An Adaptive Metamodel-Based Subset Importance Sampling approach for the assessment of the functional failure probability of a thermal-hydraulic passive system

Nicola Pedroni, Enrico Zio

Applied Mathematical Modelling 48 (2017) 269–288

Contents lists available at ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

An Adaptive Metamodel-Based Subset Importance Sampling approach for the assessment of the functional failure probability of a thermal-hydraulic passive system

Nicola Pedroni [a,*], Enrico Zio [a,b]

[a] Chair on Systems Science and the Energetic Challenge, Fondation Electricité de France (EdF), Laboratoire Genie Industriel (LGI), Université Paris-Saclay-Ecole CentraleSupelec, Grande Voie des Vignes, 92290 Chatenay-Malabry, France
[b] Energy Department, Politecnico di Milano, Via Ponzio 34/3, 20133 Milano, Italy

ARTICLE INFO

Article history:
Received 9 June 2015
Revised 8 March 2017
Accepted 3 April 2017
Available online 7 April 2017

Keywords:
Nuclear passive safety system
Small failure probability
Efficient Monte Carlo
Subset Simulation
Non-parametric importance sampling
Adaptive metamodeling

ABSTRACT

An Adaptive Metamodel-Based Subset Importance Sampling (AM-SIS) approach, previously developed by the authors, is here employed to assess the (small) functional failure probability of a thermal-hydraulic (T-H) nuclear passive safety system. The approach relies on an iterative Importance Sampling (IS) scheme that efficiently couples the powerful characteristics of Subset Simulation (SS) and fast-running Artificial Neural Networks (ANNs). In particular, SS and ANNs are intelligently employed to build and progressively refine a fully nonparametric estimator of the ideal, zero-variance Importance Sampling Density (ISD), in order to: (i) increase the robustness of the failure probability estimates and (ii) decrease the number of expensive T-H simulations needed (together with the related computational burden).

The performance of the approach is thoroughly compared to that of other efficient Monte Carlo (MC) techniques on a case study involving an advanced nuclear reactor cooled by helium.

1. Introduction

The nuclear industry is characterized by levels of risk that are among the lowest with respect to any engineering application, as requested by the strong public perception. In the past and current generations of reactor designs, these low levels of risk are typically achieved by resorting mostly to active safety systems. On the other hand, the most innovative reactor concepts, such as those of Generation IV, aim at relying almost entirely on passive safety systems, thus becoming as much independent as possible from any kind of external input, especially energy [1]. Moreover, the simplicity of the design of these systems is such that mechanical failures become very unlikely to happen, in contrast to active systems [2–7].

Yet, uncertainties in the operation of these systems (due to not fully understood physical phenomena and scarce, if not null, operating experience [8,9]) may give rise to functional failures. A passive system fails to perform its function (e.g., decay heat removal by natural circulation) due to deviations from its expected behavior, which lead the load imposed on the system to overcome its capacity [10–24]. A classical approach to estimating such *functional failure* probability stems from the deterministic modeling of the Thermo-Hydraulics (T-H) of the passive system by a (possibly long-running) computer code

* Corresponding author.
E-mail addresses: nicola.pedroni@centralesupelec.fr, nicolapedroni@gmail.com (N. Pedroni), enrico.zio@polimi.it, enrico.zio@centralesupelec.fr (E. Zio).

http://dx.doi.org/10.1016/j.apm.2017.04.003
0307-904X/

and the subsequent Monte Carlo (MC) propagation of the uncertainties, described by properly identified Probability Density Functions (PDFs), to some model outputs with respect to which the failure event is defined. The propagation is based on repeated computer runs (or simulations) of the T-H code in correspondence of different sets of the uncertain input values sampled from their PDFs [25–36].

In practice, a passive safety system is obviously designed so that its failure is a rare event. As a consequence, its probability of failure is quite small (e.g., about 10^{-5} or less) and its estimation may require a large amount of T-H computer code runs [37]. Since each simulation of the detailed, mechanistic T-H model may take several hours in some cases [33], the standard MC simulation-based procedure typically requires significant computational efforts. More efficient techniques must then be sought [38,39].

Two families of methods are typically employed to address these issues. On one side, fast-running surrogate regression models (also called response surfaces or metamodels) can be properly built to mimic the response of the original T-H model code: since the metamodel is, in general, much faster to be evaluated, the problem of long computing times is circumvented. Several kinds of surrogate metamodels have been recently applied to safety-related nuclear, structural and hydrogeological problems, including polynomial Response Surfaces (RSs) [40–43], polynomial chaos expansions [44–47], stochastic collocations [48,49], Artificial Neural Networks (ANNs) [50–53], Support Vector Machines (SVMs) [54,55] and kriging [56–61].

On the other side, advanced Monte Carlo Simulation methods can be adopted to realize robust estimations by means of a limited amount of random samples [62–64]. This class of methods includes Stratified Sampling [65,66], Subset Simulation (SS) [67–74], the Response Conditioning Method (RCM) [75,76] for structural reliability analysis (also developed from SS and employing approximate solutions), Line Sampling (LS) [77–84] and splitting methods [85–89]. However, the advanced MCS methods that is adopted more frequently is perhaps Importance Sampling (IS), where the original PDF of the uncertain inputs is replaced by a proper Importance Sampling Density (ISD): this favors the MC samples to lie close to the failure region, thus artificially increasing the frequency of the (rare) failure event [90–92]. In this respect, it is possible to show that there is an optimal (ideal) ISD that makes the standard deviation of the MC estimator equal to zero. On the other hand, this optimal ISD cannot be used in practice, because its mathematical expression contains the (a priori unknown) failure probability itself. Within this context, several techniques in various fields of research have been proposed to *approximate* the "ideal" optimal ISD: see, e.g., the Adaptive Kernel (AK) [93,94], the Cross-Entropy (CE) [95–97], the Variance Minimization (VM) [98] and the Markov Chain Monte Carlo-Importance Sampling (MCMC-IS) [99] methods.

In the present paper, the Adaptive Metamodel-based Subset Importance Sampling (AM-SIS) method, *originally* developed by the authors [100], is adopted. In the approach, the characteristics of several advanced computational methods of literature are efficiently *combined*. In particular, SS and ANN metamodels are coupled within an adaptive IS scheme in the following two stages [99,100]:

1. a random sample *approximately* distributed as the ideal, zero-variance ISD is created by resorting to the Subset Simulation (SS) technique. Notice that at this step we substitute the expensive T-H code with an ANN metamodel (*adaptively* refined in the areas close to the failure region by means of the samples *iteratively* produced by SS), with the aim of reducing the related computational burden. Then, we adopt a *fully nonparametric* PDF (proposed in Ref. [99] and relying on the Gibbs Sampler [101]) to "fit" the samples thereby generated and obtain an estimator for the optimal ISD;
2. we employ the quasi-optimal ISD estimator built at step (1) above to perform Importance Sampling (IS) and assess the functional failure probability of the T-H passive system.

Notice that the idea of "interlacing" metamodels with efficient MCS schemes has been already proposed in the literature for decreasing the computational burden associated to the quantification of small probabilities: see, e.g., Refs. [102–113]. In these works, the metamodel is built and progressively refined (by means of samples intelligently generated by the advanced MC technique) until a desired level of accuracy and precision in the failure probability estimate is obtained.

The main contributions of the present work with respect to the reference paper by Ref. [99] are the following:

- we adopt Subset Simulation (SS) for the generation of samples approximately distributed according to the zero-variance ISD;
- we employ ANNs to decrease the computational effort related to the approximation of the zero-variance ISD: in particular, we train these metamodels through an *adaptive* algorithm that makes a smart use of the samples produced by SS with the objective of progressively ameliorating the ANN accuracy around the failure region;
- we assess the efficiency of AM-SIS in the estimation of small functional failure probabilities with a *very small* number of calls to the T-H code (e.g., of the order of few tens or hundreds): this is relevant for those practical cases where a single simulation needs several hours to run;
- we extensively and systematically compare the performance of AM-SIS to that of numerous other advanced probabilistic simulation methods of literature.

In addition, the novel contents of this paper in relation to the previous article by the authors [100] are the following:

- in Ref. [100], the focus of the comparisons between AM-SIS and other MC schemes (in particular, standard MCS, LHS [65], IS [90–94] and LS [77–84]) is *exclusively* put on the efficiency of stage (2) of the algorithm, i.e., of random importance sampling for failure probability estimation. In this work, instead, comparisons are made *also* to other methods that combine SS and metamodels and that are very close to stage (1) of our approach. In particular, the following techniques

are considered: (i) the ^{2}SMART algorithm [105], which combines SS and SVMs in an original active learning scheme, where few training points are placed sequentially in the uncertain variable space, in order to build a *sequence* of SVM metamodels as accurate surrogates of the SS intermediate thresholds; (ii) the Neural Network-based Subset Simulation (SS-NN) method [113], in which information is transferred from the MCMC algorithm to the ANN training process in order to efficiently construct the metamodel in a sequence of "*moving windows*";
- also the performance of stage (1) of the AM-SIS method is assessed *more thoroughly* by carrying out a *larger* set of comparisons (e.g., by considering different sample sizes for failure probability estimation);
- we have modified and improved the technique adopted to identify, select and "distribute" in the entire uncertain input space the (training) samples used to adaptively refine the ANN metamodel through the iterations of the AM-SIS algorithm.

All the analyses are performed with regards to the same case study considered in Ref. [100], which concerns a Gas-cooled Fast Reactor (GFR) cooled by naturally circulating helium after a Loss Of Coolant Accident (LOCA) (modified from Ref. [26]).

The remainder of the paper is organized as follows. In Section 2, the concepts of functional failure analysis for T-H passive systems are synthetically summarized. Section 3, a quite detailed description of the AM-SIS technique proposed in this work is given. Section 4 presents the case study concerning the passive cooling of a GFR, and the corresponding results are shown and discussed in Section 5. Some conclusions are drawn in the last Section. Finally, for the sake of paper self-consistency, the Appendix reports thorough mathematical details concerning the construction of the fully nonparametric ISD proposed in Ref. [99] and based on the well-known Gibbs Sampler [101].

2. The qualitative and quantitative steps of functional failure analysis of T-H passive systems

The basic steps of the qualitative and quantitative phase of the functional failure analysis of a T-H passive system are [25]:

1. Characterize the accident scenario in which the passive system under consideration operates;
2. Define the function that the passive system is expected to perform, e.g. decay heat removal, vessel cooling and depressurization, etc.
3. Identify the design parameters values (e.g., power level, system pressure, heat exchanger initial wall temperature, etc.) related to the reference configuration of the passive system and the corresponding nominal function.
4. Identify the possible failure modes of the passive system for the accident scenario under consideration.
5. Build a mechanistic, best-estimate (typically long-running) T-H model to simulate the passive system behavior in the accident scenario of interest and perform best estimate calculations.
6. Identify the factors of *uncertainty* in the results of the best estimate T-H calculations. In the present work, we consider only *epistemic* uncertainties (due to the limited knowledge on some phenomena and processes – e.g., models, parameters and correlations used in the T-H analysis – and to the paucity of the related operational and experimental data available), as done also in the previous article by the authors [100].
7. Quantify the uncertainties identified at the previous step by proper probability distributions.
8. Propagate the (epistemic) uncertainties through the deterministic T-H model in order to assess the probability of *functional failure* of the passive system. Formally, let: $\mathbf{x}=\{x_1, x_2, ..., x_j, ..., x_{n_i}\}$ be the vector of the relevant uncertain (input) model parameters/variables; $\boldsymbol{\mu}_{\mathbf{y}}(\mathbf{x})$ be a (possibly nonlinear) deterministic function representing the complex, computationally expensive T-H code; $\boldsymbol{y}=\{y_1, y_2, ..., y_l, ..., y_{n_o}\}=\boldsymbol{\mu}_{\mathbf{y}}(\mathbf{x})$ be the vector of the (uncertain) T-H model outputs of interest; $Y(\mathbf{x})$ be a single-valued indicator of the performance of the passive system (e.g., the fuel peak cladding temperature); and α_Y a limit value that represents the system failure criterion.[1] For illustrating purposes, we suppose that the passive system functions correctly as long as $Y(\mathbf{x}) < \alpha_Y$. The probability $P(F)$ of system functional failure can, then, be expressed by the multidimensional integral:

$$P(F) = \iint ... \int I_F(\mathbf{x}) q(\mathbf{x}) d\mathbf{x}, \tag{1}$$

where $q(\cdot)$ is the joint Probability Density Function (PDF) describing the uncertainty in the parameters $\mathbf{x}$, F is the failure region (where $Y(\mathbf{x}) \geq \alpha_Y$) and $I_F(\cdot)$ is an indicator function such that $I_F(\boldsymbol{x}) = 1$, if $\boldsymbol{x} \in F$ (i.e., if $Y(\mathbf{x}) \geq \alpha_Y$) and $I_F(\boldsymbol{x}) = 0$, otherwise (i.e., if $Y(\mathbf{x}) < \alpha_Y$).

The MCS procedure for estimating (1) entails that a typically *large* number N_T of samples of the values of the system parameters $\boldsymbol{x}$ be drawn from the corresponding probability distributions and used to evaluate ($\boldsymbol{y}(\boldsymbol{x})$ and) $Y(\boldsymbol{x})$ by running the T-H code. An estimate $\hat{P}(F)^{N_T}$ of the probability of failure $P(F)$ can, then, be computed by dividing the number of times that $Y(\mathbf{x}) > \alpha_Y$ by the total number of samples N_T. The cost related to this process can be impractical, when the running time for each T-H model simulation (i.e., for each evaluation of $Y(\boldsymbol{x})$) takes several hours (which is often the case for T-H passive systems). In order to tackle these issues, we adopt the Adaptive Metamodel-based Subset Importance Sampling

[1] Note that the choice of a *single-valued* performance function does not reduce the generality of the approach, because any multidimensional vector of physical quantities (i.e., the vector $\boldsymbol{y}$ of the outputs of the T-H code in this case) can be conveniently re-expressed as a scalar parameter by resorting to suitable min-max transformations: see Refs. [67,68,72] for details.

(AM-SIS) method (proposed by the authors in Ref. [100]), which combines the characteristics of two advanced MCS schemes (i.e., Importance Sampling-IS and Subset Simulation-SS) and a metamodel (i.e., an Artificial Neural Network-ANN) (Section 3).

3. The Adaptive Metamodel-Based Subset Importance Sampling (AM-SIS) method

The Adaptive Metamodel-based Subset Importance Sampling (AM-SIS) method, proposed in Ref. [100] and here employed, basically relies on an *iterative* Importance Sampling (IS) scheme, whereby the Importance Sampling Density (ISD) is constructed and *adaptively* refined so as to favor the MC samples to be near the failure region, thus increasing the frequency of occurrence of the (rare) failure event [90–92]. In more detail, it consists of two main steps [98,99,118]:

1. We build an estimator $\hat{g}^*(\mathbf{x})$ of the so-called "optimal ISD" $g^*(\mathbf{x}) = \frac{I_F(\mathbf{x})q(\mathbf{x})}{P(F)}$, i.e., the ISD that minimizes the variance of the MC estimator $\hat{P}(F)^{N_T}$ [90–92,107] (Section 3.1);
2. We perform Importance Sampling (IS), in which the estimator $\hat{g}^*(\mathbf{x})$ built at step 1 above is employed as an ISD in order to assess the functional failure probability $P(F)$ (1) of the T-H passive system (Section 3.2).

3.1. Adaptive approximation of the ideal, zero-variance ISD by means of Subset Simulation and metamodels

Many approaches have been proposed to *approximate* the optimal ISD $g^*(\mathbf{x})$ or to at least identify one providing small standard deviation of the MC estimator: see, e.g., Refs. [90,91,94–99] for details. The estimator $\hat{g}^*(\mathbf{x})$ is, here, constructed in two stages:

a. we employ the Subset Simulation (SS) method [67,68,70,73] to create a population $\{\mathbf{z}_F^m : m = 1, 2, ..., M\}$ of M samples *approximately* distributed according to the optimal ISD $g^*(\cdot)$ (4), i.e., $\{\mathbf{z}_F^m : m = 1, 2, ..., M\} \sim g^*(\cdot) = I_F(\cdot)q(\cdot)/P(F)$.[2] Notice that here we substitute the expensive T-H code $\mathbf{y}(\mathbf{x}) = \boldsymbol{\mu}_\mathbf{y}(\mathbf{x})$ with a quick-running ANN regression model (*adaptively* refined by means of the samples *iteratively* produced by SS), with the aim of reducing the computational burden related to this step (Section 3.1.1);
b. we "fit" the samples $\{\mathbf{z}_F^m : m = 1, 2, ..., M\}$ by means of a proper PDF, in order to obtain an estimator $\hat{g}^*(\mathbf{x})$ for $g^*(\mathbf{x})$. To this aim, we adopt the *fully nonparametric* PDF proposed in Ref. [99] and relying on the Gibbs Sampler [101] (Section 3.1.2).

3.1.1. Subset Simulation and adaptively trained ANN metamodels for producing samples distributed according to the optimal ISD

As highlighted above, in order to build an estimator $\hat{g}^*(\cdot)$ for $g^*(\cdot) = I_F(\cdot)q(\cdot)/P(F)$ we need a population $\{\mathbf{z}_F^m : m = 1, 2, ..., M\}$ of M samples *approximately* distributed according to the optimal ISD, i.e., $\{\mathbf{z}_F^m : m = 1, 2, ..., M\} \sim g^*(\cdot)$. The Subset Simulation (SS) method is here adopted to this aim. The strength of SS is that it converts the simulation of a single event (in this case, the failure event F) into a sequence of n_F simulations of intermediate conditional events corresponding to subsets F_i, $i = 1, 2, ..., n_F$, of the uncertain parameter space. During simulation, Markov Chain Monte Carlo (MCMC) based on the Metropolis–Hastings (MH) algorithm [67] is employed to produce the conditional samples (belonging to the intermediate subregions F_i): in this way, the successive intermediate subsets F_i are progressively populated by the conditional samples up to the target (failure) region F [67,68]. It can be shown that following this procedure, the stationary density of the (failure) points lying in F is $q(\cdot|F) = I_F(\cdot)q(\cdot)/P(F)$, which is exactly the ideal, zero-variance ISD $g^*(\cdot)$ [67,73]. On the other hand, it is worth noting that a single run of the SS method may require a significant number N_c of evaluations of the long-running model, depending on the magnitude of $P(F)$ (i.e., approximately $N_c = M \cdot \log_{p0}[P(F)]$, with $M \geq 100$ for producing reliable estimates, which may imply even thousands of calls to the T-H code). Thus, in order to reduce the associated computational burden, we substitute the T-H code $\mathbf{y}(\mathbf{x}) = \boldsymbol{\mu}_\mathbf{y}(\mathbf{x})$ by a regression model $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{w}^*)$, depending on a vector of adjustable parameters $\boldsymbol{w}^*$ to be properly calibrated by means of a *finite* (and possibly *reduced*) set of input/output data examples (i.e., patterns), also called Design Of Experiments (DOE) [114]. In this work, three-layered feed-forward Artificial Neural Networks (ANNs) trained by the error back-propagation algorithm are considered [52,53,115–117]. They are structured in three *layers* (namely, the input, hidden and output layers), which contain respectively n_i, n_h and n_o computing elements (named *neurons* or *nodes* and mathematically described by sigmoidal basis functions). These computing units are linked by weighed connections (called *synapses*). Details about ANN regression models are not reported here for brevity: the interested reader may refer to the cited references and the copious literature in the field.

The core of the method (proposed in Ref. [100] and adopted in the present manuscript) is that the ANN metamodel is built and *adaptively* refined by means of the samples *iteratively* produced by SS itself, according to the following steps [105,119] (Fig. 1):

1. Construct and train an initial ANN regression model $\boldsymbol{f}(\boldsymbol{z}, \boldsymbol{w}^0)$ by means of a DOE $D_{TR}^0 = \{(\mathbf{z}_p^0, \mathbf{y}_p^0),\ p = 1, 2, ..., N_{TR}^0\}$ constituted by N_{TR}° elements selected by plain random sampling, Latin Hypercube Sampling (LHS), partial/full factorial designs or other more sophisticated experimental design methods. By so doing, we create a "first trial" surrogate of the T-H

[2] Notice that the change of name of the 'dummy input vector' from $\boldsymbol{x}$ to $\boldsymbol{z}$ has no 'conceptual reason' and is done only for notational simplicity and coherence with the description of the method reported in the following Sections.

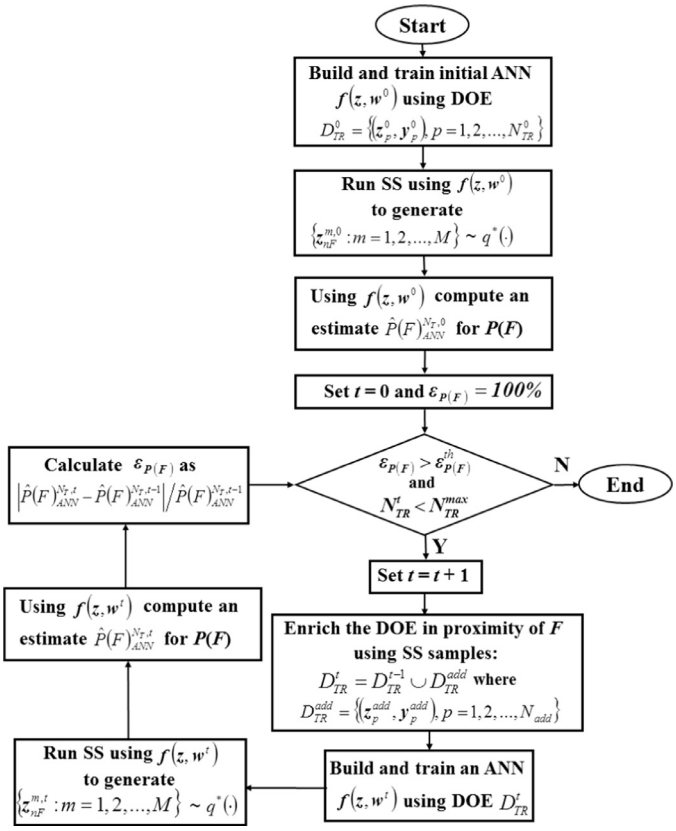

Fig. 1. Sketch of the procedure used for generating samples distributed as the ideal, zero-variance ISD by means of Subset Simulation and adaptively trained ANN regression models within the AM-SIS algorithm (Section 3.1.1).

model code represented by the unknown nonlinear deterministic function $\boldsymbol{y} = \boldsymbol{\mu_y}(\boldsymbol{z})$. Notice that the ANN structure (in practice, the number $n_h{}^0$ of hidden neurons) has to be chosen so that the total number $n_{p,ANN}^0$ of adjustable parameters $\boldsymbol{w}^0$ is lower than the size of the DOE (i.e., $n_{p,ANN}^0 < N_{TR}{}^0$);

2. Run SS employing the ANN regression model $\boldsymbol{f(z, w^0)}$ (instead of the original T-H code) to generate M samples $\{\boldsymbol{z}_{nF}^{m,0} : m = 1, 2, ..., M\}$ approximately distributed as $g^*(\cdot) = q(\cdot \mid F) = I_F(\cdot)q(\cdot)/P(F)$. Notice that this simulation is almost costless from the computational viewpoint, since it requires calling only the fast-running ANN model $\boldsymbol{f(z, w^0)}$ instead of the long-running T-H code $\boldsymbol{\mu_y}(\boldsymbol{z})$: thus, the number M of conditional samples iteratively produced by the SS algorithm in each intermediate conditional region F_i can be chosen to be quite large, in order to increase the accuracy and robustness of the procedure (e.g., $M = 50{,}000$ in this paper);
3. Produce an estimate $\hat{P}(F)_{ANN}^{N_T,0}$ of the failure probability $P(F)$ by using $\boldsymbol{f(z, w^0)}$ (the estimate provided by SS at the previous step 2 can be straightforwardly used). As for step 2 above, this operation is almost computationally costless;
4. Set $t = 0$ and $\varepsilon_{P(F)} = 100\%$, where t is the iteration index and $\varepsilon_{P(F)}$ stands for the relative variation between two consecutive ANN-based estimates of $P(F)$ produced during the adaptive and iterative procedure of metamodel refinement;
5. while $\varepsilon_{P(F)} > \varepsilon_{P(F)}^{th}$ (i.e., the change between the ANN-based failure probability estimates at two consecutive iterations is larger than a proper threshold $\varepsilon_{P(F)}^{th}$) *and* $N_{TR}{}^t < N_{TR}{}^{max}$ (i.e., the size of the DOE – in other words, the overall number of code runs – remains smaller than a maximal predefined value $N_{TR}{}^{max}$), *repeat* the following operations in order to *refine* the ANN regression model in proximity of the failure region F and to produce a sample approximately distributed as the "ideal" ISD $g^*(\cdot) = I_F(\cdot)q(\cdot)/P(F)$:
   (a) $t = t + 1$;
   (b) select N_{add} points $\{\boldsymbol{z}_p^{add}: p = 1, 2, ..., N_{add}\}$ among all those produced by SS at iteration $(t - 1)$: a high fraction of these points is selected "around" the limit state in order to better describe the response of the T-H passive system in the areas close to the failure region F; the remaining elements are "distributed" in proximity of the boundaries of the other intermediate conditional subsets F_i identified by SS. By so doing, a reliable "anchoring" of the ANN metamodel is possible also in the relevant areas of the uncertain input space "visited" by the SS algorithm (in order not to interrupt the flow of the presentation, further technical details about this step are given at the end of this Section);

(c) join the N_{add} new elements to the current DOE after N_{add} new evaluations of the system performance function by means of the original T-H model $\boldsymbol{\mu}_{\mathbf{y}}(\mathbf{z})$: thus, $D_{TR}^{t} = D_{TR}^{t-1} \cup D_{TR}^{add}$, where $D_{TR}^{add} = \{(\mathbf{z}_p^{add}, \mathbf{y}_p^{add} = \mu_{\mathbf{y}}(\mathbf{z}_p^{add})),\ p = 1, 2, ..., N_{add}\}$; also, the size N_{TR}^{t} of D_{TR}^{t} is $N_{TR}^{t} = N_{TR}^{t-1} + N_{add}$. It is important to note that now the current DOE is obviously enriched in the areas close to the failure region;

(d) employ the updated DOE D_{TR}^{t} to construct an updated ANN regression model $\mathbf{f}(\mathbf{x}, \mathbf{w}^{t})$. Since the size of the DOE has increased, also the number of the ANN adjustable parameters (in practice, the number of hidden neurons) can be increased until $n_{p,ANN}^{t} < N_{TR}^{t}$. Also, notice that the new ANN metamodel $\mathbf{f}(\mathbf{x}, \mathbf{w}^{t})$ is now expected to be more accurate in describing the behavior of the original T-H code around the region F of interest;

(e) generate M samples $\{\mathbf{z}_{nF}^{m,t} : m = 1, 2, ..., M\}$ approximately distributed as $g^{*}(\cdot)$ by running SS with the ANN regression model $\mathbf{f}(\mathbf{x}, \mathbf{w}^{t})$;

(f) produce an ANN-based estimate $\hat{P}(F)_{ANN}^{N_T,t}$ of P(F) and compute $\varepsilon_{P(F)}$ as $|\hat{P}(F)_{ANN}^{N_T,t} - \hat{P}(F)_{ANN}^{N_T,t-1}| / \hat{P}(F)_{ANN}^{N_T,t-1}$. Notice that SS has to be run at each iteration in order to obtain both the samples $\{\mathbf{z}_{nF}^{m,t} : m = 1, 2, ..., M\}$ and the failure probability estimates $\hat{P}(F)_{ANN}^{N_T,t}$ used for the convergence check. In order to "remove" the variability that stems from different random numbers (in subsequent simulations), the random generator is reinitialized using the same seed at each iteration;

6. the failure region F does not change excessively from one iteration to another, i.e., it is reliably "located", and the ANN is sufficiently accurate in its proximity [119]. Thus, the vectors $\{\mathbf{z}_{nF}^{m,t} : m = 1, 2, ..., M\}$ produced at the last iteration t are retained as if they were the "optimal ones" $\{\mathbf{z}_{F}^{m} : m = 1, 2, ..., M\}$, distributed as the *ideal, zero-variance* ISD $g^{*}(\cdot) = I_F(\cdot)q(\cdot)/P(F)$.

At the end of the procedure, the total number $N_{c,aux}$ of T-H code runs carried out to refine the metamodel is $N_{c,aux} = N_{TR}{}^{t} = N_{TR}{}^{0} + t \cdot N_{add}$.

Some considerations are in order with respect to the algorithm above. As already said, at step 4.b N_{add} points are selected (among *all* those generated by SS) to be added to the current DOE D_{TR}^{t-1}. A relevant fraction K of these N_{add} points is chosen "across" the limit state, in proximity of the failure region $F_{nF} \equiv F$. This is practically implemented as follows: (i) consider all the SS samples belonging to the failure region $F_{nF} \equiv F = \{\mathbf{z}: Y(\mathbf{z}) > \alpha_Y\}$ and to the preceding intermediate conditional region $F_{nF-1} = \{\mathbf{z}: Y(\mathbf{z}) > y_{nF\text{-}1}\}$ as a 'pool' of possible candidates to be included into the DOE (notice that these SS samples 'cover' the limit state); (ii) identify the candidate for which the *minimal* distance from the elements of the current DOE is the *largest* with respect to the other candidates of the 'pool', and add it to the current DOE (this allows filling the "empty" areas of the input space); (iii) remove the selected vector from the 'pool' of candidates; (vi) repeat steps (ii)–(iii) until $K \cdot N_{add}$ new elements are included into the DOE.

By means of the same procedure, the remaining $(1 - K) \cdot N_{add}$ vectors are equally "distributed" in proximity of the boundaries of the *other* SS intermediate conditional regions: by way of example, the set of candidates considered for refining the ANN in proximity of region $F_i = \{\mathbf{z}: Y(\mathbf{z}) > y_i\}$ is obviously represented by all the SS samples lying *across* threshold y_i, i.e., $\{\mathbf{z}: Y(\mathbf{z}) > y_{i\text{-}1} \cap Y(\mathbf{z}) < y_{i+1}\}$. This way of proceeding ensures an acceptable quality of the ANN metamodel *also* in other "key areas" of the uncertain input space *besides* the failure region F itself: actually, in order to produce a set of points $\{\mathbf{z}_{F}^{m} : m = 1, 2, ..., M\}$ that are distributed as $g^{*}(\cdot) = I_F(\cdot)q(\cdot)/P(F)$, the SS algorithm needs to *gradually* and *accurately* visit the entire uncertain input space and its intermediate conditional regions. Differently from Ref. [100] (where K was set to 0.5 by trial-and-error and the remaining $(1 - K)N_{add}$ points were randomly distributed across the uncertain input space), in this paper the fraction K mentioned above is computed as $2 \bullet N_{add}/n_F$: in other words, the number of points used to refine the metamodel around the limit state $\{\mathbf{z}: Y(\mathbf{z}) = \alpha_Y\}$ is twice the number of those used to improve the ANN in proximity of the other intermediate SS thresholds. This choice is motivated as follows: (i) higher importance is given to the failure region with respect to other areas of the uncertain input space; (ii) the value of K changes depending on the number n_F of intermediate regions, i.e., according to the magnitude of the failure probability $P(F)$; (iii) this value is empirically found to perform satisfactorily on the case study analysed.

Only for illustration purposes, Fig. 2 pictorially shows the key concepts underlying the procedure for enriching the DOE. The rectangles represent the elements of the current DOE in regions $F_{nF} \equiv F$ and F_{nF-1}, whereas the circles represent the 'pool' of possible candidates generated by SS. We suppose that three among these candidates have to be selected and added to the DOE. At each step, we identify the candidate whose *minimal* distance from the elements of the current DOE is the *largest* with respect to the other candidates of the 'pool' (see the arrows in Fig. 2). This candidate is added to the DOE, which is thus *dynamically updated*. The procedure is repeated (illustrative steps a–d in Fig. 2) until the desired number of points (i.e., three in the case of Fig. 2) is included.

Also, notice that steps 2 and 4.f of the algorithm above require the computation of an ANN-based estimate $\hat{P}(F)_{ANN}^{N_T}$ of $P(F)$. However, approximating the original system model output by an ANN empirical regression model (trained by means of a *finite* set of data) bears two effects: (i) the corresponding estimates *may* be subject to a *bias* (which makes them *not accurate*); (ii) the associated estimates *are* subject to additional *uncertainty* (which makes them *less precise*). To address issue (i), we employ a Bootstrap-based Bias Correction (BBC) procedure relying on an ensemble of B (e.g., $B = 100$) ANN regression models constructed on B 'fictitious' data sets randomly sampled with replacement from the original one: see Refs. [52,53,120] for details. On the other hand, we do not tackle issue (ii). Actually, contrary to other works of literature involving the combination of metamodeling and advanced MCS, e.g., Refs. [103–111], in this paper we do *not* use the metamodel to

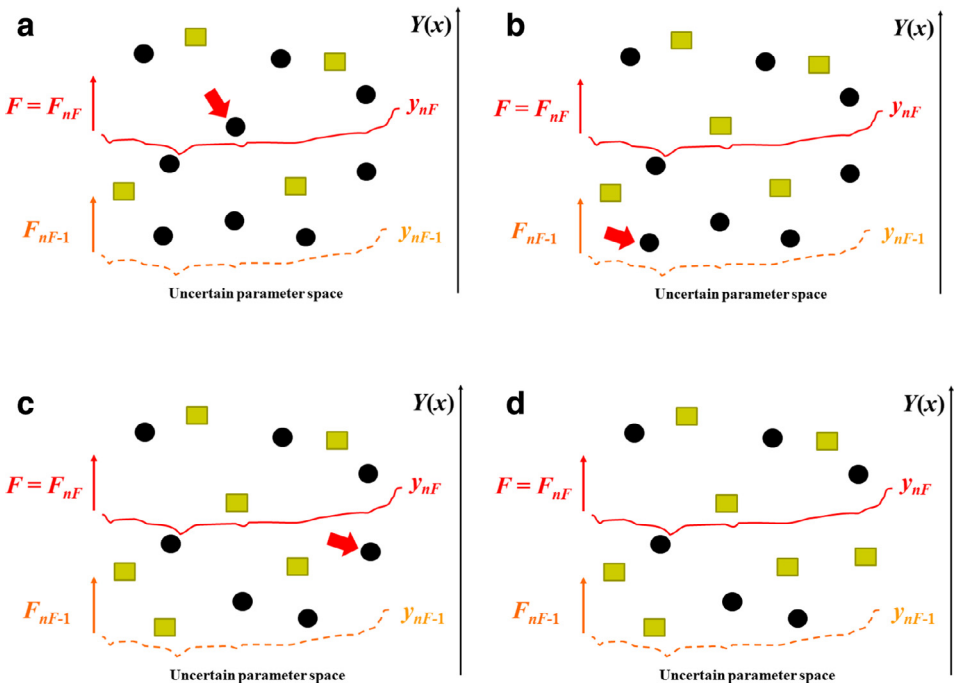

Fig. 2. Conceptual steps underlying the procedure for enriching the DOE in proximity of the failure region F (step 4.b of the iterative algorithm of ANN refinement). Rectangles: current DOE; circles: 'pool' of candidates generated by SS; arrows: candidate added to the DOE. a) identification of the first candidate to be added; b) inclusion of the first candidate into the DOE and selection of the second; c) inclusion of the second candidate into the DOE and selection of the third; and d) inclusion of the third candidate into the DOE.

provide the *final* estimate $\hat{P}(F)^{N_T}$ of the failure probability: instead, we employ it *only* as a "support tool" for constructing the quasi-optimal ISD (in facts, in the IS phase of the present method the *original* system model is used, see the following Section 3.2). In other words, the *only* objective of the ANN-based iterative algorithm described above is that of *accurately* "locating" the failure region F of interest *without any bias*: in this view, we perform *only* the amount of system model evaluations that are *strictly necessary* to achieve this objective, whereas *no* further code runs are devoted to increase the *precision* (or, conversely, to reduce the *uncertainty*) of the metamodel estimates.

In addition, discussion is warranted on the convergence criterion adopted (step 4.f above). Actually, (i) the ultimate goal of this first stage of AM-SIS is to draw samples that are a satisfactory representation of the quasi-optimal ISD and (ii) the ANN trained here is not used in the following failure probability estimation step (Section 3.2). This may lead to conclude that the failure probability estimated through the iterations of SS and ANN (steps 3 and 4.f) is not a "proper" measure of convergence for the algorithm of Fig. 1. On the other hand, it is important to notice that the *accuracy* and *unbiasedness* of the ANN discussed above (i.e., its capability of correctly "*classifying*" and "*distinguishing*" safe and failure configurations) is absolutely fundamental for obtaining samples distributed as $g^*(\cdot)=I_F(\cdot)q(\cdot)/P(F)$ by means of SS. In this respect, also note that the correct identification of failure/safe points *combined* with the mechanism of MCMC (which can generate random samples distributed according to *any* PDF) should be enough to guarantee that the system configurations produced at the end of step 6 of the algorithm are representative of the optimal ISD $g^*(\cdot)$.

Obviously, these considerations do not imply that the convergence criterion selected is the only possible (or the best overall). For example, one could also look at the difference (say, through relative *entropy*) [121] between the established ISDs at subsequent iterations. On the other hand, convergence of an entire probability distribution over the whole uncertain parameter space may be much more demanding (in terms of required code runs) than a simple classification of failure/safe configurations.

Finally, a consideration is in order with respect to the impact of the *dimensionality* of the problem (i.e., of the dimension of $\boldsymbol{x}$) on the ANN quality and, thus, on the density approximation. Increasing the dimension of the uncertain variable space actually brings rather challenging issues, including the so-called *curse of dimensionality*. In practice, we are trying to train a metamodel on a learning dataset whose size is not increased sufficiently with the dimension of the space, giving rise to the so-called *empty space phenomenon* [105]. This issue seems unavoidable and limits the application of any algorithm belonging to the metamodel family to engineering problems with a small number of input parameters (say tens), unless some *dimensionality reduction* strategy is adopted: e.g., principal component analysis, feature selection or extraction and sensitivity analysis methods [122–124].

3.1.2. Constructing an estimator of the optimal ISD by means of a fully nonparametric Probability Density Function (PDF)

The iterative ANN-based algorithm presented above produces a population of samples approximately distributed as the ideal, zero-variance ISD: however, these samples *cannot* be used *directly* to define an unbiased IS estimator, because the corresponding sampling density $g^*(\cdot)$ is *not* known *explicitly*. Moreover, the sampled states $\{\mathbf{z}_F^m : m = 1, 2, ..., M\}$ are not independent. A first (naive) idea might be to just use the empirical distribution of $\{\mathbf{z}_F^m : m = 1, 2, ..., M\}$ for IS sampling. But this distribution does not satisfy the requirement of having a nonzero density wherever $I_F(\mathbf{x})q(\mathbf{x}) > 0$. To satisfy this requirement, a second idea is to fit a kernel density estimate to this sample, based on a kernel whose density is nonzero wherever $I_F(\mathbf{x})q(\mathbf{x}) > 0$ [93,94]. However, kernel density estimation by standard methods is difficult in high dimensions. Finally, in Ref. [99] the authors propose to adopt the one step ahead transition density $\kappa(\mathbf{x}|\mathbf{z})$ of a Markov chain, when the chain is in state $\mathbf{z}_F^m$, and average over the M states $\{\mathbf{z}_F^m : m = 1, 2, ..., M\}$. In this way, the optimal ISD $g^*(\mathbf{x})$ can be approximated by $\hat{g}^*(\mathbf{x})$ (2):

$$\hat{g}^*(\mathbf{x}) = \frac{1}{M}\sum_{m=1}^{M}\kappa(\mathbf{x}|\mathbf{z}_F^m). \tag{2}$$

Following the suggestion of Ref. [99], in this paper we use the transition density $\kappa(\mathbf{x}|\mathbf{z}_F^m)$ adopted by the Gibbs Sampler [101] (see the Appendix for thorough technical details).

The advantage of the ISD (2) is that it is not limited by parametric models (e.g., normal, exponential, lognormal, etc.), as it happens instead for the Cross Entropy (CE) [95] or Variance Minimization (VM) [125] techniques. On the other hand, for ISD (2) the positivity condition (i.e., that $\hat{g}^*(\mathbf{x}) > 0$ whenever $I_F(\mathbf{x})q(\mathbf{x}) > 0$) is not guaranteed in the rare-event setting. Thus, $\hat{g}^*(\mathbf{x})$ is here modified to assure that it is nonzero over the entire support of $g^*(\mathbf{x})$. To this aim, we consider a "weighed average" of $\hat{g}^*(\mathbf{x})$ with the original density $q(\boldsymbol{x})$, i.e.:

$$\hat{g}_w^*(\mathbf{x}) = w \cdot q(\mathbf{x}) + (1 - w) \cdot \hat{g}^*(\mathbf{x}), \tag{3}$$

where w is a weight arbitrarily selected between 0 and 1 (in this work, $w = 0.01$).

3.2. Importance sampling from the estimator of the optimal ISD

The well-known 'composition method' is employed to draw N_T samples from $\hat{g}_w^*(\mathbf{x})$ (3). In extreme synthesis, for $k = 1, 2, ..., N_T$, a uniform random number u is generated in [0, 1): if $u < w$, then the vector $\boldsymbol{x}^k$ is sampled from the original PDF $q(\boldsymbol{x})$; otherwise, from $\hat{g}^*(\mathbf{x}) = \frac{1}{M}\sum_{m=1}^{M}\kappa(\mathbf{x}|\mathbf{z}_F^m)$ (2) (for the sake of brevity, we do not report here details about the steps necessary to draw samples from the mixture $\hat{g}^*(\mathbf{x})$ (2) of transition densities: the interested reader is referred to the Appendix).

Since the failure probability $P(F) = E_q[I_F(\boldsymbol{x})q(\boldsymbol{x})]$ (1) can be obviously expressed also as $E_{\hat{g}_w^*}[\frac{I_F(\mathbf{x})q(\mathbf{x})}{\hat{g}_w^*(\mathbf{x})}]$, the AM-SIS estimator $\hat{P}(F)^{N_T}$ of the failure probability $P(F)$ is then [90–99]:

$$\hat{P}(F)^{N_T} = \frac{1}{N_T}\sum_{k=1}^{N_T}\frac{I_F(\mathbf{x}^k)q(\mathbf{x}^k)}{\hat{g}_w^*(\mathbf{x}^k)} = \frac{1}{N_T}\sum_{k=1}^{N_T}\frac{I_F(\mathbf{x}^k)q(\mathbf{x}^k)}{w \cdot q(\mathbf{x}^k) + (1 - w) \cdot \hat{g}^*(\mathbf{x}^k)}. \tag{4}$$

The sample standard deviation $\hat{\sigma}[\hat{P}(F)^{N_T}]$ of (4) can be also straightforwardly computed since the realizations $I_F(\mathbf{x}^k)q(\mathbf{x}^k)/\hat{g}_w^*(\mathbf{x}^k)$, $k = 1, 2, ..., N_T$, are *independent*. With respect to that, it is also worth noting that the estimators $\hat{P}(F)^{N_T}$ and $\hat{\sigma}[\hat{P}(F)^{N_T}]$ are *unbiased* for the mean and the standard deviation, even if (i) the samples $\{\mathbf{z}_F^m : m = 1, 2, ..., M\}$ iteratively generated by SS are by construction *not independent* and are *not exactly* distributed as $g^*(\mathbf{x})$ (see Section 3.1.1) and (ii) the quality of the ISD approximation in the first step of the algorithm is low (see Section 3.1.2). On the other hand, the *performance* of the AM-SIS algorithm obviously depends on the quality of this approximation.

Notice again that the *original* system model (*not* the ANN metamodel employed at the previous step) is used in this IS-based failure probability estimation phase. Actually, as explained in Section 3.1.1, one of the effects of the approximation of the original system model output by an empirical regression model (trained by means of a *finite* set of data) is that the associated estimates *are* subject to additional *uncertainty* (which makes them *less precise*). Employing instead the ANN *only* as a "support tool" for constructing the quasi-optimal ISD (i.e., for *accurately* "locating" the failure region F) allows performing *only* the amount of system model evaluations that are *strictly necessary* to achieve this objective: thus, *no* additional code runs are devoted to increase the *precision* (or, conversely, to reduce the *uncertainty*) of the ANN metamodel estimates, with obvious benefits from the point of view of the reduced computational cost.

4. Case study: functional failure analysis of a T-H passive system of literature

As in the previous paper by the authors [100], the case study involves a Gas-cooled Fast Reactor (GFR) cooled by natural convection after a Loss Of Coolant Accident (LOCA) [26]. The reactor is a 600-MW GFR cooled by helium flowing through separate channels in a silicon carbide matrix core [126].

In case of a LOCA, long-term heat removal is ensured by natural circulation in a given number N_{loops} of identical and parallel heat removal loops. However, in order to achieve a sufficient heat removal rate by natural circulation, it is necessary

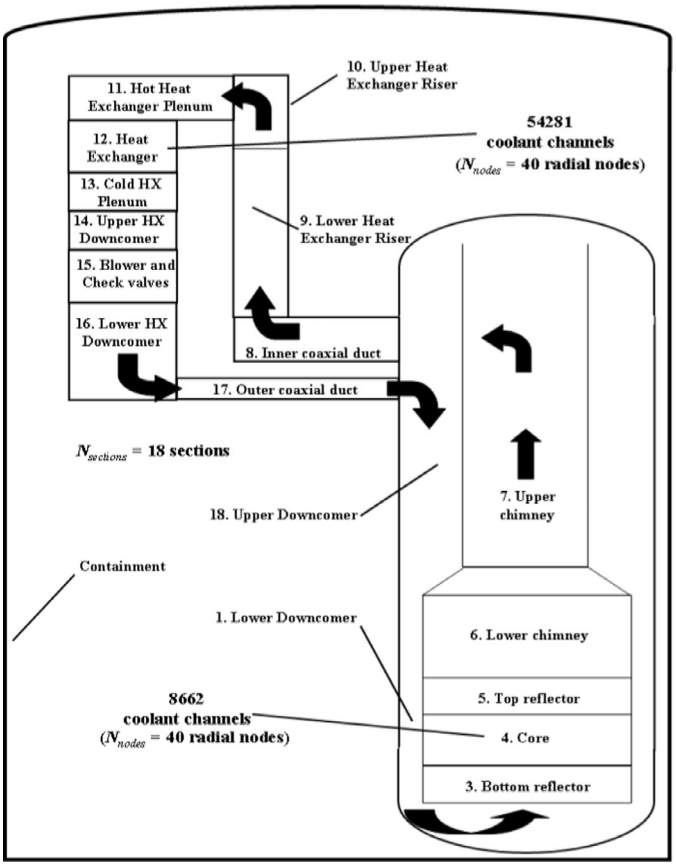

Fig. 3. One loop of passive decay heat removal system of the 600-MW GFR [26].

to maintain an elevated pressure even after the LOCA. This is accomplished by a guard containment, which surrounds the reactor vessel and power conversion unit, and holds the pressure at a level that is reached after the depressurization of the system [26].

The configuration of the passive system for the removal of decay heat is shown schematically in Fig. 3, where only one loop out of N_{loops} is reported for clarity sake: the flow path of the cooling helium gas is suggested by the black arrows. As shown in Fig. 3, the loop has been divided into $N_{sections} = 18$ sections; technical details about the geometrical and structural properties of these sections are not reported here for brevity: the interested reader may refer to [26].

It is worth noting that as in Ref. [26] the subject of the present analysis is the quasi-steady-state natural circulation cooling that takes place after the LOCA transient has occurred. Thus, the analyses hereafter reported refer to this steady-state period and are conditional to the successful inception of natural circulation. As a consequence, the analyses do not consider the probability of not starting natural convection nor the probability of not building up and maintaining a high pressure level in the guard containment.

4.1. The deterministic T-H model

To simulate the steady-state behavior of the system, a one-dimensional thermal-hydraulic MATLAB code has been implemented. The code treats all the N_{loops} multiple loops as identical and each loop is divided in $N_{sections} = 18$ sections (Fig. 3). Further, the sections corresponding to the heater (i.e., the reactor core, item 4 in Fig. 3) and the cooler (i.e., the heat exchanger, item 12 in Fig. 3) are divided in a proper number N_{nodes} of axial nodes to compute the temperature and flow gradients with sufficient detail (40 nodes are chosen for the present analysis). Both the average and hot channels are modeled in the core so that the increase in temperature in the hot channel due to the radial peaking factor can be calculated [26].

From a strictly mathematical point of view, obtaining a steady-state solution amounts to dropping the time dependent terms in the energy and momentum conservation equations [127]; in practice, the code balances the pressure losses around the loops so that friction and form losses are compensated by the buoyancy term, while at the same time maintaining the heat balance in the heater (i.e., the reactor core, item 4 in Fig. 3) and cooler (i.e., the heat exchanger, item 12 in Fig. 3).

Table 1
Equations governing the heat transfer process in each node l = 1, 2, ..., N_{nodes} of both the heater and the cooler of the 600-MW GFR passive decay heat removal system of Fig. 3, together with a description of the physical parameters and their units of measure.

Heat transfer equations

$$\dot{Q}_l = \dot{m} c_{p,l} (T_{out,l} - T_{in,l}), \quad l = 1, 2, \ldots, N_{nodes} \qquad (5)$$

$$\dot{m} c_{p,l} (T_{out,l} - T_{in,l}) = S_l h_l (T_{wall,l} - T_{bulk,l}), \quad l = 1, 2, \ldots, N_{nodes} \qquad (6)$$

Parameter	Description	Unit of measure
$\dot{Q}_l$	Heat flux in the lth node	kW
$\dot{m}_l$	Mass flow rate in the lth node	kg/s
$c_{p,l}$	Specific heat at constant pressure in the lth node	kJ/kg K
$T_{out,l}$	Temperature measured at the outlet of the lth node	K
$T_{in,l}$	Temperature measured at the inlet of the lth node	K
$T_{wall,l}$	Temperature measured at the wall channel of the lth node	K
$T_{bulk,l}$	Temperature measured at the bulk of the lth node	K
S_l	Heat-exchanging surface in the lth node	m^2
h_l	Heat transfer coefficient in the lth node	kW/m^2 K

Table 2
Equation stating the balance between buoyancy and pressure losses along the closed loop of the 600-MW GFR passive decay heat removal system of Fig. 3, together with a description of the physical parameters and their units of measure.

Mass flow rate equation

$$\sum_{s=1}^{N_{sections}} \left[\rho_s g H_s + f_s \frac{L_s}{D_s} \frac{\dot{m}^2}{2\rho_s A_s^2} + K_s \frac{\dot{m}^2}{2\rho_s A_s^2} \right] = 0 \qquad (7)$$

Parameter	Description	Unit of measure
ρ_s	Coolant density in the sth section of the loop	kg/m^3
H_s	Height of the sth section of the loop	m
f_s	Friction factor in the sth section of the loop	/
L_s	Length of the sth section of the loop	m
D_s	Hydraulic diameter of the sth section of the loop	m
$\dot{m}$	Mass flow rate in the sth section of the loop	kg/s
A_s	Flow area of the sth section of the loop	m^2
K_s	Form loss coefficient of the sth section of the loop	–

Eqs. (5) and (6) in Table 1 govern the heat transfer process in each node $l = 1, 2, \ldots, N_{nodes}$ of both the heater and the cooler.

Eq. (5) in Table 1 states the equality between the enthalpy increase between the flow at the inlet and the flow at the outlet in any node, whereas Eq. (6) in Table 1 regulates the heat exchange between the channel wall and the bulk of the coolant.

The mass flow rate is determined by a balance between buoyancy and pressure losses along the closed loop according to Eq. (7) in Table 2.

Eq. (7) in Table 2 states that the sum of buoyancy (first term), friction losses (second term) and form losses (third term) should be equal to zero along the closed loop [26].

Notice that the heat transfer coefficients h_l, $l = 1, 2, \ldots, N_{nodes}$, in (6) in Table 1 are functions of the fluid characteristics and the geometry, and are calculated through proper correlations covering forced-, mixed- and free-convection regimes, in both turbulent and laminar flow; further, different Nusselt number correlations are used in the different regimes to obtain values for the heat transfer coefficients. Also the friction factors f_s, $s = 1, 2, \ldots, N_{sections}$, in (7) in Table 2 are functions of the fluid characteristics and geometry and are calculated using appropriate correlations [26]. An iterative algorithm is used to find a solution that satisfies simultaneously the heat balance and pressure loss equations.

4.2. Uncertainties in the deterministic T-H model

There is incomplete knowledge on the properties of the system and the conditions in which the passive phenomena develop. This kind of uncertainty, often termed epistemic, affects the model representation of the passive system behavior [9,29,128].

The nine model parameters $\boldsymbol{x} = \{x_j: j = 1, 2, ..., 9\}$ affected by epistemic uncertainty are summarized in Table 3 [26]: the mean and standard deviations of the corresponding Gaussian PDFs are also reported.

The sequential steps undertaken to propagate the epistemic (i.e., model and parameter) uncertainties through the deterministic T-H equations described in the previous Section 4.1 are:

i Draw samples of the uncertain input *parameters* x_1 (i.e., power), x_2 (i.e., pressure) and x_3 (i.e., cooler wall temperature) from the respective epistemic probability distributions (Table 3).

Table 3
Epistemic uncertainties considered for the 600-MW GFR passive decay heat removal system of Fig. 3 [26].

	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%

ii Using the values sampled for input parameters x_1, x_2 and x_3 (step i. above), compute Nusselt numbers and friction factors in forced-, mixed- and free-convection regimes by means of deterministic correlations. In this way, the *deterministically computed* values of Nusselt numbers and friction factors are dependent on the values of the *uncertain* input parameters x_1, x_2 and x_3.

iii Draw samples of the multiplicative error factors x_4, x_5, …, x_9, for Nusselt numbers and friction factors in forced-, mixed- and free-convection regimes, from the respective epistemic probability distributions (Table 3).

iv Multiply the deterministic values of Nusselt numbers and friction factors (computed at step ii. above) by the corresponding error factors (sampled at step iii. above). In this way, the model uncertainties (i.e., the multiplicative error factors) are treated *independently* from the parameter uncertainties; yet, notice that the model uncertainties are applied to the results of deterministic correlations, which in turn depend on the uncertainties in the input parameters (i.e., power, pressure and cooler wall temperature).

v Using an iterative algorithm, find a solution that satisfies simultaneously the heat balance (Table 1 in Section 4.1) and the pressure losses equations (Table 2 in Section 4.1).

4.3. Failure criteria of the T-H passive system

The passive decay heat removal system of Fig. 3 is considered failed whenever the temperature of the coolant helium leaving the core (item 4 in Fig. 3) 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 Fig. 3) and in the stainless steel cross ducts connecting the reactor vessel and the cooler (items from 6 to 11 in Fig. 3) [26].

Indicating by $\mathbf{x}$ the vector of the 9 uncertain system parameters of Table 3 (Section 4.2) and by $T^{hot}_{out,core}(\mathbf{x})$ and $T^{avg}_{out,core}(\mathbf{x})$ the coolant outlet temperatures in the hot and average channels, respectively, the failure event F can be written as follows:

$$F = \left\{\mathbf{x}:\ T^{hot}_{out,core}(\mathbf{x}) > 1200\right\} \cup \left\{\mathbf{x}:\ T^{avg}_{out,core}(\mathbf{x}) > 850\right\}. \tag{8}$$

Notice that, in the notation of the preceding Section 2, $T^{hot}_{out,core}(\mathbf{x}) = y_1(\boldsymbol{x})$ and $T^{avg}_{out,core}(\mathbf{x}) = y_2(\boldsymbol{x})$ are the two target outputs of the T-H model.

The failure event F in (8) can be condensed into a single performance indicator $Y(\boldsymbol{x})$ (Section 2) as:

$$Y(\mathbf{x}) = max\left\{\frac{T^{hot}_{out,core}(\mathbf{x})}{1200}, \frac{T^{avg}_{out,core}(\mathbf{x})}{850}\right\} = max\left\{\frac{y_1(\mathbf{x})}{1200}, \frac{y_2(\mathbf{x})}{850}\right\}, \tag{9}$$

so that the failure event F becomes specified as:

$$F = \{\mathbf{x}:\ Y(\mathbf{x}) > 1\}. \tag{10}$$

In the notation of Section 2, the failure threshold α_Y is then equal to one. The locus of points $\boldsymbol{x}$ for which the performance indicator $Y(\boldsymbol{x})$ is *equal* to the failure threshold α_Y, i.e., $\{\mathbf{x}:\ Y(\mathbf{x})=\alpha_Y\}$, is referred to as *limit state*.

Notice that this case study has been already used by the authors in previous works [52,53,72,81,82]. However, in this paper the corresponding thermal-hydraulic MATLAB code has been modified in order to include $N_{loop} = 6$ decay heat removal loops in the safety system: in this configuration, the true value of the functional failure probability $P(F)$ is $1.8089 \cdot 10^{-7}$.

5. Application of the AM-SIS algorithm for the functional failure analysis of the T-H passive system of Section 4

The objective of the application is the assessment of the *very small* functional failure probability $P(F)$ (i.e., $P(F) = 1.8089 \cdot 10^{-7}$) of the T-H passive system of Section 4, by means of the AM-SIS method. The *accuracy* and *precision* of the proposed technique are compared to those of several simulation methods of literature. Two sets of comparisons are carried out. In Section 5.1, the focus is *mainly* on the efficiency of stage 2 of the algorithm (Section 3.2), i.e., of random sampling for failure probability estimation; in Section 5.2, comparisons are made to other methods combining Subset Simulation (SS) and metamodels that are instead very close to stage 1. of the proposed approach (Section 3.1).

Table 4
Parameters of the AM-SIS algorithm.

Parameter	Value	Description	Phase in AM-SIS algorithm
N_{TR}^{0}	$3n_i = 27$	Initial DOE size	Initial ANN training
$N_{TR}^{\max}$	4000	Upper limit for DOE size	ANN training and refinement
N_{add}	$3n_i = 27$	Number of samples used for DOE enrichment at each iteration	ANN training and refinement
$\varepsilon_{P(F)}^{th}$	0.01	Threshold for checking convergence of the ANN-based failure probability estimates	ANN training and refinement
B	100	Number of bootstrapped ANNs in the procedure of bias correction of the failure probability estimates	ANN training and refinement
M	1000	Number of SS samples produced in each intermediate conditional region	SS
p_0	0.1	Fraction of the M samples acting as MCMC seeds in each intermediate conditional region of SS	SS
w	0.01	Weight of the original PDF in the fully nonparametric ISD	IS

5.1. Comparison set 1: efficiency of the Importance Sampling stage

The following simulation methods are considered for comparison purposes:

i standard Monte Carlo Simulation (MCS).
ii Latin Hypercube Sampling (LHS) [65].
iii classical Importance Sampling (IS), where the "design point" of the system is adopted as the "center" of the ISD in the standard Normal space (we hereafter refer to this technique as IS-Des) [90,91].
iv Adaptive Kernel-based Importance Sampling (AK-IS), where we build an estimator of the ideal, zero-variance ISD on the basis of a weighed sum of (optimized) Gaussian kernels centered on some failure samples produced by a Markov chain [93,94].
v Subset Simulation (SS) [67,68] (see Section 3.1.1).
vi Optimized Line Sampling (LS), where the failure domain F is "explored" by lines (instead of random points): we optimally identify an "important direction" that points to the failure domain of interest and we solve a number of conditional "one-dimensional" problems along such direction [82].
vii Adaptive Metamodel-based Subset Importance Sampling (AM-SIS).

Details about approaches i)–vi) are not given here for brevity: the interested reader is referred to the cited references.

The AM-SIS algorithm is run with the parameters set at the values reported in Table 4. As suggested by Ref. [119], we set parameters N_{TR}^{0} and N_{add} equal to three times the dimension n_i of the input vector $\boldsymbol{x}$ (i.e., $n_i = 9$ in this case). Parameter $\varepsilon_{P(F)}^{th}$ is employed to verify the convergence of the ANN failure probability estimates $\hat{P}(F)_{ANN}^{N_T}$ during the refinement stage. If $\varepsilon_{P(F)}^{th}$ is too high (e.g., $\varepsilon_{P(F)}^{th} = 0.1$), the algorithm can converge quickly towards a location of the uncertain input space that is far from the failure region F (which results in biased estimates $\hat{P}(F)_{ANN}^{N_T}$); conversely, a very low value (e.g., $\varepsilon_{P(F)}^{th} = 0.0001$) could involve too much computational time (due to an excessive increase in the number t of iterations). The value $\varepsilon_{P(F)}^{th} = 0.01$ has been found to represent a satisfactory trade-off between ANN accuracy and acceptable computational effort.

Notice that the number n_i of inputs to the ANN regression model is equal to 9 (i.e., the number of uncertain inputs in Table 3 of Section 4.2), whereas the number n_o 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 4.2). The common error back-propagation algorithm based on RMSE minimization is applied to train the ANNs [116]. Notice that a *regularization* term is added in the RMSE function in order to avoid *overfitting* of the data, which is frequent in the presence of small-sized training sets (as those of the present application). Also, the number n_h of hidden neurons is optimally selected (at each iteration of the algorithm) as the one that minimizes the ANN Root Mean Squared Error (RMSE) (under the "constraint" that the total number of ANN adjustable parameters must be always smaller than the cardinality of the training data set).

We have performed $S = 10{,}000$ independent runs of each method and in each simulation $s = 1, 2, \ldots, S$, we have obtained an estimate $\hat{P}(F)_s^{N_T}$ of the functional failure probability $P(F)$ (in practice, we have built an empirical distribution of the failure probability estimator $\hat{P}(F)^{N_T}$). We have then computed the following performance indices: (i) the expected value $E[\hat{P}(F)^{N_T}]$ of the failure probability estimator $\hat{P}(F)^{N_T}$; (ii) its standard deviation $\sigma[\hat{P}(F)^{N_T}]$ and (iii) the Figure Of Merit (FOM) defined as $\mathrm{FOM} = \frac{1}{\sigma^2[\hat{P}(F)^{N_T}]\cdot t_{comp}}$, where t_{comp} is the computational time required by the simulation method. Since $\sigma^2[\hat{P}(F)^{N_T}] \propto N_T$ and approximately $t_{comp} \propto N_T$, the FOM is almost independent of N_T. Obviously, the higher the FOM, the higher the computational efficiency of the approach.

It is very important to notice that differently from Ref. [100], at each simulation run $s = 1, 2, \ldots, S$, *both* stages of the AM-SIS method are repeated (i.e., the construction of the quasi-optimal ISD and the IS phases). In this way, the impact of the "*variability*" of the ISD construction step is also included in the standard deviation $\sigma[\hat{P}(F)^{N_T}]$ of $\hat{P}(F)^{N_T}$. This is fundamental since it allows also an "*indirect*" assessment of the effectiveness of this stage of the algorithm (besides that of IS) on the failure probability estimation process.

Table 5
Values of the performance indicators $E[\hat{P}(F)^{N_T}]$ and $\sigma[\hat{P}(F)^{N_T}]$ and FOM obtained with $N_T = 3200$ samples by methods i)–vii).

Functional failure probability ("True" value, $P(F) = 1.8089 \cdot 10^{-7}$)					
			Performance indicators ($N_T = 3200$; $S = 10{,}000$)		
Method	$N_{c,P(F)}$	$N_{c,aux}$	$E[\hat{P}(F)^{N_T}]$	$\sigma[\hat{P}(F)^{N_T}]$	*FOM*
Standard MCS (i)	$N_T = 3200$	0	$1.5627 \cdot 10^{-7}$	$8.8357 \cdot 10^{-6}$	$8.0057 \cdot 10^{5}$
LHS (ii)	$N_T = 3200$	0	$1.7875 \cdot 10^{-7}$	$8.2655 \cdot 10^{-6}$	$9.1483 \cdot 10^{5}$
IS-Des (iii)	$N_T = 3200$	461.7	$1.8080 \cdot 10^{-7}$	$2.8529 \cdot 10^{-8}$	$8.2386 \cdot 10^{10}$
AK-IS (iv)	$N_T = 3200$	461.7	$1.8199 \cdot 10^{-7}$	$2.3130 \cdot 10^{-8}$	$1.2564 \cdot 10^{11}$
SS (v)	$N_T = 3200$	0	$1.9880 \cdot 10^{-7}$	$1.0967 \cdot 10^{-7}$	$5.1964 \cdot 10^{9}$
Optimized LS (vi)	$2 \cdot N_T = 6400$	461.7	$1.7997 \cdot 10^{-7}$	$8.0194 \cdot 10^{-9}$	$4.9540 \cdot 10^{11}$
AM-SIS (vii)	$2 \cdot N_T = 6400$	461.7	$1.8087 \cdot 10^{-7}$	$3.8506 \cdot 10^{-9}$	$1.0866 \cdot 10^{12}$

The SS-based ANN refinement phase converges after $t = 16.1$ iterations (on average) and $N_{c,aux} = 461.7$ calls (on average) to the original, long-running T-H model code. The samples $\{\mathbf{z}_F^m : m = 1, 2, ..., M = 1000\}$ thereby generated are, then, used to construct the quasi-optimal ISD for the subsequent IS step. As highlighted in Section 3.1, the *original* T-H code (*not* the ANN metamodel) is used in the IS-based failure probability estimation phase. Table 5 reports the values of the performance indicators $E[\hat{P}(F)^{N_T}]$, $\sigma[\hat{P}(F)^{N_T}]$ and FOM obtained with $N_T = 3200$ samples by the simulation methods i)-vii) (notice that since N_T is the same for all the simulation methods, performance indicators $E[\hat{P}(F)^{N_T}]$ and $\sigma[\hat{P}(F)^{N_T}]$ can be compared fairly). We include also the total number of T-H simulations required by each method, which is the *critical* parameter driving the overall computational cost (i.e., t_{comp}) in those cases where a single simulation of the model code lasts several hours. In particular, $N_{c,P(F)}$ represents the number of runs used by the algorithm *only* to *assess* the failure probability $P(F)$. On the other hand, $N_{c,aux}$ represents the number of *additional* (*auxiliary*) code simulations required to *set up* the method: for example, for IS-Des, AK-IS and AM-SIS, $N_{c,aux}$ code runs are employed to construct the corresponding Importance Sampling Densities (ISDs) (i.e., by identification of the "design point" of the problem, by the construction of Markov chains and by adaptive Subset Simulation, respectively); instead, for optimized LS, $N_{c,aux}$ code runs are employed to identify the LS "important direction" by Genetic Algorithm (GA)-based minimization of the variance of the LS failure probability estimator. In this respect, two further considerations have to be made. First, in order for the comparison to be fair, the same amount of additional code simulations $N_{c,aux}$ has been employed for the IS-Des (iii), AK-IS (iv), optimized LS (vi) and AM-SIS (vii) methods: this has been obtained by using the *same* ANN metamodel (constructed by the adaptive SS-based strategy of Section 3.1) in the "set up stage" of *all* the above mentioned methods. Secondly, it is worth noting that for both the optimized LS and the AM-SIS approaches we use $N_{c,P(F)} = 2 \bullet N_T$ T-H analyses to *assess* the failure probability $P(F)$ with N_T random samples. Actually, for both methods we need to calculate N_T conditional "one-dimensional" failure probability estimates by resorting to a linear or quadratic interpolation of the passive system performance function $Y(\boldsymbol{x})$: this procedure requires *two* or *three* calls of the model code for *each* random sample drawn (see the Appendix and Ref. [82] for thorough mathematical details).

It can be seen that in the present case the AM-SIS method provides far *more precise* failure probability estimates than all the other approaches: actually, the standard deviation $\sigma[\hat{P}(F)^{N_T}]$ is about 3–30 times smaller than that of the other "advanced" methods iii)-vi) (i.e., IS-Des, AK-IS, SS and optimized LS), whereas it is around 3 *orders of magnitude* smaller than that of the "classical" sampling schemes i) and ii) (i.e., standard MCS and LHS). Finally, although the computational cost associated to the AM-SIS method is higher than that of the other techniques (because the total number of calls to the T-H code is more than two times larger and because the evaluation of the ISD $\hat{g}_w^*(\mathbf{x})$ (6) can be cumbersome in high-dimensional problems), the overall computational efficiency of the approach is still consistently higher: actually, the FOM is about 2–210 times larger than that of the other "advanced" methods iii)-vi), whereas it is even 6 *orders of magnitude* larger than that of the "classical" sampling schemes i) and ii).

The previous set of comparisons seems to suggest that in this particular case study, the AM-SIS method presents higher computational efficiency than the other simulation methods considered. However, in general this should be achieved with a *small* number of samples (and, thus, of T-H model evaluations: say, few tens or hundreds depending on the application), because in practice the T-H computer codes may require several hours to run a single simulation [28,29]. Thus, as in Ref. [100], we consider here a practical situation where the number $N_{c,P(F)}$ of calls to the T-H code allowed for assessing the *small* failure probability $P(F) = 1.8089 \cdot 10^{-7}$ is set to few tens (e.g., $N_{c,P(F)} = 20$ in this case). The results are summarized in Table 6.

Notice that when $N_T = 20$ simulations are used, standard MCS and LHS cannot obviously generate any failure sample, so that $E[\hat{P}(F)^{N_T}]$ is equal to 0. This result is expected: actually, the assessment of probabilities near 10^{-7} by means of standard MCS and LHS with $N_T = 20$ samples is not efficient, because on average $20 \bullet 10^{-7} = 0.000002$–0 failure samples are generated in the failure region of interest. Even considering all the $S = 10{,}000$ repetitions, on average $20 \bullet 10{,}000 \bullet 10^{-7} = 0.02$–0 failure samples are generated in this application. Also, notice that in this case SS is not applicable. Actually, at least $M = 100$ samples have to be generated in *each* subset F_i, $i = 1, 2, ..., n_F$, to produce reliable failure probability estimates: thus, if a failure probability $P(F)$ around 10^{-7} has to be estimated (which is often the case for passive safety systems), then *at least* $M \cdot n_F = 100 \cdot 7 = 700$ samples have to be generated (see Section 3.1.1). On the contrary, employing ISDs (for IS-Des,

Table 6
Values of the performance indicators $E[\hat{P}(F)^{N_T}]$ and $\sigma[\hat{P}(F)^{N_T}]$ and FOM obtained with $N_{c,P(F)} = 20$ T-H code runs by methods i)–vii) [100].

Functional failure probability ("True" value, $P(F) = 1.8089 \cdot 10^{-7}$)					
			Performance indicators ($N_{c,P(F)} = 20$; $S = 10{,}000$)		
Method	N_T	$N_{c,aux}$	$E[\hat{P}(F)^{N_T}]$	$\sigma[\hat{P}(F)^{N_T}]$	*FOM*
Standard MCS (i)	$N_{c,P(F)} = 20$	0	0	$1.2185 \cdot 10^{-4}$	$6.7352 \cdot 10^{5}$
LHS (ii)	$N_{c,P(F)} = 20$	0	0	$1.1352 \cdot 10^{-4}$	$7.7599 \cdot 10^{5}$
IS-Des (iii)	$N_{c,P(F)} = 20$	461.7	$1.7641 \cdot 10^{-7}$	$5.8805 \cdot 10^{-7}$	$2.7556 \cdot 10^{10}$
AK-IS (iv)	$N_{c,P(F)} = 20$	461.7	$1.8097 \cdot 10^{-7}$	$2.8350 \cdot 10^{-7}$	$1.1830 \cdot 10^{11}$
SS (v)	–	0	–	–	–
Optimized LS (vi)	$N_{c,P(F)}/2 = 10$	461.7	$1.8155 \cdot 10^{-7}$	$1.3576 \cdot 10^{-7}$	$5.7594 \cdot 10^{11}$
AM-SIS (vii)	$N_{c,P(F)}/2 = 10$	461.7	$1.8088 \cdot 10^{-7}$	$5.9837 \cdot 10^{-8}$	$1.1565 \cdot 10^{12}$

AK-IS and AM-SIS) or preferential lines (for optimized LS) to explore the failure domain F allows *accurate* estimations of very small failure probabilities even with very few samples: actually, it can be seen that the expected value $E[\hat{P}(F)^{N_T}]$ is very close to the truth (i.e., $P(F) = 1.8089 \cdot 10^{-7}$) for all the above mentioned advanced MCS techniques.

It can be seen that in the present case study the AM-SIS method again provides *more precise* failure probability estimates than the other approaches: actually, the standard deviation $\sigma[\hat{P}(F)^{N_T}]$ is about 2–10 times smaller than that of the other efficient MCS methods iii)-vi) (i.e., IS-Des, AK-IS and optimized LS), whereas it is about 3 *orders of magnitude* smaller than that of the classical sampling schemes i) and ii) (i.e., standard MCS and LHS). As before, also the global computational efficiency of the AM-SIS approach is still consistently higher: actually, the FOM is about 2 to 42 times larger than that of methods iii)-vi), and it is 6 *orders of magnitude* larger than that of sampling schemes i) and ii) [100].

In conclusion, the results obtained in this particular case study seem to confirm the possibility of obtaining robust estimates of small failure probabilities by the AM-SIS method with a very small number N_T of samples drawn and with a very small number N_c of calls to the original long-running code: actually, $N_c = N_{c,aux} + N_{c,P(F)} = 461.7 + 20 = 481.7$ simulations (on average) have turned out to be sufficient for accurately and precisely assessing a failure probability of the order of 10^{-7}.

Then, on the basis of the satisfactory performances obtained by AM-SIS, we are allowed to formulate a positive statement with respect to the possibility of applying the approach to other *realistic, nonlinear* and *non-monotonous* problems of interest in nuclear passive systems risk analysis.

5.2. Comparison set 2: efficiency of the ISD construction stage

In this Section, comparisons are made to methods combining Subset Simulation (SS) and metamodels that are quite close to stage 1 of the proposed approach (i.e., to the quasi-optimal ISD construction phase of Section 3.1). In particular, the following techniques are considered:

i the ^{2}SMART algorithm [105], where SS and Support Vector Machines (SVMs) are combined in an original active learning scheme. The idea is to place sequentially as few training points as possible in the uncertain variable space, in order to build a *sequence* of SVM metamodels as accurate surrogates of the SS intermediate thresholds. For each step of the algorithm, conditional probabilities are estimated by simulations at negligible computational cost on these SVM regression models;

ii the Neural Network-based Subset Simulation (SS-NN) method [113], whose idea is to transfer information from the Metropolis–Hastings algorithm to the ANN training process in order to effectively build the metamodel in a sequence of *moving ranges*: such "training windows" are identified as the upper and lower bounds of the uncertain variables generated by the Markov chains at *each* intermediate SS level. In more detail, the following strategy is adopted (referred to as 'SS-NN2' in Ref. [113]). Each initial seed of the Markov chains is taken as the center of a closed *subset* of the uncertain parameter space, defined by lower and upper bounds located at a properly selected distance. The ANN is then effectively trained at each subset with a negligible computational cost using an LHS design of experiment. The ANN is then used as a robust metamodel in order to increase the SS efficiency by significantly enriching the sample space at each SS level with a minimum additional computational cost. In order to achieve that, a large number of samples (with respect to the "standard" SS procedure) is generated at each subsequent SS level using the MCMC algorithm and the corresponding failure function values are evaluated with the trained ANN instead of the real numerical model [113].

Further details are not given here for the sake of brevity: the interested reader is referred to the cited references. It is however important to notice that these methods differ from AM-SIS in what follows: (a) they develop one (or more) metamodel(s) for *each* SS intermediate level, rather than through iterations of the *entire* SS; (b) the trained metamodel is used in the failure probability *estimation* phase. Finally, notice that this type of comparison was not included in the previous paper by the authors [100].

As before, $S = 100$ independent runs of each method are carried out and in each simulation $s = 1, 2, \ldots, S$, an estimate $\hat{P}(F)_s^{N_T}$ of the functional failure probability $P(F)$ is computed.

Table 7
Values of the performance indicators $E[\hat{P}(F)^{N_T}]$ and $\sigma[\hat{P}(F)^{N_T}]$ and FOM obtained by the ^{2}SMART, SS-NN2 and AM-SIS methods.

Functional failure probability ("True" value, $P(F) = 1.8089 \cdot 10^{-7}$)						
				Performance indicators ($S = 100$)		
Method	N_c	$N_{c,P(F)}$	$N_{c,aux}$	$E[\hat{P}(F)^{N_T}]$	$\sigma[\hat{P}(F)^{N_T}]$	*FOM*
^{2}SMART	4270	–	–	$1.7972 \cdot 10^{-7}$	$1.8471 \cdot 10^{-8}$	$1.0414 \cdot 10^{12}$
SS-NN2 ($M = 6100$)	4270	–	–	$1.8280 \cdot 10^{-7}$	$8.5600 \cdot 10^{-8}$	$6.4766 \cdot 10^{10}$
SS-NN2 ($M = 61{,}000$)	4270	–	–	$1.7892 \cdot 10^{-7}$	$2.7385 \cdot 10^{-8}$	$1.0181 \cdot 10^{11}$
AM-SIS	4270	$2N_T = 3829.9$	440.1	$1.8085 \cdot 10^{-7}$	$5.6399 \cdot 10^{-9}$	$1.0833 \cdot 10^{12}$

The parameters of the ^{2}SMART algorithm are set to the (quasi-optimal) values suggested in Ref. [105]. An average number of $N_c = 4270$ system model code evaluations is carried out for each replicated simulation run s (i.e., around N_c/nint $\{-\log_{10}[P(F)]\} = 610$ evaluations for each SS intermediate region).[3] For the comparison to be fair, also the SS-NN2 method is run using 610 training points (i.e., real model evaluations) per SS level. Two configurations are considered: in particular, $M = 6100$ and 61,000 random samples are generated by MCMC in each SS intermediate region and evaluated by the trained ANN, respectively. Finally, the AM-SIS algorithm is run with the parameters set at the values reported in Table 4. For each iteration $s = 1, 2, \ldots, S$, the ISD construction stage of AM-SIS is run until convergence and the corresponding number $N_{c,aux}$ of system model evaluation is recorded: in the present application, the algorithm converges after $t = 15.3$ iterations on average (for a total of $N_{c,aux} = 440.1$ code runs on average). Then, $N_{c,P(F)} = 4270 - N_{c,aux}$ code runs are employed in the failure probability estimation step (this is done to compare the efficiency of AM-SIS, ^{2}SMART and SS-NN2 using the same total number $N_c = 4270$ of simulations).

The results are reported in Table 7. It can be seen that all the methods produce unbiased (or slightly biased) estimates: the percentage bias is 0.65%, 1.06% and 0.02% for the ^{2}SMART, SS-NN2 and AM-SIS methods, respectively. On the other hand, AM-SIS performs better than the other methods in terms of *precision* of the failure probability estimate: actually, the standard deviation $\sigma[\hat{P}(F)^{N_T}]$ produced by AM-SIS is about 3 smaller than that of ^{2}SMART, whereas it is around 15 and 5 times smaller than that of SS-NN2 with $M = 6100$ and 61,000, respectively. This seems to confirm the positive effect of using the metamodel *only* as a "support tool" for constructing the quasi-optimal ISD and *not* for providing also the *final* estimate $\hat{P}(F)^{N_T}$ of the failure probability (see Section 3.1). From the analysis of the FOM it can be seen instead that the *overall* performance of AM-SIS is only *slightly* better than that of ^{2}SMART: this is due to the (possibly high) computational time required by the evaluation of the multi-dimensional nonparametric ISD (6) (see the Appendix for technical details). On the other hand, it is worth noticing that the (possibly cumbersome) evaluation of the ISD (6) has a relevant impact on the overall computational cost of the method *only* in those cases where the system model code is relatively quick to run (e.g., when it takes less than one minute for a single simulation, like in the present case). In other cases (i.e., when the performance function requires minutes or even hours to be evaluated), the overall computational time is *almost entirely* determined by the total number N_c of code runs (which is actually kept constant in the present comparison). Finally, it is evident that AM-SIS performs significantly better than SS-NN2, as its FOM is about 10–15 times higher.

These analyses confirms the very satisfactory performance of AM-SIS in the present case study, also with respect to other approaches that employ SS and metamodels and that are quite similar to our method in the ISD construction stage.

6. Conclusions

The Adaptive Metamodel-based Subset Importance Sampling (AM-SIS) method, previously developed by the authors [100], has been used for estimating the very low functional failure probability ($P(F) \approx 10^{-7}$) of a nuclear passive safety system.

The method originally combines the powerful features of different techniques of literature in two stages: (1) Subset Simulation (SS) and fast-running Artificial Neural Networks (ANNs) (progressively refined in proximity of the failure region) have been coupled, within an iterative framework, to generate samples distributed according to the optimal, zero-variance Importance Sampling Density (ISD); (2) Importance Sampling (IS) has been performed using a fully nonparametric estimator of the ideal ISD.

In Ref. [100], *only* the efficiency of stage (2) of the algorithm (i.e., random sampling for failure probability estimation) was compared to that of other MC approaches (in particular, standard Monte Carlo Simulation, Latin Hypercube Sampling, several types of Importance Sampling schemes and Line Sampling). In the present work: (a) comparisons have been made *also* to other methods that combine SS and metamodels and that are very close to stage (1) of our approach (in particular, the ^{2}SMART and the Neural Network-based Subset Simulation algorithms); (b) the performance of stage (1) of the AM-SIS method has been assessed *more thoroughly* by carrying out a *larger* set of comparisons (e.g., by considering different sample sizes for failure probability estimation).

[3] The MATLAB code "FERUM4.1" available at Ref. [129] has been used for the implementation of the 2SMART algorithm.

From the results obtained, it can be seen that in the present case study AM-SIS achieves *better* (or at least *comparable*) performances with respect to all the other methods considered. Actually, *higher accuracy* and *precision* in the failure probability estimates have been obtained at a *lower* overall *computational cost* (i.e., by means of a reduced number of random samples drawn and, correspondingly, of expensive system model evaluations): indeed, the two stages of ISD construction (by ANN and SS) and failure probability estimation have required only *few hundreds* of code runs (i.e., about 500).

On the basis of the satisfactory performances obtained by AM-SIS in the present work, we are allowed to formulate a positive statement with respect to the possibility of applying the approach to other *realistic, nonlinear* and *non-monotonous* problems of interest in nuclear passive systems risk analysis; also, they suggest its *possible* integration in full scale PRA applications, provided that the numerous possible accident scenarios and outcomes can be handled computationally in an efficient way. On the other hand, it is worth admitting that more thorough comparisons (carried out on other application examples) would be needed to claim the *overall superiority* of AM-SIS with respect to *all* the other simulation methods considered in this work.

In addition, a comment on the overall *robustness* of the approach is necessary. Actually, AM-SIS improves the efficiency of a failure probability estimation by incorporating into the analysis some information about the problem (i.e., about the passive system of interest). The information here considered is in terms of an "*approximate solution*" (in this case, an ANN metamodel). As far as modeling details are concerned, such approximate solution is expected to capture the main essence of the real system response, at least within some domain of applicability that gives the main contribution to the target failure probability. In this view, the approximate solutions provide a great source of insight for understanding system behavior and they represent invaluable assets in the development of engineering. On the other hand, it should be kept in mind that although the results obtained are very promising, the AM-SIS approach would obviously become defective if the approximate solution used (i.e., the ANN in this case) could no longer "correlate" with the *real* target response. This limit evidently characterizes all the metamodel-based approaches and other efficient simulation methods relying on approximate solutions, such as the Response Conditioning Method (RCM) [75,76] for structural reliability analysis, also developed from SS.

Finally, a word of caution is also in order with respect to the "curse of dimensionality" that affects AM-SIS in two senses: (i) any algorithm involving the construction of a metamodel suffers when the dimensionality of the input space increases, because the DOE becomes sparser with a power law relationship [130]. This issue is unavoidable and limits the application of any algorithm belonging to the metamodel family to engineering problems with a small number of input parameters (say tens), unless some dimensionality reduction strategy is adopted (e.g., principal component analysis or feature selection); (ii) the variance of the fully nonparametric estimator of the optimal ISD can deteriorate with increasing dimension of the input space, as reported in Ref. [99]. These issues will be the objects of future research.

Appendix

Fully nonparametric estimator for the optimal, zero-variance ISD based on the Gibbs Sampler

As described in Section 3.1.2, using a set of points $\{\mathbf{z}_F^m : m = 1, 2, ..., M\}$ approximately distributed as the ideal, zero-variance ISD $g^*(\cdot) = I_F(\cdot)q(\cdot)/P(F)$, an estimator $\hat{g}^*(\mathbf{x})$ for $g^*(\mathbf{x})$ is here constructed as follows [99]:

$$\hat{g}^*(\mathbf{x}) = \frac{1}{M}\sum_{m=1}^{M}\kappa(\mathbf{x}|\mathbf{z}_F^m), \tag{A1}$$

where $\kappa(\mathbf{x}|\mathbf{z}_F^m)$ is the one step ahead transition density $\kappa(\mathbf{x}|\mathbf{z})$ of a Markov chain, considered when the chain is in state $\mathbf{z}_F^m$. Following the suggestion of Ref. [99], in this paper a transition density $\kappa(\mathbf{x}|\mathbf{z}_F^m)$ adopted by the Gibbs Sampler [101] is considered. Since our *target* density is the zero-variance ISD $g^*(\mathbf{x})$, then the transition density $\kappa(\mathbf{x}|\mathbf{z})$ of the corresponding well-known Gibbs Sampler assumes the following form:

$$\kappa(\mathbf{x}|\mathbf{z}) = \prod_{j=1}^{n_i} g^*\left(x_j \middle| x_1, x_2, ..., x_{j-1}, z_{j+1}, z_{j+2}, ..., z_{n_i}\right), \tag{A2}$$

where $g^*(x_j|x_1, x_2, ..., x_{j-1}, z_{j+1}, z_{j+2}, ..., z_{n_i})$ is the *conditional* zero-variance ISD of variable x_j, given that all the other components are fixed to $x_1, x_2, ..., x_{j-1}, z_{j+1}, z_{j+2}, ..., z_{n_i}$.

Tailoring (A2) to the particular case of failure probability estimation of interest to the present paper and remembering that $g^*(\mathbf{x}) = I_F(\mathbf{x})q(\mathbf{x})/P(F)$, then $\hat{g}^*(\mathbf{x})$ (A1) can be rewritten as:

$$\hat{g}^*(\mathbf{x}) = \frac{1}{M}\sum_{m=1}^{M}\prod_{j=1}^{n_i}\frac{q\left(x_j \middle| x_1, x_2, ..., x_{j-1}, z_{F,j+1}^m, ..., z_{F,n_i}^m\right) I_F\left(x_j \middle| x_1, x_2, ..., x_{j-1}, z_{F,j+1}^m, ..., z_{F,n_i}^m\right)}{P\left[Y(\mathbf{x}) > \alpha_Y | x_1, x_2, ..., x_{j-1}, z_{F,j+1}^m, ..., z_{F,n_i}^m\right]}, \tag{A3}$$

where $q(x_j|x_1, x_2, ..., x_{j-1}, z_{F,j+1}^m, ..., z_{F,n_i}^m)$ is the *one-dimensional conditional* PDF of variable x_j; $I_F(x_j|x_1, x_2, ..., x_{j-1}, z_{F,j+1}^m, ..., z_{F,n_i}^m)$ is the indicator of system failure expressed as a "one-dimensional" function of variable x_j, given that all the other variables are *fixed* to the values $x_1, x_2, ..., x_{j-1}, z_{F,j+1}^m, ..., z_{F,n_i}^m$; finally, $P[Y(\mathbf{x}) > \alpha_Y|x_1, x_2, ..., x_{j-1}, z_{F,j+1}^m, ..., z_{F,n_i}^m]$ can be interpreted as a "one-dimensional" conditional probability of system failure computed by letting variable x_j vary according to its uncertainty distribution and keeping all the other variables fixed

to the values $x_1, x_2, ..., x_{j-1}, z^m_{F,j+1}, ..., z^m_{F,n_i}$: in other words, $P[Y(\mathbf{x}) > \alpha_Y | x_1, x_2, ..., x_{j-1}, z^m_{F,j+1}, ..., z^m_{F,n_i}] = \int_{x_j} q(x_j|x_1, x_2, ..., x_{j-1}, z^m_{F,j+1}, ..., z^m_{F,n_i}) \cdot I_F(x_j|x_1, x_2, ..., x_{j-1}, z^m_{F,j+1}, ..., z^m_{F,n_i}) dx_j$. In passing, notice that the argument of the product in (A3) just represents the one-dimensional conditional PDF of variable x_j *truncated* on the failure support.

Finally, by way of example and only for illustration purposes, in what follows the algorithmic steps necessary to draw random samples from the transition density $\kappa(\mathbf{x}|\mathbf{z}^m_F)$ in (A1) and (A3) are shown with reference to three-dimensional input vector $\boldsymbol{x}=\{x_1, x_2, x_3\}$ given $\mathbf{z}^m_F=\{z^m_{F,1}, z^m_{F,2}, z^m_{F,3}\}$ for simplicity. First, a random realization x^*_1 for variable x_1 is sampled from its one-dimensional distribution, conditional on the values $x_2=z^m_{F,2}$ and $x_3=z^m_{F,3}$ and *truncated* on the failure support, i.e., $\cdot \frac{q(x_1|z^m_{F,2}, z^m_{F,3}) \cdot I_F(x_1|z^m_{F,2}, z^m_{F,3})}{\int_{x_1} q(x_1|z^m_{F,2}, z^m_{F,3}) \cdot I_F(x_1|z^m_{F,2}, z^m_{F,3})} = \frac{q(x_1|z^m_{F,2}, z^m_{F,3}) \cdot I_F(x_1|z^m_{F,2}, z^m_{F,3})}{P[Y(\mathbf{x})>\alpha_Y|z^m_{F,2}, z^m_{F,3}]}$. Obviously this requires identifying the *boundaries* of the failure region in the one-dimensional space of variation of x_1, given $x_2=z^m_{F,2}$ and $x_3=z^m_{F,3}$. A procedure similar to that employed by the Line sampling technique can be used to this aim [79]: (i) select two or three different 'trial' values of x_1 (e.g., x^1_1, x^2_1, and x^3_1); (ii) evaluate the system performance function in correspondence of the three 'trial' points $\boldsymbol{x}^1=\{x^1_1, z^m_{F,2}, z^m_{F,3}\}$, $\boldsymbol{x}^2=\{x^2_1, z^m_{F,2}, z^m_{F,3}\}$ and $\boldsymbol{x}^3=\{x^3_1, z^m_{F,2}, z^m_{F,3}\}$, obtaining values $Y(\boldsymbol{x}^1)$, $Y(\boldsymbol{x}^2)$ and $Y(\boldsymbol{x}^3)$; (iii) fit the points $[x^i_1, Y(\boldsymbol{x}^i)]$ thereby obtained by a first or second order polynomial and determine its root(s): such roots represent the boundaries of the failure region in the one-dimensional space of variation of x_1, given $x_2=z^m_{F,2}$ and $x_3=z^m_{F,3}$; (iv) determine $P[Y(\mathbf{x}) > \alpha_Y|z^m_{F,2}, z^m_{F,3}] = \int_{x_1} q(x_1|z^m_{F,2}, z^m_{F,3}) \cdot I_F(x_1|z^m_{F,2}, z^m_{F,3})$ and sample realization x^*_1 from the one-dimensional conditional distribution of x_1 above (for common probability distribution this can be done analytically). Then, once x^*_1 is available, using the same steps (i)-(iv) a random realization x^*_2 for variable x_2 is sampled from its one-dimensional distribution conditional on the values $x_1=x^*_1$ (i.e., the value of x_1 *just* sampled) and $x_3=z^m_{F,3}$ and truncated on the failure support. Finally, having $x_2=x^*_2$, a sample x^*_3 for x_3 is again obtained from its one-dimensional distribution conditional on the values $x_1=x^*_1$ and $x_2=x^*_2$ (i.e., the values of x_1 and x_2 *just* sampled) and *truncated* on the failure support.

References

[1] IAEA, 1991. Safety Related Terms for Advanced Nuclear Plant. IAEA TECDOC-626.
[2] A.K. Nayak, M.R. Gartia, A. Antony, G. Vinod, R.K. Sinha, Passive system reliability analysis using the APSRA methodology, Nucl. Eng. Des. 238 (2008) 1430–1440.
[3] A.K. Nayak, V. Jain, M.R. Gartia, A. Srivastava, H. Prasad, A. Anthony, A.J. Gaikwad, S.K. Bhatia, R.K. Sinha, Reliability assessment of passive isolation condenser system using APSRA methodology, Ann. Nucl. Energy 35 (2008) 2270–2279.
[4] A.K. Nayak, V. Jain, M.R. Gartia, H. Prasad, A. Anthony, S.K. Bhatia, R.K. Sinha, Reliability assessment of passive isolation condenser system of AHWR using APSRA methodology, Reliab. Eng. Syst. Saf. 94 (2009) 1064–1075.
[5] S.K. Ahn, I.S. Kim, K.M. Oh, Deterministic and risk-informed approaches for safety analysis of advanced reactors: part I, deterministic approaches, Reliab. Eng. Syst. Saf. 95 (2010) 451–458.
[6] I.S. Kim, S.K. Ahn, K.M. Oh, Deterministic and risk-informed approaches for safety analysis of advanced reactors: part II, risk-informed approaches, Reliab. Eng. Syst. Saf. 95 (2010) 459–468.
[7] V. Hassija, C.S. Kumar, K. Velusamy, Markov analysis for time dependent success criteria of passive decay heat removal system, Ann. Nucl. Energy 72 (2014) 298–310.
[8] G.E. Apostolakis, The concept of probability in safety assessment of technological systems, Science 250 (1990) 1359.
[9] USNRC, Guidance on the Treatment of Uncertainties Associated with PRAs in Risk-Informed Decision Making, US Nuclear Regulatory Commission, Washington, DC, 2009 NUREG-1855.
[10] L. Burgazzi, Reliability evaluation of passive systems through functional reliability assessment, Nucl. Technol. 144 (2003) 145.
[11] L. Burgazzi, Addressing the uncertainties related to passive system reliability, Prog. Nucl. Energy 49 (2007) 93–102.
[12] L. Burgazzi, Evaluation of the dependencies related to passive system failure, Nucl. Eng. Des. 239 (12) (2009) 3048–3053.
[13] L. Burgazzi, Addressing the challenges posed by advanced reactor passive safety system performance assessment, Nucl. Eng. Des. 241 (5) (2011) 1834–1841.
[14] L. Burgazzi, Reliability study of a special decay heat removal system of a gas-cooled fast reactor demonstrator, Nucl. Eng. Des. 280 (2014) 473–480.
[15] T.H. Woo, U.C. Lee, Dynamical reliability of the passive system in the very high temperature gas cooled reactor, Ann. Nucl. Energy 36 (2009) 1299–1306.
[16] T.H. Woo, U.C. Lee, Passive system reliability in the nuclear power plants (NPPs) using statistical modeling, Nucl. Eng. Des. 239 (12) (2009) 3014–3020.
[17] T.H. Woo, U.C. Lee, The statistical analysis of the passive system reliability in the nuclear power plants (NPPs), Prog. Nucl. Energy 52 (2010) 456–461.
[18] S.J. Han, J.E. Yang, A quantitative evaluation of reliability of passive systems within probabilistic safety assessment framework for VHTR, Ann. Nucl. Energy 37 (2010) 345–358.
[19] E. Zio, N. Pedroni, Building confidence in the reliability assessment of thermal-hydraulic passive systems, Reliab. Eng. Syst. Saf. 94 (2) (2009) 268–281.
[20] D.G. Kang, S.H. Ahn, S.H. Chang, A combined deterministic and probabilistic procedure for safety assessment of beyond design basis accidents in nuclear power plant: Application to ECCS performance assessment for design basis LOCA redefinition, Nucl. Eng. Des. 260 (2013) 165–174.
[21] M. Kumar, A. Chakravarty, A.K. Nayak, H. Prasad, V. Gopika, Reliability assessment of passive containment cooling system of an advanced reactor using APSRA methodology, Nucl. Eng. Des. 278 (2014) 17–28.
[22] M. Wang, S. Qiu, G. Su, W. Tian, Preliminary study of parameter uncertainty influence on pressurized water reactor core design, Prog. Nucl. Energy 68 (2013) 200–209.
[23] M. Wang, D. Zhang, S. Qiu, W. Tian, G. Su, Accident analyses for china pressurizer reactor with an innovative conceptual design of passive residual heat removal system, Nucl. Eng. Des. 272 (2014) 45–52.
[24] Y. Yu, S. Wang, F. Niu, Y. Lai, Z. Kuang, Effect of Nu correlation uncertainty on safety margin for passive containment cooling system in AP1000, Prog. Nucl. Energy 79 (2015) 1–7.
[25] M. Marquès, J.F. Pignatel, P. Saignes, F. D' Auria, L. Burgazzi, C. Müller, R. Bolado-Lavin, C. Kirchsteiger, V. La Lumia, I. Ivanov, Methodology for the reliability evaluation of a passive system and its integration into a probabilistic safety assessment, Nucl. Eng. Des. 235 (2005) 2612–2631.
[26] L. Pagani, G.E. Apostolakis, P. Hejzlar, The impact of uncertainties on the performance of passive systems, Nucl. Technol. 149 (2005) 129–140.
[27] C. Bassi, M. Marquès, 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, 2008 Paper 87376.
[28] F.J. Mackay, G.E. Apostolakis, P. Hejzlar, Incorporating reliability analysis into the design of passive cooling systems with an application to a gas-cooled reactor, Nucl. Eng. Des. 238 (1) (2008) 217–228.
[29] G. Patalano, G.E. Apostolakis, P. Hejzlar, Risk-informed design changes in a passive decay heat removal system, Nucl. Technol. 163 (2008) 191–208.

[30] T.S. Mathews, M. Ramakrishnan, U. Parthasarathy, A.J. Arul, C. Senthil Kumar, Functional reliability analysis of safety grade decay heat removal system of Indian 500 MWe PFBR, Nucl. Eng. Des. 238 (9) (2008) 2369–2376.
[31] T.S. Mathews, A.J. Arul, U. Parthasarathy, C.S. Kumar, M. Ramakrishnan, K.V. Subbaiah, Integration of functional reliability analysis with hardware reliability: An application to safety grade decay heat removal system of Indian 500 MWe PFBR, Ann. Nucl. Energy 36 (2009) 481–492.
[32] T.S. Mathews, A.J. Arul, U. Parthasarathy, C.S. Kumar, K.V. Subbaiah, P. Mohanakrishnan, Passive system reliability analysis using response conditioning method with an application to failure frequency estimation of Decay Heat Removal of PFBR, Nucl. Eng. Des. 241 (6) (2011) 2257–2270.
[33] C.J. Fong, G.E. Apostolakis, D.R. Langewish, P. Hejzlar, N.E. Todreas, M.J. Driscoll, Reliability analysis of a passive cooling system using a response surface with an application to the flexible conversion ratio reactor, Nucl. Eng. Des. 239 (12) (2009) 2660–2671.
[34] A.J. Arul, N.K. Iyer, K. Velusamy, Adjoint operator approach to functional reliability analysis of passive fluid dynamical systems, Reliab. Eng. Syst. Saf. 94 (2009) 1917–1926.
[35] A.J. Arul, N. Kannan Iyer, K. Velusamy, Efficient reliability estimate of passive thermal hydraulic safety system with automatic differentiation, Nucl. Eng. Des. 240 (10) (2010) 2768–2778.
[36] F. Mezio, M. Grinberg, G. Lorenzo, M. Giménez, Integration of the functional reliability of two passive safety systems to mitigate a SBLOCA+BO in a CAREM-like reactor PSA, Nucl. Eng. Des. 270 (2014) 109–118.
[37] G.I. Schueller, On the treatment of uncertainties in structural mechanics and analysis, Comput. Struct. 85 (2007) 235–243.
[38] E. Zio, N. Pedroni, How to effectively compute the reliability of a thermal-hydraulic passive system, Nucl. Eng. Des. 241 (1) (2011) 310–327.
[39] E. Zio, N. Pedroni, Monte Carlo simulation-based sensitivity analysis of the model of a thermal-hydraulic passive system, Reliab. Eng. Syst. Saf. 107 (2012) 90–106.
[40] C. Bucher, T. Most, A comparison of approximate response function in structural reliability analysis, Probab. Eng. Mech. 23 (2008) 154–163.
[41] A.B. Liel, C.B. Haselton, G.G. Deierlein, J.W. Baker, Incorporating modeling uncertainties in the assessment of seismic collapse risk of buildings, Struct. Saf. 31 (2) (2009) 197–211.
[42] M.E. Kartal, H.B. Başağa, A. Bayraktar, Probabilistic nonlinear analysis of CFR dams by MCS using response surface method, Appl. Math. Model. 35 (6) (2011) 2752–2770.
[43] Y.C. Bai, X. Han, C. Jiang, R.G. Bi, A response-surface-based structural reliability analysis method by using non-probability convex model, Appl. Math. Model. 38 (15–16) (2014) 3834–3847.
[44] R.G. Ghanem, P.D. Spanos, Stochastic Finite elements: a Spectral Approach, Springer, Berlin, 1991.
[45] P. Kersaudy, B. Sudret, N. Varsier, O. Picon, J. Wiart, A new surrogate modeling technique combining Kriging and polynomial chaos expansions – application to uncertainty analysis in computational dosimetry, J. Comput. Phys. 286 (2015) 103–117.
[46] R. Schobi, B. Sudret, J. Wiart, Polynomial-chaos-based Kriging, Int. J. Uncertain. Quantif. (2015) 02/2015. arXiv.org >stat >arXiv: 1502.03939.
[47] B. Sudret, C.V. Mai, Computing derivative-based global sensitivity measures using polynomial chaos expansions, Reliab. Eng. Syst. Saf. 134 (2015) 241–250.
[48] I. Babuska, F. Nobile, R. Tempone, A stochastic collocation method for elliptic partial differential equations with random input data, SIAM J. Numer. Anal. 45 (3) (2007) 1005–1034.
[49] L.W. Ng, M.S. Heldred, Multifidelity uncertainty quantification using non-intrusive polynomial chaos and stochastic collocation, in: Proceedings of the 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference
20th AI 23-26 April 2012, Honolulu, Hawaii, 2012.
[50] J.B. Cardoso, J.R. De Almeida, J.M. Dias, P.G. Coelho, Structural reliability analysis using Monte Carlo simulation and neural networks, Adv. Eng. Softw. 39 (2008) 505–513.
[51] J. Cheng, Q.S. Li, R.C. Xiao, A new artificial neural network-based response surface method for structural reliability analysis, Probab. Eng. Mech. 23 (2008) 51–63.
[52] N. Pedroni, E. Zio, G.E. Apostolakis, Comparison of bootstrapped artificial neural networks and quadratic response surfaces for the estimation of the functional failure probability of a thermal-hydraulic passive system, Reliab. Eng. Syst. Saf. 95 (4) (2010) 386–395.
[53] E. Zio, G.E. Apostolakis, N. Pedroni, Quantitative functional failure analysis of a thermal–hydraulic passive system by means of bootstrapped artificial neural networks, Ann. Nucl. Energy 37 (5) (2010) 639–649.
[54] J.E. Hurtado, S, tructural Reliability—Statistical Learning Perspectives, 17, Springer, 2004 Lecture Notes in Applied and Computational Mechanics.
[55] J.E. Hurtado, Filtered importance sampling with support vector margin: a powerful method for structural reliability analysis, Struct. Saf. 29 (2007) 2–15.
[56] I. Kaymaz, Application of Kriging method to structural reliability problems, Struct. Saf. 27 (2) (2005) 133–151.
[57] B. Bichon, M. Eldred, L. Swiler, S. Mahadevan, J. McFarland, Efficient global reliability analysis for nonlinear implicit performance functions, AIAA J. 46 (10) (2008) 2459–2468.
[58] J. Bect, D. Ginsbourger, L. Li, V. Picheny, E. Vazquez, Sequential design of computer experiments for the estimation of a probability of failure, Stat. Comput. 22 (2012) 773–793.
[59] P. Wang, Z. Lu, Z. Tang, An application of the Kriging method in global sensitivity analysis with parameter uncertainty, Appl. Math. Model. 37 (9) (2013) 6543–6555.
[60] X. Yang, Y. Liu, Y. Zhang, Z. Yue, Probability and convex set hybrid reliability analysis based on active learning Kriging model, Appl. Math. Model. 39 (14) (2015) 3954–3971.
[61] L. Zhang, Z. Lu, P. Wang, Efficient structural reliability analysis method based on advanced Kriging model, Appl. Math. Model. 39 (2) (2015) 781–793.
[62] C.P. Robert, G. Casella, Monte Carlo statistical Methods, 319, Springer, New York, 2004.
[63] G. Rubino, B. Tuffin, Rare Event Simulation Using Monte Carlo Methods, Wiley, New York, 2009.
[64] E. Zio, N. Pedroni, Reliability estimation by advanced Monte Carlo simulation, in: J. Faulin, A.A. Juan, S. Martorell, J.E. Ramirez-Marquez (Eds.), Simulation Methods for Reliability and Availability of Complex Systems (Springer Series in Reliability Engineering), Springer-Verlag, London, United Kingdom, 2010, pp. 3–39.
[65] J.C Helton, F.J. Davis, Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems, Reliab. Eng. Syst. Saf. 81 (2003) 23–69.
[66] M. Munoz Zuniga, et al., Adaptive directional stratification for controlled estimation of the probability of a rare event, Reliab. Eng. Syst. Saf. 96 (12) (2011) 1691–1712.
[67] S.K. Au, J.L. Beck, Estimation of small failure probabilities in high dimensions by subset simulation, Probab. Eng. Mech. 16 (4) (2001) 263–277.
[68] S.K. Au, J.L. Beck, Subset simulation and its application to seismic risk based on dynamic analysis, J. Eng. Mech. 129 (8) (2003) 1–17.
[69] J. Ching, J.L. Beck, S.K. Au, Hybrid subset simulation method for reliability estimation of dynamical systems subject to stochastic excitation, Probab. Eng. Mech. 20 (2005) 199–214.
[70] S.K. Au, Z.H. Wang, S.M. Lo, Compartment fire risk analysis by advanced Monte Carlo simulation, Eng. Struct. 29 (9) (2007) 2381–2390.
[71] F. Cadini, D. Avram, N. Pedroni, E. Zio, Subset Simulation of a reliability model for radioactive waste repository performance assessment, Reliab. Eng. Syst. Saf. 100 (2012) 75–83.
[72] E. Zio, N. Pedroni, Estimation of the functional failure probability of a thermal-hydraulic passive systems by means of subset simulation, Nucl. Eng. Des. 239 (2009) 580–599.
[73] S.K. Au, Y. Wang, Engineering Risk Assessment With Subset Simulation, John Wiley & Sons, 2014.
[74] B. Wang, D. Wang, J. Jiang, J. Zhang, P. Sun, Efficient functional reliability estimation for a passive residual heat removal system with subset simulation based on importance sampling, Prog. Nucl. Energy 78 (2015) 36–46.
[75] S.K. Au, Augmenting approximate solutions for consistent reliability analysis, Probab. Eng. Mech. 22 (1) (2007) 77–87.

[76] D.Q. Li, T. Xiao, Z. Cao, K.K Phoon, C.B. Zhou, Efficient and consistent reliability analysis of soil slope stability using both limit equilibrium analysis and finite element analysis, Appl. Math. Model. 40 (9–10) (2016) 5216–5229.
[77] G.I. Schueller, H.J. Pradlwarter, P.S. Koutsourelakis, A critical appraisal of reliability estimation procedures for high dimensions, Probab. Eng. Mech. 19 (2004) 463–474.
[78] H.J. Pradlwarter, G.I. Schueller, P.S. Koutsourelakis, D.C. Charmpis, Application of line sampling simulation method to reliability benchmark problems, Struct. Saf. 29 (2007) 208–221.
[79] G.I. Schueller, H.J. Pradlwarter, Benchmark study on reliability estimation in higher dimensions of structural systems – an overview, Struct. Saf. 29 (3) (2007) 167–182.
[80] M.A. Valdebenito, H.J. Pradlwarter, G.I. Schueller, The role of the design point for calculating failure probabilities in view of dimensionality and structural nonlinearities, Struct. Saf. 32 (2) (2010) 101–111.
[81] E. Zio, N. Pedroni, Functional failure analysis of a thermal-hydraulic passive system by means of line sampling, Reliab. Eng. Syst. Saf. 94 (11) (2009) 1764–1781.
[82] E. Zio, N. Pedroni, An optimized line sampling method for the estimation of the failure probability of nuclear passive systems, Reliab. Eng. Syst. Saf. 95 (12) (2010) 1300–1313.
[83] B.S. Wang, D.Q. Wang, J. Jiang, J.M. Zhang, Efficient estimation of the functional reliability of a passive system by means of an improved line sampling method, Ann. Nucl. Energy 55 (2013) 9–17.
[84] L. Li, Z. Lu, Interval optimization based line sampling method for fuzzy and random reliability analysis, Appl. Math. Model. 38 (13) (2014) 3124–3135.
[85] Z.I. Botev, D.P. Kroese, Efficient Monte Carlo simulation via the generalized splitting method, Stat. Comput. 22 (1) (2012) 1–16.
[86] Z.I. Botev, et al., Static network reliability estimation via generalized splitting, INFORMS J. Comput. 25 (1) (2013) 56–71.
[87] L. Murray, H. Cancela, G. Rubino, A splitting algorithm for network reliability estimation, IIE Trans. 45 (2) (2013) 177–189.
[88] M. Villén-Altamirano, J. Villén-Altamirano, The rare event simulation method RESTART: efficiency analysis and guidelines for its application, Network Performance Engineering, Springer, Berlin, Heidelberg, 2011, pp. 509–547.
[89] J. Villén-Altamirano, Asymptotic optimality of RESTART estimators in highly dependable systems, Reliab. Eng. Syst. Saf. 130 (2014) 115–124.
[90] S.K. Au, J.L. Beck, Importance sampling in high dimensions, Struct. Saf. 25 (2) (2003) 139–163.
[91] S.K. Au, Probabilistic failure analysis by importance sampling Markov chain simulation, J. Eng. Mech. 130 (3) (2004) 303–311.
[92] S. Asmussen, et al., Efficient simulation of tail probabilities of sums of correlated lognormals, Ann. Oper. Res. 189 (1) (2011) 5–23.
[93] S.K. Au, J.L. Beck, A new adaptive importance sampling scheme for reliability calculations, Struct. Saf. 21 (1999) 135–158.
[94] J. Morio, Extreme quantile estimation with nonparametric adaptive importance sampling, Simul. Model. Pract. Theory 27 (2012) 76–89.
[95] R.Y. Rubinstein, D.P. Kroese, The Cross-Entropy method: a Unified Approach to Combinatorial optimization, Monte-Carlo simulation and Machine Learning, Springer, 2004.
[96] P.-T. De Boer, A tutorial on the cross-entropy method, Ann. Oper. Res. 134 (1) (2005) 19–67.
[97] Z.I. Botev, D.P. Kroese, An efficient algorithm for rare-event probability estimation, combinatorial optimization, and counting, Methodol. Comput. Appl. Probab. 10 (4) (2008) 471–505.
[98] S. Asmussen, P.W. Glynn, Stochastic Simulation: Algorithms and Analysis: Algorithms and Analysis, 57, Springer, 2007.
[99] Z.I. Botev, P. L'Ecuyer, B. Tuffin, Markov chain importance sampling with applications to rare event probability estimation, Stat. Comput. 23 (2) (2013) 271–285.
[100] N. Pedroni, E. Zio, Estimating the small failure probability of a nuclear passive safety system by means of an efficient adaptive metamodel-based subset importance sampling method, in: Luca Podofillini, Bruno Sudret, Bozidar Stojadinovic, Enrico Zio, Wolfgang Kröger (Eds.), Proceedings of the European Safety and RELiability Conference on Safety and Reliability of Complex Engineered Systems, ESREL 2015, Zurich, Switzerland, Taylor and Francis Group, London, UK, 7-10 September 2015, pp. 1853–1861, doi:10.1201/b19094-241.
[101] R.M. Neal, "Probabilistic Inference Using Markov Chain Monte Carlo Methods". Technical Report CRG-TR-93-1 (Department of Computer Science, University of Toronto, 1993)
[102] V.J. Romero, L.P. Swiler, A.A. Giunta, Construction of response surfaces based on progressive-lattice-sampling experimental designs with application to uncertainty propagation, Struct. Saf. 26 (2) (2004) 201–219.
[103] B. Echard, N. Gayton, M. Lemaire, AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation, Struct. Saf. 33 (2011) 145–154.
[104] B. Echard, N. Gayton, M. Lemaire, N. Relun, A combined importance sampling and Kriging reliability method for small failure probabilities with time-demanding numerical methods, Reliab. Eng. Syst. Saf. 111 (2013) 232–240.
[105] J.-M. Bourinet, F. Deheeger, M. Lemaire, Assessing small failure probabilities by combined subset simulation and support vector machines, Struct. Saf. 33 (2011) 343–353.
[106] M. Balesdent, J. Morio, J. Marzat, Kriging-based adaptive importance sampling algorithms for rare event estimation, Struct. Saf. 44 (2013) (2013) 1–10.
[107] V. Dubourg, B. Sudret, F. Deheeger, Metamodel-based importance samplingfor structural reliability analysis, Probab. Eng. Mech. 33 (2013) 47–57.
[108] W. Fauriat, N. Gayton, AK-SYS: an adaptation of the AK-MCS method for system reliability, Reliab. Eng. Syst. Saf. 123 (2014) 137–144.
[109] F. Cadini, F. Santos, E. Zio, An improved adaptive kriging-based importance technique for sampling multiple failure regions of low probability, Reliab. Eng. Syst. Saf. 131 (2014) (2014) 109–117.
[110] F. Cadini, F. Santos, E. Zio, Passive systems failure probability estimation by the meta-AK-IS2 algorithm, Nucl. Eng. Des. 277 (2014) (2014) 203–211.
[111] F. Cadini, A. Gioletta, E. Zio, Improved metamodel-based importance sampling for the performance assessment of radioactive waste repositories, Reliab. Eng. Syst. Saf. 134 (2015) (2015) 188–197.
[112] H. Zhao, Z. Yue, Y. Liu, Z. Gao, Y. Zhang, An efficient reliability method combining adaptive importance sampling and Kriging metamodel, Appl. Math. Model. 39 (7) (2015) 1853–1866.
[113] V. Papadopoulos, D.G. Giovanis, N.D. Lagaros, M. Papadrakakis, Accelerated subset simulation with neural networks for reliability analysis, Comput. Methods Appl. Mech. Eng. 223 (2012) 70–80.
[114] E. Zio, A study of the bootstrap method for estimating the accuracy of artificial neural networks in predicting nuclear transient processes, IEEE Trans. Nucl. Sci. 53 (3) (2006) 1460–1470.
[115] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, 1995.
[116] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning internal representations by error back-propagation, in: D.E. Rumelhart, J.L. McClelland (Eds.), Parallel Distributed Processing: Exploration in the Microstructure of Cognition, 1, MIT Press, Cambridge, MA, 1986.
[117] G. Cybenko, Approximation by superpositions of a sigmoidal function, Math. Control Signals Syst. 2 (1989) 303–314.
[118] P. L'Ecuyer, M. Mandjes, B. Tuffin, Importance sampling and rare event simulation, in: G. Rubino, B. Tuffin (Eds.), Rare Event Simulation Using Monte Carlo Methods, Wiley, 2009, pp. 17–38. Chapter 2.
[119] N. Roussouly, F. Petitjean, M. Salaun, A new adaptive response surface method for reliability analysis, Probab. Eng. Mech. 32 (2013) 103–115.
[120] B. Efron, R.J. Thibshirani, An introduction to the bootstrap, Monographs on Statistics and Applied Probability, 57, Chapman and Hall, New York, 1993.
[121] ThomasM. Cover, JoyA. Thomas, Elements of Information Theory, John Wiley & Sons, 2012.
[122] E. Borgonovo, E. Plischke, Sensitivity analysis: a review of recent advances, Eur. J. Oper. Res. 248 (3) (2016) 869–887. http://dx.doi.org/10.1016/j.ejor.2015.06.032.
[123] A. Saltelli, Global Sensitivity Analysis: The Primer, John Wiley, Chichester, England; Hoboken, NJ, 2008.
[124] B. Sudret, Global sensitivity analysis using polynomial chaos expansions, Reliab. Eng. Syst. Saf. 93 (7) (2008) 964–979.
[125] R.Y. Rubinstein, D.P. Kroese, Simulation and the Monte Carlo Method, second ed., Wiley, New York, 2007.

[126] W. Williams, P. Hejzlar, M.J. Driscoll, W.J. Lee, P. Saha, Analysis of a Con-vection Loop for a GFR Post-LOCA Decay Heat Removal From a Block-Type Core, Department of Nuclear Engineering, 2003 MIT-ANP-TR-095.
[127] P.K. Vijayan, M. Sharma, D. Saha, Steady state and stability characteristics of single-phase natural circulation in a rectangular loop with different heater and cooler orientations, Exp. Therm Fluid Sci. 31 (8) (2007) 925–945.
[128] USNRC, An Approach for Using Probabilistic Risk Assessment in Risk-Informed Decisions on Plant-Specific Changes to the Licensing Basis, US Nuclear Regulatory Commission, Washington, DC, 2002 NUREG-1.174 – Revision 1.
[129] J.-M. Bourinet, C. Mattrand, V. Dubourg, A review of recent features and improvements added to FERUM software, in: Proceedings of the 10th International Conference on Structural Safety and Reliability (ICOSSAR'09), Osaka, Japan, 2009. http://www.ifma.fr/FERUM. (accessed 30.08.16)
[130] F.M. Hemez, S. Atamturktur, The dangers of sparse sampling for the quantifi-cation of margin and uncertainty, Reliab. Eng. Syst. Saf. 96 (2011) 1120–1231.