUREG-1.174 – Revision 1, US Nuclear Regulatory Commission, Washington, DC.

[45] Helton JC, Johnson JD, Sallaberry CJ, Storlie CB. Survey on sampling-based methods for uncertainty and sensitivity analysis. Reliability Engineering and System Safety 2006; 91: 1175-1209.

FIGURES

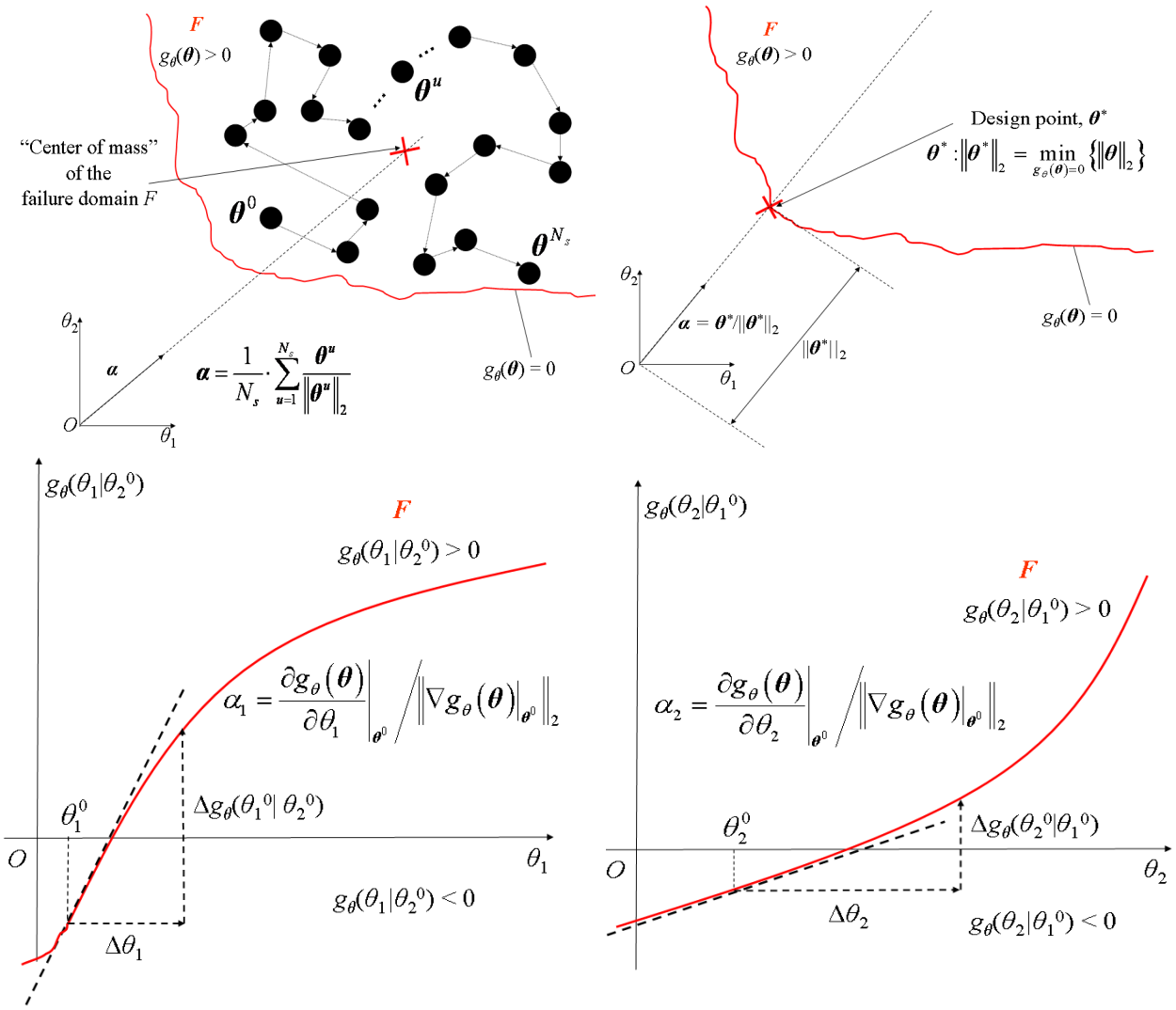

Figure 1. Methods for estimating the Line Sampling important unit vector $\boldsymbol{\alpha}$*. Top, left: normalized "center of mass" of the failure domain F in the standard normal space [25]; top, right: direction of the design point of the problem in the standard normal space [27], [36]; bottom, left and right: normalized gradient of the PF* $g_\theta(\cdot)$ *evaluated at a selected point* $\boldsymbol{\theta}^0$ *(e.g., the nominal point) in the standard normal space [26]*

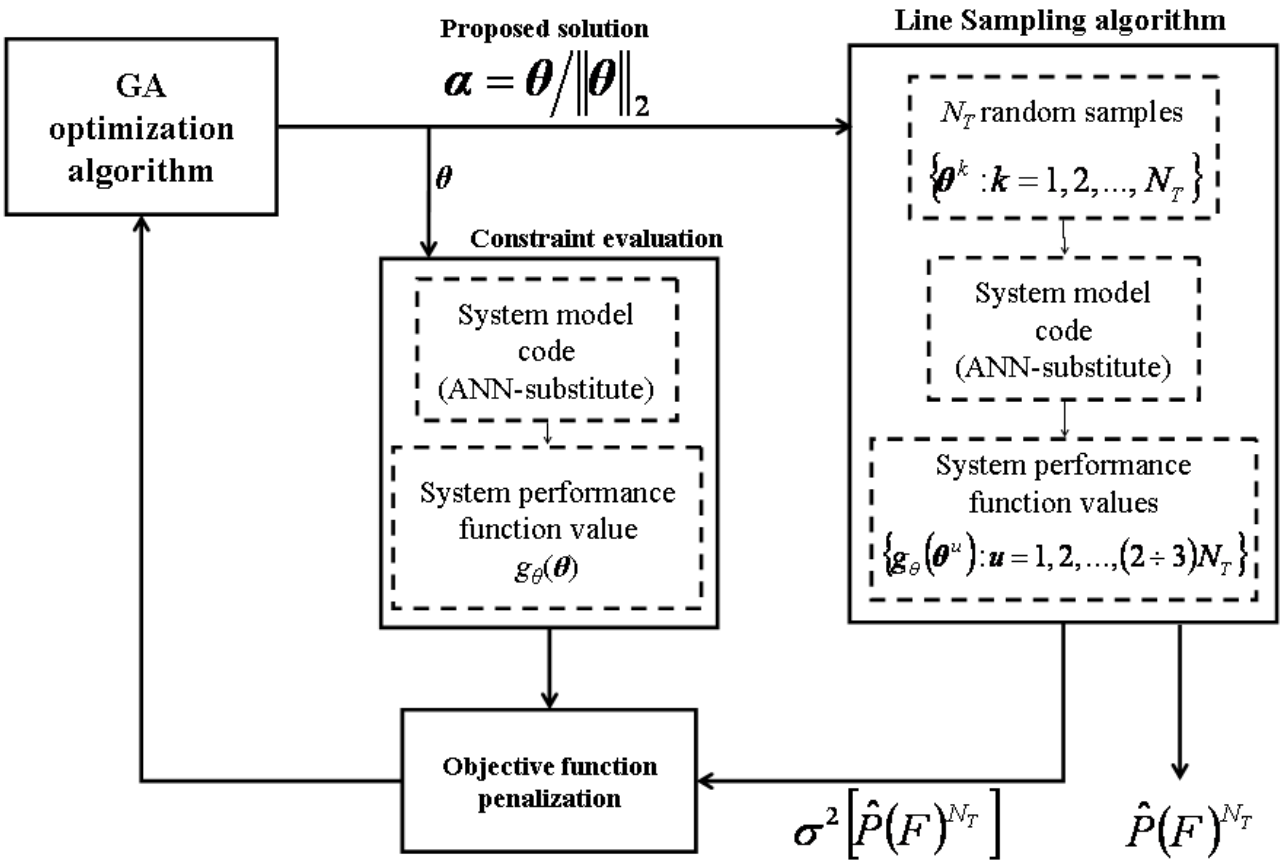

Figure 2. Proposed method for estimating the LS important direction $\boldsymbol{\alpha}$*: minimization of the variance* $\sigma^2\left[\hat{P}(F)^{N_T}\right]$ *of the LS failure probability estimator* $\hat{P}(F)^{N_T}$

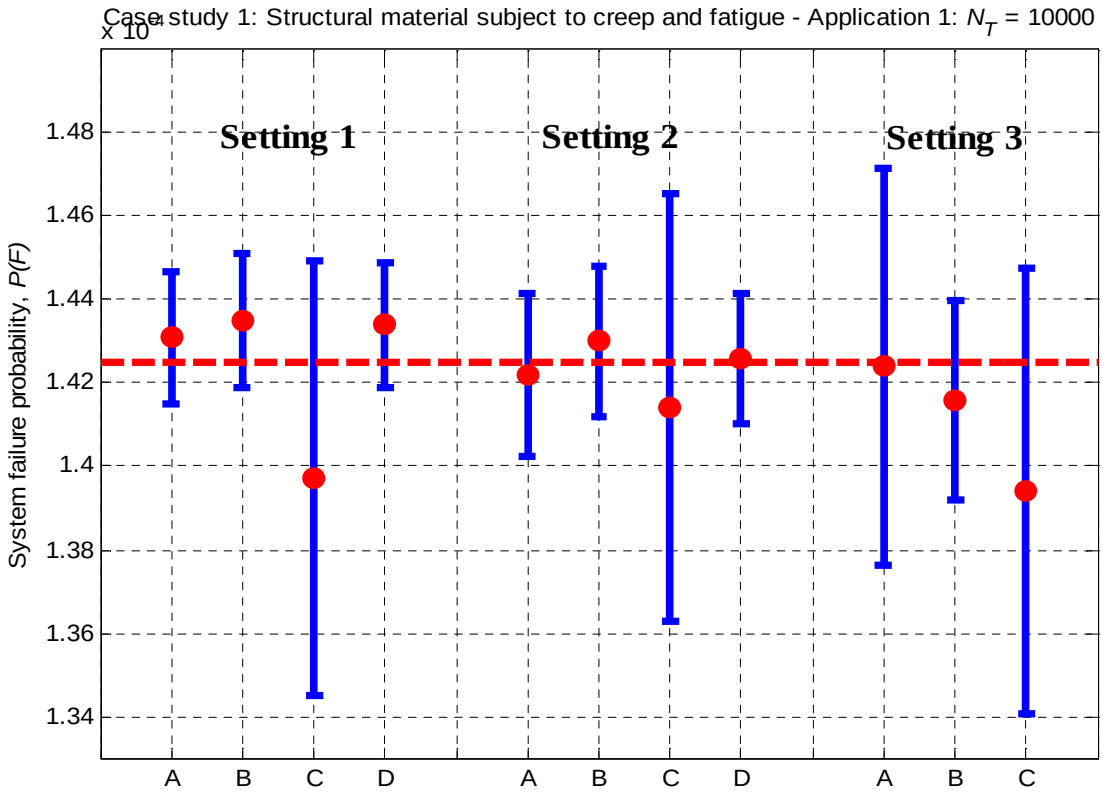

Figure 3. Point estimates $\hat{P}(F)^{N_T}$ *(dots) of the failure probability P(F) obtained with* N_T *= 10000 samples in settings 1, 2 and 3 (Table 3) of Application 1 of Case study 1, along with the corresponding 95% CIs (bars) and the true (i.e., reference) value of the system failure probability P(F) (i.e.,* $P(F) = 1.425 \cdot 10^{-4}$*) (dashed line)*

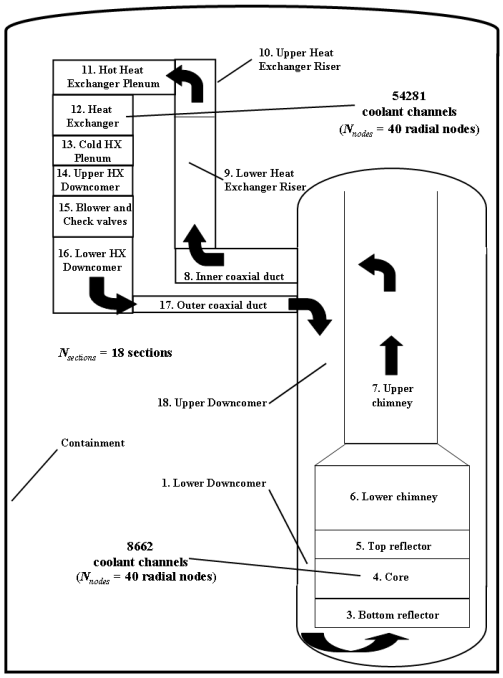

Figure 4. Schematic representation of one loop of the 600-MW GFR passive decay heat removal system [3]

TABLES

Methods of literature		
Concept	**Evaluations to be performed**	**Computational tools adopted**
"Center of mass" of F (Section 4.1.1)	- Evaluation of the performance function $g_\theta(\boldsymbol{\theta})$ during MCMC to verify if $\boldsymbol{\theta}$ belongs to the failure domain F, i.e., if $g_\theta(\boldsymbol{\theta}) > 0$	Original system model code ANN
Design point (Section 4.1.2)	- Minimization of the distance $\|\boldsymbol{\theta}\|_2$ in (4)	GA
	- Evaluation of the performance function $g_\theta(\boldsymbol{\theta})$ to verify if $\boldsymbol{\theta}$ is a feasible solution to (4), i.e., if $\boldsymbol{\theta}$ belongs to the failure surface $g_\theta(\boldsymbol{\theta}) = 0$	Original system model code ANN
Gradient (Section 4.1.3)	- Evaluation of the performance function $g_\theta(\boldsymbol{\theta})$ to estimate the gradient $\nabla g_\theta(\boldsymbol{\theta})$ (5) by numerical differentiation	Original system model code ANN
Method proposed in this paper		
Concept	**Function to be performed**	**Computational tools adopted**
Variance minimization (Section 4.2)	- Minimization of the variance $\sigma^2\left[\hat{P}(F)^{N_T}\right]$ of the LS failure probability estimator $\hat{P}(F)^{N_T}$	GA
	- Calculation of the variance $\sigma^2\left[\hat{P}(F)^{N_T}\right]$ of the LS failure probability estimator $\hat{P}(F)^{N_T}$	LS algorithm
	- Evaluation of the performance function $g_\theta(\boldsymbol{\theta})$ for the estimation of the failure probability $\hat{P}(F)^{N_T}$ and its variance $\sigma^2\left[\hat{P}(F)^{N_T}\right]$ during the LS simulation	ANN
	- Evaluation of the performance function $g_\theta(\boldsymbol{\theta})$ to verify if $\boldsymbol{\theta}$ is a feasible solution to (6), i.e., if $\boldsymbol{\theta}$ belongs to the failure domain F (where $g_\theta(\boldsymbol{\theta}) > 0$)	ANN

Table 1. Summary of the methods employed in this work for estimating the LS important direction $\boldsymbol{\alpha}$

Name	Shape	Mean, μ	Standard deviation, σ (% of μ)
N_c, x_1	Log-Normal	5490	20%
N_f, x_2	Log-Normal	17100	20%
n_c, x_3	Log-Normal	549	20%
n_f, x_4	Log-Normal	4000	20%
θ_1, x_5	Normal	0.42	20%
θ_2, x_6	Normal	6	20%

Table 2. Shapes and parameters (i.e., mean μ and standard deviation σ) of the probability distribution functions associated to the uncertain variables $\{x_j: j = 1, 2, ..., 6\}$ of the probabilistic model (7) for creep and fatigue in structural materials [2]

Experimental settings considered for the determination of the important direction $\boldsymbol{\alpha}$					
Setting number		**Method used to estimate $\boldsymbol{\alpha}$**	**Model used to evaluate the system performance function $g_\theta(\cdot)$**	**Number of system performance function evaluations, N_α**	**Number of system model code runs, $N_{code,\alpha}$**
1	**A**	MCMC	Original system model code	10000	10000
	B	Design point		$\leq$ 10000*	$\leq$ 10000*
	C	Gradient		10000	10000
				Not available**	Not available**
2	**A**	MCMC	ANN (N_{train} = 50; N_{val} = 10; N_{train}''= 100; N_{val}''= 20; N_{test} = 10)	10000	(N_{train} + N_{val} + N_{train}'' + N_{val}'' + N_{test}) = 190
	B	Design point		$\leq$ 10000*	(N_{train} + N_{val} + N_{train}'' + N_{val}'' + N_{test}) = 190
	C	Gradient		10000	(N_{train} + N_{val} + N_{train}'' + N_{val}'' + N_{test}) = 190
	D	Variance minimization		Not available**	(N_{train} + N_{val} + N_{train}'' + N_{val}'' + N_{test}) = 190
3	**A**	MCMC	Original system model code	(N_{train} + N_{val} + N_{train}'' + N_{val}'' + N_{test}) = 190	(N_{train} + N_{val} + N_{train}'' + N_{val}'' + N_{test}) = 190
	B	Design point		$\leq$ (N_{train} + N_{val} + N_{train}'' + N_{val}'' + N_{test}) = 190*	$\leq$ (N_{train} + N_{val} + N_{train}'' + N_{val}'' + N_{test}) = 190*
	C	Gradient		(N_{train} + N_{val} + N_{train}'' + N_{val}'' + N_{test}) = 190	(N_{train} + N_{val} + N_{train}'' + N_{val}'' + N_{test}) = 190

Table 3. Different experimental settings 1, 2 and 3 considered for Application 1 of Case study 1. The three settings differ by i) the method for determining the important direction $\boldsymbol{\alpha}$; ii) the model for evaluating the system performance function $g_\theta(\cdot)$; iii) the number N_α of system performance evaluations and iv) the total number $N_{code,\alpha}$ of actual runs of the original system model code required by the whole process of determination of the LS important direction $\boldsymbol{\alpha}$

* The number N_α of system performance evaluations depends on the speed of convergence of the GA optimization algorithm

** The number N_α of system performance evaluations depends on the speed of convergence of the GA optimization algorithm and on the number N_T of samples drawn in step 2. of Section 4.2

Case study 1: Structural material subject to creep and fatigue - Application 1: N_T = 10000		
Setting 1		
Method	**ε [%]**	**w_{CI} [%]**
A	0.421	2.222
B	0.702	2.282
C	1.965	7.323
Setting 2		
Method	**ε [%]**	**w_{CI} [%]**
A	0.211	2.723
B	0.351	2.516
C	0.772	7.199
D	0.070	2.204
Setting 3		
Method	**ε [%]**	**w_{CI} [%]**
A	0.070	6.697
B	0.632	5.345
C	2.175	7.502

Table 4. Values of the performance indicators ε and w_{CI} obtained with N_T = 10000 samples in settings 1, 2 and 3 (Table 3) of Application 1 of Case study 1

Case study 1: Structural material subject to creep and fatigue - Application 2		
LS		
Sample size, N_T	**$\bar{\varepsilon}$ [%]**	**$\overline{w}_{CI}$ [%]**
5	16.305	98.535
10	11.506	68.619
20	8.663	52.973
30	7.130	39.595
40	6.373	34.321
50	5.654	29.361
LS + LHS		
Sample size, N_T	**$\bar{\varepsilon}$ [%]**	**$\overline{w}_{CI}$ [%]**
5	16.198	92.477
10	11.504	61.820
20	8.349	48.107
30	7.111	37.655
40	6.084	32.094
50	5.266	27.393
IS		
Sample size, N_T	**$\bar{\varepsilon}$ [%]**	**$\overline{w}_{CI}$ [%]**
5	75.041	390.88
10	55.433	285.60
20	39.523	201.56
30	32.014	160.43
40	27.349	140.39
50	25.537	135.03
IS + LHS		
Sample size, N_T	**$\bar{\varepsilon}$ [%]**	**$\overline{w}_{CI}$ [%]**
5	65.771	319.97
10	33.745	219.21
20	25.321	161.70
30	22.437	150.84
40	19.826	105.93
50	17.593	90.315

Table 5. Values of the performance indicators $\bar{\varepsilon}$ (11) and $\overline{w}_{CI}$ (13) obtained with N_T = 5, 10, 20, 30, 40 and 50 samples by the LS, LS + LHS, IS and IS + LHS methods in Application 2 of Case study 1

	Name	Mean, μ	Standard deviation, σ (% of μ)
Parameter uncertainty	Power (MW), x_1	18.7	1%
	Pressure (kPa), x_2	1650	7.5%
	Cooler wall temperature (°C), x_3	90	5%
Model uncertainty	Nusselt number in forced convection, x_4	1	5%
	Nusselt number in mixed convection, x_5	1	15%
	Nusselt number in free convection, x_6	1	7.5%
	Friction factor in forced convection, x_7	1	1%
	Friction factor in mixed convection, x_8	1	10%
	Friction factor in free convection, x_9	1	1.5%

Table 6. Epistemic uncertainties considered for the 600-MW GFR passive decay heat removal system of Figure 4 [3]

Case study 2 – Thermal-hydraulic passive system		
LS		
Sample size, N_T	$\bar{\varepsilon}$ **[%]**	$\overline{w}_{CI}$ **[%]**
5	16.045	84.175
10	12.547	58.292
20	8.313	39.095
30	7.459	34.832
40	5.466	27.728
50	3.848	19.324
LS + LHS		
Sample size, N_T	$\bar{\varepsilon}$ **[%]**	$\overline{w}_{CI}$ **[%]**
5	15.156	67.387
10	7.378	31.264
20	6.179	26.682
30	5.486	26.419
40	3.092	16.345
50	2.373	13.590
IS		
Sample size, N_T	$\bar{\varepsilon}$ **[%]**	$\overline{w}_{CI}$ **[%]**
5	84.801	386.026
10	45.982	223.828
20	36.499	154.531
30	33.846	115.521
40	21.790	93.308
50	18.281	85.22
IS + LHS		
Sample size, N_T	$\bar{\varepsilon}$ **[%]**	$\overline{w}_{CI}$ **[%]**
5	38.671	212.079
10	30.174	122.647
20	22.943	106.231
30	21.195	100.854
40	18.461	73.522
50	16.916	71.069

Table 7. Values of the performance indicators $\bar{\varepsilon}$ and $\overline{w}_{CI}$ obtained with N_T = 5, 10, 20, 30, 40 and 50 samples by the LS, LS + LHS, IS and IS + LHS methods in Case study 2