

Artificial Neural Networks and quadratic Response Surfaces for the functional failure analysis of a thermal-hydraulic passive system

George Apostolakis, Nicola Pedroni, Enrico Zio

► To cite this version:

George Apostolakis, Nicola Pedroni, Enrico Zio. Artificial Neural Networks and quadratic Response Surfaces for the functional failure analysis of a thermal-hydraulic passive system. Proceedings of the 10th International PSAM Conference, 2010, pp.1 - 12. <hal-00721042>

HAL Id: hal-00721042

<https://hal.archives-ouvertes.fr/hal-00721042>

Submitted on 26 Jul 2012

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

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

Artificial Neural Networks and quadratic Response Surfaces for the functional failure analysis of a thermal-hydraulic passive system

George Apostolakis^a, Nicola Pedroni^b, Enrico Zio^{b*}

^aMassachusetts Institute of Technology, Nuclear Science and Engineering Dept., Cambridge, USA

^bPolitecnico di Milano, Energy Department, Milano, Italy

Abstract: In this paper, bootstrapped Artificial Neural Network (ANN) and quadratic Response Surface (RS) empirical regression models are used as fast-running surrogates of a thermal-hydraulic (T-H) system code to reduce the computational burden associated with the estimation of the functional failure probability of a T-H passive system.

The ANN and quadratic RS models are built on few data representative of the input/output nonlinear relationships underlying the T-H code. Once built, these models are used for performing, in reasonable computational time, the numerous system response calculations required for failure probability estimation. A bootstrap of the regression models is implemented for quantifying, in terms of confidence intervals, the uncertainties associated with the estimates provided by ANNs and RSs.

The alternative empirical models are compared on a case study of an emergency passive decay heat removal system of a Gas-cooled Fast Reactor (GFR).

Keywords: epistemic uncertainties, passive system reliability, artificial neural network, quadratic response surface

1. INTRODUCTION

Modern nuclear reactor concepts make use of passive safety features [1], which do not need external input (especially energy) to operate and, thus, are expected to improve the safety of nuclear power plants because of simplicity and reduction of both human interactions and hardware failures [2].

However, the aleatory and epistemic uncertainties involved in the *operation* and *modeling* of passive systems are usually larger than for active systems [3], [4]. Due to these uncertainties, there is a nonzero probability that the physical phenomena involved in the operation of a passive system (e.g., natural circulation) do not occur as expected, thus leading to the failure of performing the intended safety function (e.g., decay heat removal) even if i) safety margins have been dimensioned and ii) no hardware failures occur [4]. In the analysis of such *functional failure* behavior, the passive system is usually modeled by a detailed, mechanistic T-H system code and the probability of failing to perform the required function is estimated based on a Monte Carlo (MC) sample of code runs which propagate the epistemic (state-of-knowledge) uncertainties in the model and in the numerical values of its parameters/variables [2], [5]-[10].

Since the probabilities of functional failure of passive systems are generally very small (e.g., of the order of 10^{-4}), a large number of samples is necessary for acceptable estimation accuracy [11]; given that the time required for each run of the detailed, mechanistic T-H system code is typically of the order of several hours [10], the MC simulation-based procedure typically requires considerable computational efforts.

To tackle the computational issue, efficient sampling techniques can be adopted for obtaining robust estimations with a limited number of input samples. Techniques like Importance Sampling (IS) [12], Stratified Sampling [13] and Latin Hypercube Sampling (LHS) [14] have been widely used in reliability analysis and risk assessment [16]. Recently, advanced sampling methods such as Subset Simulation (SS) [17] and Line Sampling (LS) [18] have been proposed for structural reliability assessment and subsequently applied to the estimation of the functional failure probability of a T-H passive system [19], [20]. These methods have been shown to improve the computational efficiency although there is no indication yet that the number of model evaluations can be reduced to below a few

* enrico.zio@polimi.it

hundreds, which may be mandatory in the presence of computer codes requiring several hours to run a single simulation.

In such cases, the only viable alternative seems that of resorting to fast-running, surrogate regression models, also called response surfaces or meta-models, to approximate the input/output function implemented in the long-running T-H model code, and then substitute it in the passive system functional failure analysis. The construction of such regression models entails running the T-H model code a predetermined, reduced number of times (e.g., 50-100) for specified values of the uncertain input parameters/variables and collecting the corresponding values of the output of interest; then, statistical techniques are employed for fitting the response surface of the regression model to the input/output data generated in the previous step. Several kinds of surrogate meta-models have been recently applied to safety related nuclear, structural and hydrogeological problems, including polynomial RSs [21], Gaussian meta-models [22] and learning statistical models such as ANNs, Radial Basis Functions (RBFs) and Support Vector Machines (SVMs) [23].

In this work, the possibility of using ANNs and quadratic RSs to reduce the computational burden associated to the functional failure analysis of a natural convection-based decay heat removal system of a GFR [5] is investigated. To keep the practical applicability in sight, a small set of input/output data examples is considered available for constructing the ANN and quadratic RS models: different sizes of the (small) data sets are considered to show the effects of this relevant practical aspect. The comparison of the potentials of the two regression techniques in the case at hand is made with respect to the estimation of the functional failure probability of the passive system.

Actually, the use of regression models in safety critical applications like nuclear power plants still raises concerns with regards to the control of their accuracy; in this paper, the bootstrap method is used for quantifying, in terms of *confidence intervals*, the uncertainty associated to the estimates provided by the ANNs and quadratic RSs [24]-[27].

The paper organization is as follows. In Section 2, a snapshot on the functional failure analysis of T-H passive systems is given. Section 3 is devoted to the detailed presentation of the bootstrap-based method for quantifying, in terms of confidence intervals, the model uncertainty associated to the estimates of safety parameters computed by ANN and quadratic RS regression models. In Section 4, the case study of literature concerning the passive cooling of a GFR is presented. In Section 5, the results of the application of bootstrapped ANNs and quadratic RSs to the functional failure probability estimations are compared. Finally, conclusions are provided in the last Section.

2. FUNCTIONAL FAILURE ANALYSIS OF T-H PASSIVE SYSTEMS

The basic steps of the quantitative phase of the functional failure analysis of a T-H passive system are [6]:

1. Detailed modeling of the passive system response by means of a deterministic, best-estimate (typically long-running) T-H code.
2. Identification of the parameters/variables, models and correlations (i.e., the inputs to the T-H code) which contribute to the uncertainty in the results (i.e., the outputs) of the best estimate T-H calculations.
3. Propagation of the uncertainties through the deterministic, long-running T-H code in order to estimate the functional failure probability of the passive system.

Step 3. above relies on multiple (e.g., many thousands) evaluations of the T-H code for different combinations of system inputs; this can render the associated computing cost prohibitive, when the running time for each T-H code simulation takes several hours (which is often the case for T-H passive systems).

The computational issue may be tackled by replacing the long-running, original T-H model code by a fast-running, surrogate regression model (properly built to approximate the output from the true system model). In this paper, classical three-layered feed-forward ANNs [28] and quadratic RSs [21] are considered for this task. The accuracy of the estimates obtained is analyzed by computing a

confidence interval by means of the bootstrap method [24]; a description of this technique is provided in the following Section.

3. BOOTSTRAPPED EMPIRICAL REGRESSION MODELING FOR POINT AND CONFIDENCE INTERVAL EVALUATION

3.1. Empirical regression modeling

As discussed in the previous Section, the computational burden posed by the functional failure analysis of T-H passive systems can be tackled by replacing the original, long-running, T-H system code by a surrogate regression model. Because calculations with the surrogate model can be typically performed quickly, the problem of long simulation times is circumvented.

Let us consider a generic meta-model to be built for performing the task of nonlinear regression, i.e., estimating the nonlinear relationship between a vector of input variables $\mathbf{x} = \{x_1, x_2, \dots, x_j, \dots, x_{n_i}\}$ and a vector of output targets $\mathbf{y} = \{y_1, y_2, \dots, y_l, \dots, y_{n_o}\}$ on the basis of a *finite* (and possibly *reduced*) set of input/output data examples (i.e., patterns), $D_{train} = \{\{\mathbf{x}_p, \mathbf{y}_p\}, p=1, 2, \dots, N_{train}\}$ [25]. It can be assumed that the target vector \mathbf{y} is related to the input vector \mathbf{x} by an unknown nonlinear deterministic function $\boldsymbol{\mu}_y(\mathbf{x})$ corrupted by a noise vector $\boldsymbol{\varepsilon}(\mathbf{x})$, i.e.,

$$\mathbf{y}(\mathbf{x}) = \boldsymbol{\mu}_y(\mathbf{x}) + \boldsymbol{\varepsilon}(\mathbf{x}). \quad (1)$$

Notice that in the present case of T-H passive system functional failure assessment, the nonlinear deterministic function $\boldsymbol{\mu}_y(\mathbf{x})$ in (1) is represented by the complex, long-running T-H mechanistic system code (e.g., RELAP5-3D) and the noise $\boldsymbol{\varepsilon}(\mathbf{x})$ in (1) could be represented by the error introduced by the numerical methods employed to calculate $\boldsymbol{\mu}_y(\mathbf{x})$ [27]; however, for simplicity the model assumption in the following is $\boldsymbol{\varepsilon}(\mathbf{x}) = \mathbf{0}$.

The objective of the regression task is to estimate $\boldsymbol{\mu}_y(\mathbf{x})$ in (1) by means of a regression function $\mathbf{f}(\mathbf{x}, \mathbf{w}^*)$ depending on a set of parameters \mathbf{w}^* to be properly determined on the basis of the available set of data D_{train} ; the algorithm used to identify the set of parameters \mathbf{w}^* is obviously dependent on the nature of the regression model adopted, but in general it aims at minimizing the mean error between the *real* outputs of the T-H code, $\mathbf{y}_p, p = 1, 2, \dots, N_{train}$, and the outputs of the regression model $\hat{\mathbf{y}}_p = \mathbf{f}(\mathbf{x}_p, \mathbf{w}^*)$ corresponding to the inputs $\mathbf{x}_p, p = 1, 2, \dots, N_{train}$; for example, the Root Mean Squared Error (RMSE) can be considered to this purpose [25]:

$$RMSE = \frac{1}{N_{train} \cdot n_o} \sum_{p=1}^{N_{train}} \sum_{l=1}^{n_o} (y_{p,l} - \hat{y}_{p,l})^2. \quad (2)$$

Once a regression model $\mathbf{f}(\mathbf{x}, \mathbf{w}^*)$ is built, it can be used as a simplified, quick-running surrogate of the original, long-running T-H system code, which can significantly reduce the computational burden associated to uncertainty propagation for the estimation of the functional failure of T-H passive systems.

In this work, both quadratic RSs and ANNs are considered. Quadratic RSs are polynomials containing linear terms, squared terms and possibly two-factors interactions between the input variables [21]. ANNs instead are computing devices inspired by the function of the nerve cells in the brain [28]. They are composed of many parallel computing units (called neurons or nodes) interconnected by weighed connections (called synapses). Each of these computing units performs a few simple operations and communicates the results to its neighbouring units. From the mathematical point of view, ANNs consist of a set of nonlinear (i.e., sigmoidal) basis functions with unknown parameters that are adjusted by a process of *training* on many different input/output examples, i.e., an iterative process of regression error minimization [29]. The particular type of ANN employed in this paper is the classical three-layered feed-forward ANN trained by the error back-propagation algorithm.

The details of these two regression models are not reported here for brevity: the interested reader may refer to the cited references and the copious literature in the field.

3.2. The bootstrap method

When using the approximation of the system output provided by an ANN empirical regression model, an additional source of uncertainty is introduced which needs to be evaluated, particularly in safety critical applications like those related to nuclear power plant technology. One way to do this by resorting to bootstrapped ANN regression models [24], i.e., an ensemble of ANN regression models, constructed on different data sets bootstrapped from the original one [25], [27]. The bootstrap method is a distribution-free inference method which requires no prior knowledge about the distribution function of the underlying population [24]. The basic idea is to generate a sample from the observed data by sampling with replacement from the original data set [24]. From the theory and practice of ensemble empirical models, it can be shown that the estimates given by bootstrapped ANN regression models is in general more accurate than the estimate of the best ANN regression model in the bootstrap ensemble of ANN regression models [25], [26].

In what follows, the steps of the bootstrap-based technique of evaluation of the so-called Bootstrap Bias Corrected (BBC) point estimate \hat{Q}_{BBC} of a generic quantity Q (e.g., a safety parameter) by a regression model $f(\mathbf{x}, \mathbf{w}^*)$, and the calculation of the associated BBC Confidence Interval (CI) are reported [25], [27]:

1. Generate a set D_{train} of input/output data examples by sampling N_{train} independent input parameters values $\mathbf{x}_p, p = 1, 2, \dots, N_{train}$, and calculating the corresponding set of N_{train} output vectors $\mathbf{y}_p = \boldsymbol{\mu}_y(\mathbf{x}_p)$ through the mechanistic T-H system code.
2. Build a regression model $f(\mathbf{x}, \mathbf{w}^*)$ on the basis of the entire data set $D_{train} = \{(\mathbf{x}_p, \mathbf{y}_p), p=1, 2, \dots, N_{train}\}$ (step 1. above) in order to obtain a simple, fast-running surrogate of the original T-H system code represented by the unknown nonlinear deterministic function $\boldsymbol{\mu}_y(\mathbf{x})$ in (1).
3. Use the regression model $f(\mathbf{x}, \mathbf{w}^*)$ (step 2. above) in place of the original T-H code to provide a point estimate $\hat{P}(F)$ for $P(F)$.
4. Build an ensemble of B (e.g., $B = 500-1000$) regression models $\{f_b(\mathbf{x}, \mathbf{w}_b^*), b=1, 2, \dots, B\}$ by resorting to the bootstrap method based on random sampling with replacement and use each of the regression models $f_b(\mathbf{x}, \mathbf{w}_b^*), b = 1, 2, \dots, B$, to calculate an estimate $\hat{Q}_b, b = 1, 2, \dots, B$, for the quantity Q of interest: by so doing, a bootstrap-based empirical probability distribution for the quantity Q is produced which is the basis for the construction of the corresponding confidence intervals. In particular, repeat the following steps for $b = 1, 2, \dots, B$:
 - a. Generate a bootstrap data set $D_{train,b} = \{(\mathbf{x}_{p,b}, \mathbf{y}_{p,b}), p=1, 2, \dots, N_{train}\}, b = 1, 2, \dots, B$, by performing random sampling with replacement from the original data set of N_{train} input/output patterns $D_{train} = \{(\mathbf{x}_p, \mathbf{y}_p), p=1, 2, \dots, N_{train}\}$ (steps 1. and 2. above). The data set $D_{train,b}$ is thus constituted by the same number N_{train} of input/output patterns drawn among those in D_{train} although, due to the sampling with replacement, some of the patterns in D_{train} will appear more than once in $D_{train,b}$, whereas some will not appear at all.
 - b. Build a regression model $f_b(\mathbf{x}, \mathbf{w}_b^*), b = 1, 2, \dots, B$, on the basis of the bootstrap data set $D_{train,b} = \{(\mathbf{x}_{p,b}, \mathbf{y}_{p,b}), p=1, 2, \dots, N_{train}\}$ (step 3.a. above).
 - c. Use the regression model $f_b(\mathbf{x}, \mathbf{w}_b^*)$ (step 4.b. above) in place of the original T-H code to provide a point estimate \hat{Q}_b of the quantity of interest Q .
5. Calculate the so-called Bootstrap Bias Corrected (BBC) point estimate \hat{Q}_{BBC} for Q as

$$\hat{Q}_{BBC} = 2\hat{Q} - \hat{Q}_{boot} \quad (3)$$

where \hat{Q} is the estimate obtained with the regression model $f(\mathbf{x}, \mathbf{w}^*)$ trained with the original data set D_{train} (steps 2. and 3. above) and \hat{Q}_{boot} is the average of the B estimates \hat{Q}_b obtained with the B regression models $f_b(\mathbf{x}, \mathbf{w}_b^*), b = 1, 2, \dots, B$ (step 4.c. above), i.e.,

$$Q_{boot} = \frac{1}{B} \sum_{b=1}^B \hat{Q}_b. \quad (4)$$

The BBC estimate \hat{Q}_{BBC} is taken as the final point estimate for Q . The explanation for expression (3) is as follows. It can be demonstrated that if there is a bias in the bootstrap average estimate \hat{Q}_{boot} in (4) compared to the estimate \hat{Q} , obtained with the single regression model $f(\mathbf{x}, \mathbf{w}^*)$ (step 3. above), then the same bias exists in the single estimate \hat{Q} compared to the true value Q of the quantity of interest. Thus, in order to obtain an appropriate, i.e. bias-corrected, estimate \hat{Q}_{BBC} of the quantity of interest Q , the estimate \hat{Q} must be adjusted by subtracting the corresponding bias ($\hat{Q}_{boot} - \hat{Q}$): as a consequence, the final, bias-corrected estimate \hat{Q}_{BBC} is $\hat{Q}_{BBC} = \hat{Q} - (\hat{Q}_{boot} - \hat{Q}) = 2\hat{Q} - \hat{Q}_{boot}$.

6. Calculate the two-sided Bootstrap Bias Corrected (BBC) $100 \cdot (1 - \alpha)\%$ Confidence Interval (CI) for the BBC point estimate in (3) by performing the following steps:

- a. Order the bootstrap estimates \hat{Q}_b , $b = 1, 2, \dots, B$, (step 4.c. above) by increasing values, such that $\hat{Q}_{(i)} = \hat{Q}_b$ for some $b = 1, 2, \dots, B$, and $\hat{Q}_{(1)} < \hat{Q}_{(2)} < \dots < \hat{Q}_{(b)} < \dots < \hat{Q}_{(B)}$.
- b. Identify the $100 \cdot \alpha / 2^{\text{th}}$ and $100 \cdot (1 - \alpha / 2)^{\text{th}}$ quantiles of the bootstrapped empirical probability distribution of Q (step 4. above) as the $[B \cdot \alpha / 2]^{\text{th}}$ and $[B(1 - \alpha / 2)]^{\text{th}}$ elements $\hat{Q}_{([B \cdot \alpha / 2])}$ and $\hat{Q}_{([B(1 - \alpha / 2)])}$, respectively, in the ordered list $\hat{Q}_{(1)} < \hat{Q}_{(2)} < \dots < \hat{Q}_{(b)} < \dots < \hat{Q}_{(B)}$; notice that the symbol $[\cdot]$ stands for ‘‘closest integer’’.
- c. Calculate the two-sided BBC $100 \cdot (1 - \alpha)\%$ CI for \hat{Q}_{BBC} as

$$\left[\hat{Q}_{BBC} - (\hat{Q}_{boot} - \hat{Q}_{([B \cdot \alpha / 2])}), \hat{Q}_{BBC} + (\hat{Q}_{([B(1 - \alpha / 2)])} - \hat{Q}_{boot}) \right]. \quad (5)$$

An important advantage of the bootstrap method is that it provides confidence intervals for a given quantity without making any model assumptions (e.g., normality); a disadvantage is that the computational cost could be high when the set D_{train} and the number of adaptable parameters \mathbf{w}^* in the regression models are large.

4. CASE STUDY

The case study considered in this work concerns the natural convection cooling function in a GFR under a post-Loss Of Coolant Accident (LOCA) condition [5]. The reactor is a 600-MW GFR cooled by helium flowing through separate channels in a silicon carbide matrix core whose design has been the subject of study in the past several years at the Massachusetts Institute of Technology (MIT) [5].

A GFR decay heat removal configuration is shown schematically in Figure 1; in the case of a LOCA, the long-term heat removal is ensured by natural circulation in a given number N_{loops} of identical and parallel loops; only one of the N_{loops} loops is reported for clarity of the picture: the flow path of the cooling helium gas is indicated by the black arrows. The loop has been divided into $N_{sections} = 18$ sections for numerical calculation; technical details about the geometrical and structural properties of these sections are not reported here for brevity: the interested reader may refer to [5].

Table 1: Epistemic uncertainties considered for the 600-MW GFR passive decay heat removal system of Figure 1 [5]

	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%

4.2. Failure criteria of the T-H passive system

The passive decay heat removal system of Figure 1 is considered failed whenever the temperature of the coolant helium leaving the core (item 4 in Figure 1) exceeds either 1200 °C in the hot channel or 850 °C in the average channel.

Indicating by \mathbf{x} the vector of the 9 uncertain system parameters of Table 1 (Section 4.1) and by $T_{out,core}^{hot}(\mathbf{x})$ and $T_{out,core}^{avg}(\mathbf{x})$ the coolant outlet temperatures in the hot and average channels, respectively, the failure region F can be written as follows:

$$F = \{\mathbf{x} : T_{out,core}^{hot}(\mathbf{x}) > 1200\} \cup \{\mathbf{x} : T_{out,core}^{avg}(\mathbf{x}) > 850\}. \quad (6)$$

Notice that, in the notation of the preceding Section 3, $T_{out,core}^{hot}(\mathbf{x}) = y_1(\mathbf{x})$ and $T_{out,core}^{avg}(\mathbf{x}) = y_2(\mathbf{x})$ are the two target outputs of the T-H model.

5. RESULTS

In this Section, the results of the application of bootstrapped ANNs and quadratic RSs for the estimation of the functional failure probability of the passive system in Figure 1 are illustrated. Some details about the construction of the ANN and quadratic RS regression models are given in Section 5.1; the estimation of the probability of functional failure of the system is addressed in Section 5.2. The uncertainties associated to the calculated quantities are estimated by bootstrapping of the regression models, as explained in Section 3.

5.1 Building the regression models

RS and ANN models have been built with training sets $D_{train} = \{(\mathbf{x}_p, \mathbf{y}_p), p = 1, 2, \dots, N_{train}\}$ of input/output data examples of different sizes $N_{train} = 20, 30, 50, 70, 100$; this has allowed extensive testing of the capability of the regression models to reproduce the outputs of the nonlinear T-H model code, based on different (small) numbers of example data. For each size N_{train} of data set, a Latin Hypercube Sample (LHS) of the 9 uncertain inputs has been drawn, $\mathbf{x}_p = \{x_{1,p}, x_{2,p}, \dots, x_{j,p}, \dots, x_{9,p}\}$, $p = 1, 2, \dots, N_{train}$. Then, the T-H model code has been run with each of the input vectors \mathbf{x}_p , $p = 1, 2, \dots, N_{train}$, to obtain the corresponding bidimensional output vectors $\mathbf{y}_p = \boldsymbol{\mu}_y(\mathbf{x}_p) = \{y_{1,p}, y_{2,p}\}$, $p = 1, 2, \dots, N_{train}$ (in the present case study, the number n_o of outputs is equal to 2, i.e., the hot- and average-channel coolant outlet temperatures, as explained in Section 4.2). The training data set $D_{train} = \{(\mathbf{x}_p, \mathbf{y}_p), p = 1, 2, \dots, N_{train}\}$ thereby obtained has been used to calibrate the adjustable parameters \mathbf{w}^* of the regression models, for best fitting the T-H model code data. More specifically, the straightforward least squares method has been used to find the parameters of the quadratic RSs [21] and the common error back-propagation algorithm has been applied to *train* the ANNs [29]. Note that a *single* ANN can be trained to estimate both outputs of the model here of interest, whereas a specific quadratic RS must be developed for each output to be estimated.

The choice of the ANN architecture is critical for the regression accuracy. In particular, the number of neurons in the network determines the number of adjustable parameters available to optimally fit the

complicated, nonlinear T-H model code response surface by interpolation of the available training data. The number of neurons in the input layer is $n_i = 9$, equal to the number of uncertain input parameters; the number n_o of outputs is equal to 2, the outputs of interest; the number n_h of nodes in the hidden layer is 4 for $N_{train} = 20, 30, 70$ and 100, whereas it is 5 for $N_{train} = 50$, determined by trial-and-error.

A validation data set $D_{val} = \{(\mathbf{x}_p, \mathbf{y}_p), p = 1, 2, \dots, N_{val}\}$ (different from the training set D_{train}) is used to monitor the accuracy of the ANN model during the training procedure: in practice, the RMSE (2) is computed on D_{val} at different phases of the training procedure. At the beginning, the RMSE computed on the validation set D_{val} typically decreases together with the RMSE computed on the training set D_{train} ; then, when the ANN regression model starts overfitting the data, the RMSE calculated on the validation set D_{val} starts increasing: this is the time to stop the training algorithm [28]. In this work, the size N_{val} of the validation set is set to 20 for all sizes N_{train} of the data set D_{train} considered, which means 20 additional runs of the T-H model code. On the contrary, it is worth noting that, to the best of the authors' knowledge, no method of this kind is available for polynomial RSs.

As measures of the ANN and RS model accuracy, the commonly adopted coefficient of determination R^2 and RMSE have been computed for each output y_l , $l = 1, 2$, on a new data set $D_{test} = \{(\mathbf{x}_p, \mathbf{y}_p), p = 1, 2, \dots, N_{test}\}$ of size $N_{test} = 20$, purposely generated for testing the regression models built [22], and thus different from those used during training and validation.

Table 2 reports the values of the coefficient of determination R^2 and of the RMSE associated to the estimates of the hot- and average- channel coolant outlet temperatures $T_{out,core}^{hot}$ and $T_{out,core}^{avg}$, respectively, computed on the test set D_{test} by the ANN and quadratic RS models built on data sets D_{train} of different sizes $N_{train} = 20, 30, 50, 70, 100$; the number of adjustable parameters w^* included in the two regression models is also reported for comparison purposes.

Table 2: Coefficient of determination R^2 and RMSE associated to the estimates of the hot- and average-channel coolant outlet temperatures $T_{out,core}^{hot}$ and $T_{out,core}^{avg}$, respectively, computed on the test set D_{test} of size $N_{test} = 20$ by the ANN and quadratic RS models built on data sets D_{train} of different sizes $N_{train} = 20, 30, 50, 70, 100$; the number of adjustable parameters w^* included in the two regression models is also reported for comparison purposes

Artificial Neural Network (ANN)							
			R^2		RMSE [°C]		
N_{train}	N_{val}	N_{test}	Number of adjustable parameters w^*	$T_{out,core}^{hot}$	$T_{out,core}^{avg}$	$T_{out,core}^{hot}$	$T_{out,core}^{avg}$
20	20	20	50	0.8937	0.8956	38.5	18.8
30	20	20	50	0.9140	0.8982	34.7	18.6
50	20	20	62	0.9822	0.9779	15.8	8.7
70	20	20	50	0.9891	0.9833	12.4	6.8
100	20	20	50	0.9897	0.9866	12.0	6.3
Quadratic Response Surface (RS)							
			R^2		RMSE [°C]		
N_{train}	N_{val}	N_{test}	Number of adjustable parameters w^*	$T_{out,core}^{hot}$	$T_{out,core}^{avg}$	$T_{out,core}^{hot}$	$T_{out,core}^{avg}$
20	0	20	55	0.5971	0.7914	75.0	26.6
30	0	20	55	0.8075	0.9348	51.9	14.8
50	0	20	55	0.9280	0.9353	31.7	14.6
70	0	20	55	0.9293	0.9356	31.4	14.3
100	0	20	55	0.9305	0.9496	31.2	13.1

The ANN outperforms the RS in all the cases considered: for example, for $N_{train} = 100$, the coefficients of determination R^2 produced by the ANN and the quadratic RS models for the hot-channel coolant outlet temperature $T_{out,core}^{hot}$ are 0.9897 and 0.9305, respectively, whereas the corresponding RMSEs are

12.0 °C and 31.2 °C, respectively. This result is due to the higher flexibility in modeling complex nonlinear input/output relationships offered by the ANN with respect to the quadratic RS: the ANN structure made of a large number of adaptable connections (i.e., the synapses) among nonlinear operating units (i.e., the neurons) allows fitting complex nonlinear functions with an accuracy which is superior to that of a plain quadratic regression model. Actually, if the original T-H model is not quadratic (which is often the case in practice), a second-order polynomial RS cannot be a *consistent* estimator, i.e., the quadratic RS estimates may never converge to the true values of the original T-H model outputs, even for a very large number of input/output data examples, in the limit for $N_{train} \rightarrow \infty$. On the contrary, ANNs have been demonstrated to be universal approximants of *continuous* nonlinear functions (under mild mathematical conditions) [31], i.e., in principle, an ANN model *with a properly selected architecture* can be a consistent estimator of any continuous nonlinear function, e.g. any nonlinear T-H code simulating the system of interest.

5.2 Functional failure probability estimation

In this Section, the bootstrapped ANNs and quadratic RSs are compared in the task of estimating the functional failure probability of the 600-MW GFR passive decay heat removal system of Figure 1. For illustration purposes, a configuration with $N_{loops} = 3$ loops is considered for the passive system of Figure 1.

Table 3 reports the values of the Bootstrap Bias Corrected (BBC) point estimates $\hat{P}(F)_{BBC}$ of the functional failure probability $P(F)$ obtained with $N_T = 500000$ estimations from the bootstrapped ANNs and quadratic RSs built on $N_{train} = 20, 30, 50, 70$ and 100 data examples; the corresponding Bootstrap Bias Corrected (BBC) 95% Confidence Intervals (CIs) are also reported. Notice that the “true” (i.e., reference) value of the functional failure probability $P(F)$ (i.e., $P(F) = 3.34 \cdot 10^{-4}$) has been obtained with a very large number N_T (i.e., $N_T = 500000$) of simulations of the original T-H code to provide a robust term of comparison. Actually, the T-H code here employed runs fast enough to allow repetitive calculations (one code run lasts on average 3 seconds on a Pentium 4 CPU 3.00GHz): the computational time required by this reference analysis is thus $500000 \cdot 3 \text{ s} = 1500000 \text{ s} \approx 417 \text{ h}$.

Table 3: Bootstrap Bias Corrected (BBC) point estimates $\hat{P}(F)_{BBC}$ and BBC 95% Confidence Intervals (CIs) for the functional failure probability $P(F)$ (i.e., $P(F) = 3.34 \cdot 10^{-4}$) obtained with $N_T = 500000$ estimations from bootstrapped ANNs (left) and RSs (right) built on $N_{train} = 20, 30, 50, 70$ and 100 data examples

Failure probability (“True” value, $P(F) = 3.34 \cdot 10^{-4}$; CPU time $\approx 417 \text{ h}$)					
Bootstrapped Artificial Neural Networks (ANNs)					
N_{train}	N_{val}	N_{test}	BBC point estimate, $\hat{P}(F)_{BBC}$	BBC-95% CI	
20	20	20	$1.01 \cdot 10^{-4}$	[0, $7.91 \cdot 10^{-4}$]	
30	20	20	$1.53 \cdot 10^{-4}$	[0, $6.70 \cdot 10^{-4}$]	
50	20	20	$2.45 \cdot 10^{-4}$	[$8.03 \cdot 10^{-5}$, $4.27 \cdot 10^{-4}$]	
70	20	20	$3.01 \cdot 10^{-4}$	[$2.00 \cdot 10^{-4}$, $4.20 \cdot 10^{-4}$]	
100	20	20	$3.59 \cdot 10^{-4}$	[$2.55 \cdot 10^{-4}$, $4.12 \cdot 10^{-4}$]	
Bootstrapped Quadratic Response Surface (RS)					
N_{train}	N_{val}	N_{test}	BBC point estimate, $\hat{P}(F)_{BBC}$	BBC-95% CI	
20	0	20	$9.81 \cdot 10^{-5}$	[0, $8.39 \cdot 10^{-4}$]	
30	0	20	$1.00 \cdot 10^{-4}$	[0, $7.77 \cdot 10^{-4}$]	
50	0	20	$2.15 \cdot 10^{-4}$	[$7.43 \cdot 10^{-5}$, $5.07 \cdot 10^{-4}$]	
70	0	20	$2.39 \cdot 10^{-4}$	[$1.16 \cdot 10^{-4}$, $4.61 \cdot 10^{-4}$]	
100	0	20	$3.17 \cdot 10^{-4}$	[$2.20 \cdot 10^{-4}$, $4.40 \cdot 10^{-4}$]	

It can be seen that as the size of the training sample N_{train} increases, both the ANN and quadratic RS provide increasingly accurate estimates of the true functional failure probability $P(F)$, as one would expect. On the other hand, in the cases of small training sets (e.g., $N_{train} = 20, 30$ and 50) the functional failure probabilities are significantly underestimated by both the bootstrapped ANN and the quadratic

RS models (e.g., the BBC point estimates $\hat{P}(F)_{BBC}$ for $P(F)$ lie between $9.81 \cdot 10^{-5}$ and $2.45 \cdot 10^{-4}$) and the associated uncertainties are quite large (e.g., the widths of the corresponding BBC 95% CIs are between $3.47 \cdot 10^{-4}$ and $7.91 \cdot 10^{-4}$). Two considerations seem in order with respect to these results. First, in these cases of small data sets available the analyst would still be able to correctly estimate the order of magnitude of a *small* failure probability (i.e., $P(F) \sim 10^{-4}$), in spite of the *low* number of runs of the T-H code performed to generate the $N_{train} = 20, 30$ or 50 input/output examples; second, the accuracy of an estimate should be evaluated in relation to the requirements of the specific application; for example, although the confidence interval provided by the bootstrapped ANNs trained with $N_{train} = 50$ samples ranges from $8.03 \cdot 10^{-5}$ to $4.27 \cdot 10^{-4}$, this variability might be acceptable for demonstrating that the system satisfies the target safety goals.

It is worth noting that although bootstrapped ANNs provide better estimates and lower model uncertainties than quadratic RSs, the difference in the performances of the two regression models is less evident than in the case of coolant temperature estimation (Table 2). This may be due to the fact that estimating the value of the functional failure probability $P(F)$ is a simpler task than estimating the exact values of the corresponding coolant outlet temperatures. For example, let the true value of the hot channel coolant outlet temperature be 1250 °C and the corresponding estimate by the regression model be 1500 °C: in such a case, the estimate is absolutely *inaccurate* in itself, but “*exact*” for the purpose of functional failure probability estimation with respect to a failure threshold of 1200 °C.

Finally, the computational times associated to the calculation of the BBC point estimates $\hat{P}(F)_{BBC}$ for $P(F)$, and the corresponding BBC 95% CIs, are compared for the two bootstrapped regression models with reference to the case of $N_{train} = 100$, by way of example: the overall CPU times required by the use of bootstrapped ANNs and RSs are on average 2.22 h and 0.43 h, respectively. These values include the time required for: i) generating the $N_{train} + N_{val} + N_{test}$ input/output examples, by running the T-H code: the corresponding CPU times are on average $(100 + 20 + 20) \cdot 3 = 420$ s = 7 min ≈ 0.12 h and $(100 + 0 + 20) \cdot 3 = 360$ s = 6 min ≈ 0.10 h for the ANNs and the RSs, respectively; ii) training the bootstrapped ensemble of $B = 1000$ ANN and RS regression models by means of the error back-propagation algorithm and the least squares method, respectively: the corresponding CPU times are on average 2 h and 0.25 h for the ANNs and the RSs, respectively; iii) performing $N_T = 500000$ evaluations of each of the $B = 1000$ bootstrapped ANN and RS regression models: the corresponding CPU times are on average 6 min (i.e., 0.1 h) and 4.5 min (i.e., about 0.08 h) for the ANNs and the RSs, respectively.

The overall CPU times required by the use of bootstrapped ANNs (i.e., approximately 2.22 h) and quadratic RSs (i.e., approximately 0.43 h) is about 188 and 970 times, respectively, lower than that required by the use of the original T-H model code (i.e., approximately 417 h). The CPU time required by the ANNs is about 5 times larger than that required by the quadratic RSs, mainly due to the elaborate training algorithm needed to build the structurally complex neural model.

6. CONCLUSIONS

In this paper, ANNs and quadratic RSs have been compared when used within a MC simulation scheme for estimating the probability of functional failure of a T-H passive system. A case study involving the natural convection cooling function in a GFR after a LOCA has been taken as reference. ANN and quadratic RS models have been constructed on the basis of sets of data of limited, varying sizes, which represent examples of the nonlinear relationships between 9 uncertain inputs and 2 relevant outputs of the T-H model code (i.e., the hot- and average-channel coolant outlet temperatures). Once built, such models have been used, in place of the original T-H model code, to estimate the functional failure probability of the system. In all the cases considered, the results have demonstrated that ANNs outperform quadratic RSs in terms of estimation accuracy. Due to their flexibility in nonlinear modeling, ANNs have been shown to provide more reliable estimates than quadratic RSs even when they are trained with very low numbers of data examples (e.g., $20, 30$ or 50) from the original T-H model code.

The bootstrap method has been employed to estimate confidence intervals on the quantities computed: this uncertainty quantification is of paramount importance in safety critical applications, in particular

when few data examples are used. In this regard, bootstrapped ANNs have been shown to produce *narrower* confidence intervals than bootstrapped quadratic RSs in all the analyses performed. On the basis of the results obtained, bootstrapped ANNs can be considered more effective than quadratic RSs in the estimation of the functional failure probability of T-H passive systems (while quantifying the uncertainty associated to the results) because they provide more *accurate* (i.e., estimates are closer to the true values) and *precise* (i.e., confidence intervals are narrower) estimates than quadratic RSs; on the other hand, the computational time required by bootstrapped ANNs is somewhat longer than that required by quadratic RSs, due to the elaborate training algorithm for building the structurally complex neural model.

References

- [1] IAEA, “*Safety related terms for advanced nuclear plant*”, IAEA TECDOC-626, (1991).
- [2] 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*”, *Annals of Nuclear Energy*, 36, pp. 481-492, (2009).
- [3] J. Helton, W.L. Oberkampf, “*Alternative representations of epistemic uncertainties*”, *Reliability Engineering and System Safety*, 85 (Special Issue), pp. 1-10, (2004).
- [4] L. Burgazzi, “*Addressing the uncertainties related to passive system reliability*”, *Progress in Nuclear Energy*, 49, pp. 93-102, (2007).
- [5] L. Pagani, G.E. Apostolakis and P. Hejzlar, “*The impact of uncertainties on the performance of passive systems*”, *Nuclear Technology*, 149, pp. 129-140, (2005).
- [6] 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, Paper 87376, (2008).
- [7] 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*”, *Nuclear Engineering and Design*, 238(1), pp. 217-228, (2008).
- [8] G. Patalano, G.E. Apostolakis, P. Hejzlar, “*Risk-informed design changes in a passive decay heat removal system*”, *Nuclear Technology*, 163, pp. 191-208, (2008).
- [9] A.J. Arul, N.K. Iyer, K. Velusamy, “*Adjoint operator approach to functional reliability analysis of passive fluid dynamical systems*”, *Reliability Engineering and System Safety*, 94, pp. 1917-1926, (2009).
- [10] 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*”, *Nuclear Engineering and Design*, 239(12), pp. 2660-2671, (2009).
- [11] G.I. Schueller, “*Efficient Monte Carlo simulation procedures in structural uncertainty and reliability analysis - recent advances*”, *Journal of Structural Engineering and Mechanics*, 32(1), pp. 1–20, (2009).
- [12] S.K. Au, “*Probabilistic failure analysis by Importance Sampling Markov Chain Monte Carlo*”, *Journal of Engineering Mechanics*, 130(3), pp. 303-311, (2004).
- [13] D.G. Cacuci, M. Ionescu-Bujor, “*A comparative review of sensitivity and uncertainty analysis of large scale systems – II: Statistical methods*”, *Nuclear Science and Engineering*, 147, pp. 204-217, (2004).
- [14] J.C. Helton, F.J. Davis, “*Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems*”, *Reliability Engineering and System Safety*, 81(1), pp. 23-69, (2003).
- [15] S.K. Au, J.L. Beck, “*Subset Simulation and its application to seismic risk based on dynamic analysis*”, *Journal of Engineering Mechanics*, 129(8), pp. 1-17, (2003).
- [16] J.C. Helton, C. Sallaberry, “*Computational implementation of sampling-based approaches to the calculation of expected dose in performance assessments for the proposed high-level radioactive waste repository at Yucca Mountain, Nevada*”, *Reliability Engineering and System Safety*, 94, pp. 699-721, (2009).

- [17] S.K. Au, J.L. Beck, “*Estimation of small failure probabilities in high dimensions by subset simulation*”, Probabilistic Engineering Mechanics, 16(4), pp. 263-277, (2001).
- [18] H.J. Pradlwarter, M.F. Pellissetti, C.A. Schenk, G.I. Schueller, A. Kreis, S. Fransen, A. Calvi, M. Klein, “*Realistic and efficient reliability estimation for aerospace structures*”, Computer Methods in Applied Mechanics and Engineering, 194, pp. 1597-1617, (2005).
- [19] E. Zio, N. Pedroni, “*Estimation of the functional failure probability of a thermal-hydraulic passive systems by means of Subset Simulation*”, Nuclear Engineering and Design, 239, pp. 580-599, (2009).
- [20] E. Zio, N. Pedroni, “*Functional failure analysis of a thermal-hydraulic passive system by means of Line Sampling*”, Reliability Engineering and System Safety, 94(11), pp. 1764-1781, (2009).
- [21] A.B. Liel, C.B. Haselton, G.G. Deierlein, J.W. Baker, “*Incorporating modeling uncertainties in the assessment of seismic collapse risk of buildings*”, Structural Safety, 31(2), pp. 197-211, (2009).
- [22] A. Marrel, B. Iooss, B. Laurent, O. Roustant, “*Calculations of Sobol indices for the Gaussian process metamodel*”, Reliability Engineering and System Safety, 94, pp. 742-751, (2009).
- [23] J.B. Cardoso, J.R. De Almeida, J.M. Dias, P.G. Coelho, “*Structural reliability analysis using Monte Carlo simulation and neural networks*”, Advances in Engineering Software, 39, pp. 505-513, (2008).
- [24] B. Efron, R.J. Tibshirani, “*An introduction to the bootstrap. Monographs on statistics and applied probability 57*”, Chapman and Hall, 1993, New York.
- [25] E. Zio, “*A study of the bootstrap method for estimating the accuracy of artificial neural networks in predicting nuclear transient processes*”, IEEE Transactions on Nuclear Science, 53(3), pp.1460-1470, (2006).
- [26] P. Secchi, E. Zio, F. Di Maio, “*Quantifying uncertainties in the estimation of safety parameters by using bootstrapped artificial neural networks*”, Annals of Nuclear Energy, 35, pp. 2338-2350, (2008).
- [27] C.B. Storlie, L.P. Swiler, J.C. Helton, C.J. Sallaberry, “*Implementation and evaluation of nonparametric regression procedures for sensitivity analysis of computationally demanding models*”, Reliability Engineering and System Safety; 94, pp. 1735-1763, (2009).
- [28] C.M. Bishop, “*Neural Networks for pattern recognition*”, Oxford University Press, 1995.
- [29] 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 (vol. 1)*”, MIT Press, 1986, Cambridge (MA).
- [30] NUREG-CR-6850, “*EPRI/NRC-RES Fire PRA methodology for nuclear power facilities, Volume 2: detailed methodology*”, US Nuclear Regulatory Commission, (2005).
- [31] G. Cybenko, “*Approximation by superposition of sigmoidal functions*”, Mathematics of Control, Signals and Systems, 2, pp. 303-314, (1989).