Artificial Neural Network Uncertainty Quantification for the Sensitivity Analysis of the SIXEP Model

U Oparaji[1, 2], R.J Sheu[2], M Bankhead[3], J Austin[3], E Patelli[1]

[1] *Institute for Risk and Uncertainty, University of Liverpool, Liverpool L69 7ZF, United Kingdom*

[2] *Institute of Nuclear Engineering and Science, National Tsing-Hua University, 101 Sec. 2, Kung Fu Road, Hsinchu, Taiwan, R.O.C.*

[3] *National Nuclear Laboratory, Chadwick House, Warrington Rd, Birchwood Park, Warrington, Cheshire, WA3 6AE, United Kingdom*

Artificial Neural Network (ANNs) are largely used to replace computational expensive models. However, these surrogate models can introduce additional uncertainty and variability on the output of interest. In particular, when it is used to estimate the sensitivity of a model, the result obtained might give a false confidence, when the analyst is not aware of the uncertainty introduced. Hence, it is of fundamental importance to first check the validity of the ANN, and then quantify the uncertainties associated with the point estimates of the model. In this paper, an ANN is constructed based on selected data representative of the input/output non-linear relationship of an underlying waste management model (SIXEP). Once constructed, the ANN is used for performing sensitivity analysis in reasonable computational time. Finally, the bootstrapped technique is adopted to quantify the uncertainty introduced by the surrogate model in terms of bias corrected confidence intervals.

I. INTRODUCTION

Various mathematical models use a large number of uncertain parameters as inputs. The impact of input parameter uncertainty leads to variability in the model output quantity of interest. A user of the underlying model of interest might want to identify the contributions of each uncertain input parameter to the variability in the output. This can be achieved by first modelling the uncertain input parameters as random variables, whose probability distribution characterizes the uncertainty in the parameter values. Using Sobol' approach to sensitivity analysis, the variance in the output of interest is decomposed into components, and apportioned to each uncertain input parameter (i.e. sensitivity indices). The sensitivity indices measure the fractional contribution of the input parameters to the variability in the output.

Nevertheless, the computational costs required to compute sensitivity indices are in huge order of magnitude. Specifically, when estimating the Sobol' indices using the variance based method, the number of model runs follows the mathematical relationship $E_M = (p+2) \times N$, where p represents the number of model uncertain parameters, and N the number of samples required. The number of samples N is usually proportional to the dimension of the model being analysed. For instance, Patelli et al., (2012) [1] required $\leq 10^5$ samples to correctly estimate the sensitivity indices of the Gravity Field and Steady-State Ocean Circulation Explorer (GOCE) satellite due to the complexity of the finite element (FE) model used (i.e. $p > 3000$). On the contrary, Baroni et al., (2014) [2] found out that the sensitivity indices convergence was reached using $N = 1024$ samples for a model of $p = 5$ uncertain parameters. Generally speaking, the computational cost required for performing sensitivity analysis can vary amongst different sets of models. This is due to the complexity of the model (i.e. large parameters), and the time required for a single deterministic simulation in each model. Hence, in order to tackle these huge computation restrictions, alternative methods that significantly reduce the computational burden must be sourced out.

In the last few decades, the use of meta-modelling tools such as Artificial Neural Networks (ANNs) have emerged and proven to be advantageous in terms of reducing the computational effort required for computing sensitivity indices. For example, Patelli et al., (2012) [1] have used ANNs as substitutes to replace an expensive FE model to speed up their analysis. The ANN used in their work was constructed based on a few data representatives of the input-output nonlinear relationships of the FE model. Once constructed, these fast-running meta-models are used for performing the numerous model evaluations required for estimating sensitivity indices. The main advantage of using an ANN for this application is its ability to capture and learn functional relationships contained within a set of training data, such that it can generalise the non-linear model relationship, and make predictions about the population from which the training data set originated from. The generalisability of a trained ANN, is measured by the predictive performance on samples not contained within the training data set.

Although ANN have good generalisability from a small set of training data, these models can introduce additional uncertainty and variability in the output quantity of interest. For example, when used in sensitivity indices of a model, the value(s) obtained might not be accurate as a result of these uncertainties present within the ANN. Hence, it is of fundamental importance to first check the validity of the ANN and then quantify the uncertainty associated with the model. Therefore, in this paper, an ANN is adopted to replicate a radioactive waste management model to reduce the computational burden required to efficiently compute the sensitivity indices of the model. Furthermore, the bootstrap approach [3] for propagating the uncertainties in the point estimate of a model will be applied to the constructed ANN, in order to quantify the uncertainties in the sensitivity indices estimates using confidence intervals. This paper is organized as follows: In Section 2 a succinct theory of global sensitivity analysis is introduced. In Section 3, the bootstrap technique used in quantifying uncertainties associated to the sensitivity indices estimates from the ANN will be discussed. This is followed by an application of the bootstrap technique in a real case study concerning the SIXEP in section 4. Section 5 presents the results obtained from this analysis. Finally, section 6 concludes the paper.

II. GLOBAL SENSITIVITY ANALYSIS

Global sensitivity analysis methods are based on the analysis of variance (ANOVA), which estimates the fractional contribution of each uncertain input parameter to the variance of the output quantity of interest [4]. The first order and total effect indices [5] are mostly used in this regard, where each index is computed by evaluating a multidimensional integral via Monte Carlo. The variance approach to sensitivity analysis decomposes the variance of the model's output into fractions that can be attributed to the uncertain input parameters of the model [5]. The total variance of the model's output is expressed as:

$$Var(Y) = \sum_{i=1}^{p} V_i + \sum_{i<j}^{p} V_{ij} + \cdots + V_{12\ldots p} \tag{1}$$

where p denotes number of uncertain parameters, V_i the partial variance for single parameters and, V_{ij} represents interactions between parameters. Mathematically, V_i and V_{ij} are expressed as:

$$V_i = Var_{X_i}(E_{X\sim i}(Y \mid X_i)) \tag{2}$$

$$V_{ij} = Var_{X_{ij}}(E_{X\sim ij}(Y \mid X_{ij})) \tag{3}$$

The first order sensitivity measure is defined as:

$$S_i = \frac{V_i}{Var(Y)} \tag{4}$$

Eq.4 measures the effect of varying the input parameters of the model, without considering interactions between parameters of the model. Similarly, the second order sensitivity measure that measures the sensitivity due to interactions between parameters of the model is defined as:

$$S_{ij} = \frac{V_{ij}}{Var(Y)} \tag{5}$$

On the other hand, the total effect sensitivity measure T_i that measures the contribution of a single parameter as well as interactions between parameters in the model is mathematically defined as:

$$T_i = 1 - \frac{Var_{X_{\sim i}}[E_{X\sim i}(Y|X_{\sim i})]}{Var(Y)} \tag{6}$$

As stated previously, when estimating S_i and T_i, the number of model evaluations follows the relationship $E_M = (p+2) \times N$, where N is the number of samples required and p the number of uncertain input parameters. For a robust estimate of the sensitivity measures, N should be greater than 10^5. However, this is computationally expensive, therefore, fast-running meta-models such as Artificial Neural Networks can be used to reduce the computational burden.

III. BOOTSTRAP ARTIFICIAL NEURAL NETWORKS

III.A. Artificial Neural Network Modelling

In this paper, we consider an ANN with n number of hidden layers to be built for performing a task of nonlinear regression such as estimating a nonlinear relationship between state variables $x = [x_1, x_2, \dots, x_j, \dots x_p]$ and a vector of output quantities of interest $y = [y_1, y_2, \dots, y_l, \dots y_p]$ on the basis of a finite training data set D_{train} provided from the original model [6]. The ANN has been constructed in the OpenCossan software developed by the COSSAN working group at the Institute for Risk and Uncertainty, University of Liverpool [7-8]. It is assumed that the input vector x is related to the target vector y by a nonlinear deterministic function $\mu_y(x)$ corrupted by a white Gaussian noise $\varepsilon(x)$. For simplicity, it is assumed that the noise $\varepsilon(x)$ is represented by a normal distribution with a mean equal to zero [9]. Hence, the aim of constructing the regression model is to estimate $\mu_y(x)$ by the means of a regression function, which depends on set of weight parameters to be determined based on the finite training data set D_{train}. The algorithm used to adjust/train the set of parameters is dependent on the type of surrogate-model being constructed. Specifically, for Artificial Neural Networks, the training algorithm minimizes the error between the output of the expensive model and that of the ANN by adjusting the weight parameters. The error is expressed mathematically as:

$$RMSE = \frac{1}{N_{train} \cdot p} \sum_{z=1}^{N_{train}} \sum_{l=1}^{p} \left(y_{z,l} - \hat{y}_{z,l}\right)^2 \tag{7}$$

Once constructed, the ANN is used in place of the expensive model to estimate any quantity of interest U such as the sensitivity measures Si and T_i.

III.B. The Bootstrap Technique

The procedures for using the bootstrap method to assess the uncertainty initiated by the use an ANN are reported in the following steps [6]. First, generate a set of training data D_{train} of input/output data by assigning uniform distributions to the uncertain input parameters of the model and sampling each of the input parameters N_{train} number of times using the Latin-Hypercube sampling algorithm to obtain the input vectors x_k, $k=1,2,\dots,N_{train}$, then evaluate the model with all the sample realizations obtained. Second, construct an ANN on the basis of the entire data set $D_{train}= \{x, y\}$ obtained in the first step in order to obtain a fast-running meta-model represented by $\mu_y(x)$. Third, use the ANN constructed in step 2 to estimate quantities of interest $\hat{U}$ of the quantity U. Specifically, draw samples of N_T new input vectors x_r, $r = 1,2,\dots, N_T$, from the corresponding probability distributions, and feed the ANN with the sampled values to obtain output vectors y_r, $r = 1,2,\dots, N_T$ for computing the estimate $\hat{U}$ for U. Since ANNs are fast running models, the computational cost is less than the expensive model, even with large number of samples N_T. This is followed by constructing a group of B (typically in the order of 500-1000) ANN models by randomly sampling the values with replacement from D_{train}, and using each of the bootstrapped ANN models to calculate an estimate $\hat{U}_b$, $b = 1,2,\dots,B$, for the quantity of interest U. By doing so, a bootstrap-based empirical probability distribution for the quantity U is produced which is used for calculating the confidence interval. In particular, repeat steps for b = 1,2,…,B. Fifth, generate a bootstrap data set $D_{train,b} = \{(x_{k,b}, y_{k,b}), k = 1,2,\dots,N_{train}\}$, $b=1,2,\dots,B$ by random sampling with replacement from the original training data set D_{train}. The data set in $D_{train,b}$ will constitute of the same number of data as D_{train}. Although due to the sampling of data with replacement, some patterns in $D_{train,b}$ will appear more than once as in D_{train} and some patterns in D_{train} will not appear at all in $D_{train,b}$. Then, construct B number of ANNs and train each ANN model with the bootstrap data set $D_{train,b} = \{(x_{k,b}, y_{k,b}), k = 1,2,\dots,N_{train}\}$, $b=1,2,\dots,B$. Compute the point estimates such as sensitivity indices $\hat{U}_b$, b=1,2,…,B from each bootstrapped ANN. Calculate the Bootstrap Bias Corrected (BBC) point estimate U_{BBC} for U as:

$$\hat{U}_{BBC} = 2\hat{U} - \hat{U}_{boot} \tag{8}$$

where $\hat{U}$ is the estimate obtained from ANN trained with training data set D_{train} and $\hat{U}_{boot}$ is the average of the B estimates $\hat{U}_b$ obtained from the B bootstrapped ANNs. Its relationship is given as:

$$\hat{U}_{boot} = \frac{1}{B}\sum_{b=1}^{B}\hat{U}_b \tag{9}$$

$\hat{U}_{BBC}$ in Eq. (8) is used as the point estimate of U, this is due to the fact that if a bias exists in the estimate $\hat{U}_{boot}$ compared with $\hat{U}$, then the same bias exists in the estimate $\hat{U}$ when compared to the true estimate U for the quantity of interest. Finally, calculate the confidence interval for $\hat{U}_{BBC}$ using the following steps:
Firstly, order the point estimates form each bootstrapped ANN $\hat{U}_b$, $b=1,2,\ldots,B$ in increasing order of magnitude $\hat{U}_1 < \hat{U}_2 < \ldots < \hat{U}_b < \ldots \hat{U}_B$. Second, identify the $100.\alpha/2^{th}$ and $100. (1\text{-}\alpha/2)^{th}$ quantiles of the bootstrapped empirical probability distribution of U as the $B.\alpha/2^{th}$ and $B. (1\text{-}\alpha/2)^{th}$ elements in the ordered list as the $\hat{U}_{([B.\alpha/2])}$ and $\hat{U}_{([B.(1-\alpha/2)])}$ respectively. Then, calculate the confidence interval of $\hat{U}_{BBC}$ as; $[\hat{U}_{BBC}-(\hat{U}_{boot}- \hat{U}_{([B.\alpha/2])}),\ \hat{U}_{BBC}+(\hat{U}_{([B.(1-\alpha/2)])} - \hat{U}_{boot})]$.

IV. CASE STUDY

The case study considered in this work uses a model of the Site Ion Xchange Effluent Plant (SIXEP) [10], a plant which is situated on the nuclear fuel reprocessing and decommissioning site at Sellafield, U.K. Since its introduction in the mid 1970's, it has been integral in reducing discharges form the site, to less than 1% of the discharges prior to it coming on-line. Discharge predictions underpinned by this model are used in the real-world to underpin discharges for site data that is publicly available from the UK environmental agency. However, the model used in this case study uses a reduced data set that has been desensitised relevant to the real-world application. The SIXEP manages effluent produced by a number of plants across Sellafield, removing radioactivity from liquid feeds. A schematic diagram of the SIXEP process is shown in Figure 1.

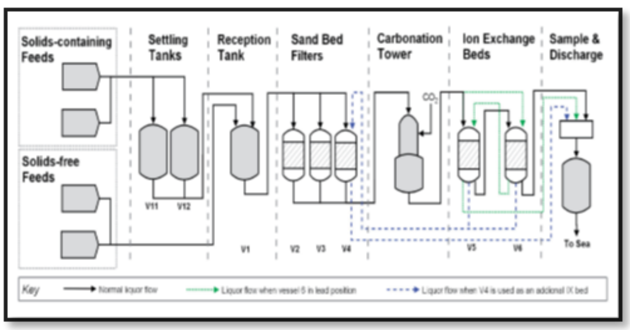

Fig 1. SIXEP Process Diagram [10]

The feeds into the SIXEP contain particulate materials, and a number of soluble radioactive isotopes which are predominantly, Caesium-137 and Strontium-90. These soluble radioactive species are removed from the liquid effluent using an ion exchange media loaded in 2 ion exchange beds which operates in series (one lead bed and one lag bed). The lead bed is replaced with fresh media when it is exhausted, and the bed that previously operated in the lag position is promoted to the lead position. The filtration and carbonation steps are present to protect the Xchange beds and have a secondary benefit of removing actinides. In order to ensure continued removal of these two key radioactive isotopes, the plant is routinely operated on the basis of feeds meeting a set of Conditions for Acceptance (CfA). These CfA define the feed envelop in terms of the acceptable concentrations of inactive species which affect the efficiency of the process. The SIXEP model is being used to test new feed compositions to prove assurance that the plant can continue to operate effectively, i.e. ensuring the discharges of Caesium-137 and Strontium-90 are kept within the required limit.

IV.A. Uncertainties Affecting the SIXEP

Conversely, there is uncertainty of the future feeds composition arising from the Sellafield site, leading to variability in the activity levels of Caesium-137 and Strontium-90 and other soluble species that affect the removal of these isotopes. This variability can lead to undesirable consequences (i.e. the discharges of the two afore-mentioned radionuclides exceeds their desired levels). Hence, we wish to evaluate if it is practical to incorporate this uncertainty into studies when using the SIXEP model to assess the risk involved, and identify the parameters that contribute significantly to this variability. It is to be noted that the uncertainty considered to affect the plant feeds are aleatory (i.e. random) in nature [11]. The consideration of these type of uncertainty in the SIXEP model leads to defining of a state vector $\boldsymbol{x}$ of 18 uncertain inputs of the SIXEP model $\boldsymbol{x} = \{x_n: n = 1, 2,..,18\}$, which are assumed to be described by the distributions given in Table.1. These uncertain inputs map out two output quantities of interest defined by the vector $\boldsymbol{y} = \{y_z \colon z=1,2\}$.The outputs of interest represents the maximum concentration of Caesium-137 and Strontium-90 respectively after approximately 540 days. Fig.2. shows a deterministic simulation from the SIXEP model using a set of representative mean values specified in Table.1.

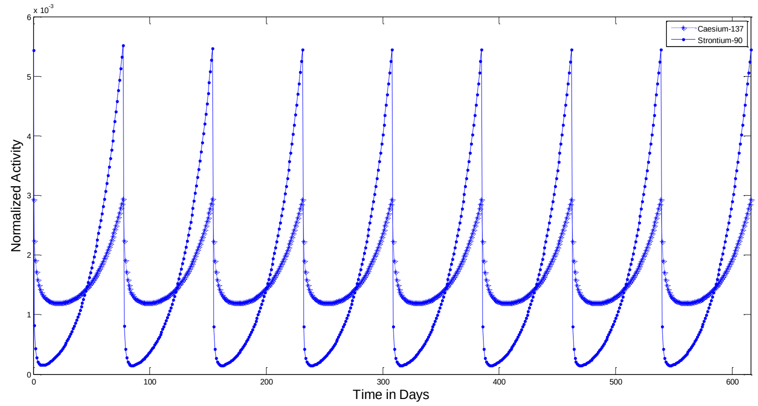

Fig. 2. Deterministic Simulation from the SIXEP Model

The results from the deterministic evaluation of the SIXEP model shown in Fig.2 illustrates the activity levels of Caesium-137 and Strontium-90 with respect to time. From Fig.2 it can be see that there is a rise and drop in the activity levels of the Caesium-137 and Strontium-90 caused by the ion exchange bed change cycle. In these simulations, an ion exchange bed change occurs every 77 days (i.e. every 11 weeks). When a new ion exchange bed comes online, the activity discharges are low, and as the ion exchange media becomes saturated, activity breaks through the bed and thus produces a rising discharge profile which drops again following the next bed change (shown as the peaks in Fig.2).

TABLE 1. Normal Distribution Hyper-Parameters of SIXEP Model Input Parameters

Parameter ID	***Mean***	***s.t.d.***	***Lower Bound***	***Upper Bound***
1	0.50E+3	0.66E+3	0.01E+2	6.63E+3
2	39.0E+3	37.0E+3	1.00E+3	210E+3
3	1.05E+3	359E+3	0.11E+3	3.00E+3
4	0.03E+3	0.02E+3	0.01E+4	0.13E+4
5	46.0E-6	36.0E-6	3.00E-6	494E-6
6	6.13E-3	1.83E-3	1.14E-3	1.42E-3
7	1.59E-5	1.28E-5	0.25E-5	14.7E-5
8	9.40E-6	1.05E-5	2.50E-7	1.06E-4
9	15.9E+4	7.10E+4	1.90E+4	4.81E+5
10	0.45E+2	0.49E+2	0.2E+1	0.24E+3
11	2.00E+3	0.62E+3	0.73E+3	4.00E+3
12	0.33E+2	0.39E+2	4.00E-2	5.30E+2
13	0.14E+1	0.30E+1	3.00E-2	0.37E+2
14	3.84E-6	1.22E-5	0.40E-12	1.06E-4
15	3.5E-3	2.82E-4	2.74E-3	4.61E-3
16	3.2E-6	3.28E-6	2.56E-7	3.58E-5
17	2.38E-6	2.93E-6	2.50E-11	2.50E-5
18	2.00E+6	2.79E+5	7.03E+5	3.00E+6

It should be noted that the scalar quantities of interest from the output result shown in Fig.2 is the maximum activity of both radionuclides on the final day of an ion exchange bed life, i.e. 616 days. Propagating the input parameter uncertainties via Monte Carlo (MC) simulation through the SIXEP model gives rise to variability in the maximum concentration of Caesium-137 and Strontium-90 at this point. It should be noted that the simulation shown in Fig.2, the SIXEP model has been started from a saved state representing steady state operation with the mean parameter values. When the parameter values are changed using the MC, it will take the model up to a maximum of 8 ion exchange bed change cycles to reach a new steady state, hence the scalar quantities of interest are taken at the end of 8 bed change cycles.

The global sensitivity analysis technique discussed in section 2 is used to identify the contributions of the uncertain parameters to the variability in the outputs of interest. However, as further discussed in section 2, computing the sensitivity indices of an expensive model such as the SIXEP requires numerous model evaluations for robust estimates, which is computationally infeasible. In fact, for this present case study, a single evaluation of the SIXEP model requires approximately 20 minutes for a single deterministic evaluation. Thus, an ANN has been used as a substitute for the SIXEP model to speed up the computation of the sensitivity indices. The use of an ANN as a substitute introduces additional uncertainty in the analysis. Hence, the bootstrap technique discussed in section 3 is incorporated into the global sensitivity analysis for the quantification of the uncertainties introduced by the use of an ANN.

V. Results

V.A. CONSTRUCTION OF ANNs

The ANN used in this paper has been constructed with input/output training sets $D_{train,} = \{(x_k, y_k), k = 1,2,\ldots,N_{train}\}$ where $N_{train} = 1020$. Specifically, the training data set D_{train} have been obtained by assigning uniform distributions to the uncertain parameters of the model given in Table.1. This was done in order to cover the admissible range of variability in the uncertain parameters. This is followed by the use of the Latin Hypercube sampling technique to draw samples out from these distributions, with each sample drawn being used to evaluate the SIXEP model to obtain output training data $y_k = [y_{k1}, y_{k2}]$. The architecture the ANN (i.e. number of hidden layer and, number of nodes) was critical for the network accuracy. In particular, a 3 hidden layer configuration with 18, 7, 2 neurons respectively has been used within each ANN to optimally fit the complicated, nonlinear input/output data. The error back propagation algorithm (see Rumelhart et al., (1986) [12]) has been used to adjust the weight parameters in the ANN. Specifically, 80% of training samples in D_{train} have been used to calibrate the ANN, while the remaining 20% have been used to validate it. To monitor the performance of the ANN, the regression error R^2 has been used. Mathematically, the R^2 is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y}_i)^2} \tag{10}$$

where y^i represents the output of the original model, $\hat{y}_i$ represents the output of the ANN. Regression errors in the range of 0.8-1 are considered acceptable. The R^2 error of the ANN constructed is shown in Table. 2

TABLE 2: Artificial Neural Network Performance Measure

Neural Network Output	Regression Error
Maximum Caesium-137	0.929
Maximum Strontium-90	0.959

V.B. Sensitivity Indices Estimation

Here, the bootstrapped approach explained in section 3.2 is used to estimate the sensitivity indices from the ANN constructed. The result of the Bootstrap Bias Corrected (BBC) estimate of the sensitivity indices are shown in Fig. 5 (a), (b) and 6 (a), (b).

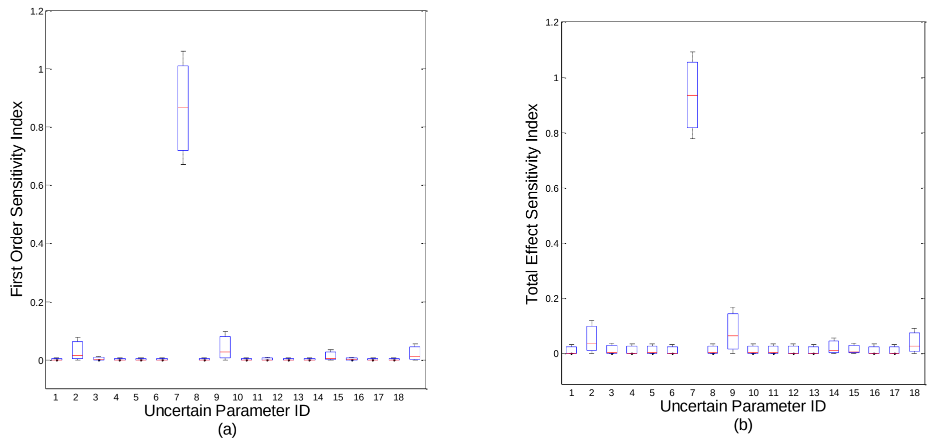

Figure 3. Quantified BBC Sensitivity Indices with respect to Caesium-137

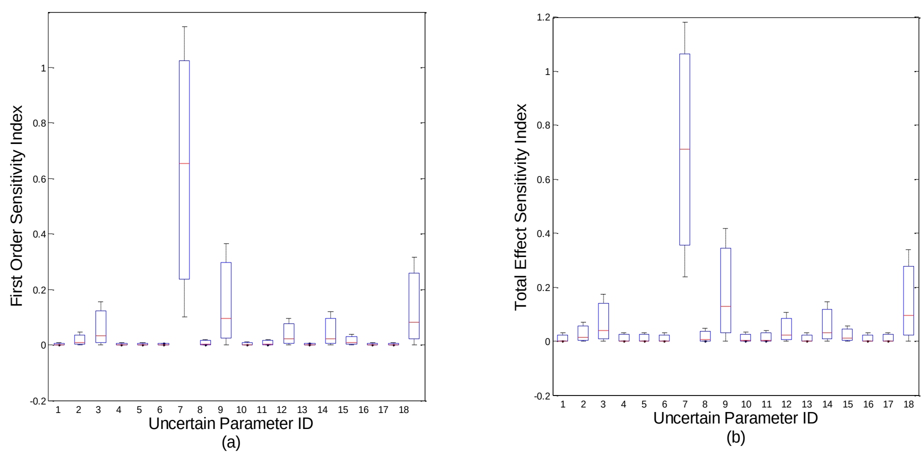

Figure 4: Quantified BBC Sensitivity Indices with respect to Strontium-90

These results have been estimated from 2.00E+6 model evaluations of B = 1000 ANNs. The red line in the figures shows the BBC estimate of the sensitivity indices, which is assumed to be the true estimate if the real model (SIXEP model) was used. The corresponding confidence intervals of the BBC is also reported in the figures. From the results obtained, the 7th parameter has the largest contribution to the variance in the outputs of interest. Also, the width of the confidence interval of the 7th parameter is the largest. This reflects that there is a high uncertainty associated in estimating the parameter. This huge uncertainty could be as a result of the number of bootstrapped ANNs constructed. Hence, future research would investigate how the number of bootstrapped ANNs affects the width of the BBC intervals. It is to the noted that although this analysis is computationally expensive, parallelization strategies have been employed. Specifically, 20 CPU cores have been used to split the jobs amongst the workers on the computer grid, hence, reducing the computational time by 95%.

VI. CONCLUSIONS

In this paper, we have presented a case study concerning a radioactive waste management plant (SIXEP) used for cleaning up radioactive effluents arising from the Sellafield, UK nuclear fuel reprocessing site. Due to the uncertainties affecting the feed composition used as inputs to the plant, the output performance of the plant can be overestimated/underestimated. Therefore, a numerical model of the SIXEP has been developed to understand the response of the plant with respect to different feed compositions. To understand how the model inputs contributes to the variability in the responses of interest, a sensitivity analysis was required. Consequently, due to the high computational cost involved in computing the sensitivity measures directly from the model, ANNs have been constructed and used as substitutes in order to reduce the computational expenditure. However, the use of ANNs adds extra uncertainties to the estimated quantity of interest. Hence, bootstrapped ANNs have been constructed to quantify the uncertainties originating from the use of an ANN. Although the computational cost required using the bootstrap technique is expensive, parallelization strategies have been employed to cut down the computational time. On the basis of the results obtained, the 7th parameter was found to have the greatest contribution to the variance of Caesium-137 and Strontium-90 discharges and was also found to have the highest margin of uncertainty determined using the bootstrap technique. The reason for this large margin of uncertainty in this parameter could be as a result of the number of bootstrap models constructed. Hence, the effect of the number of bootstrap models constructed on the margin of uncertainties would be investigated in the future.

ACKNOWLEDGMENTS

This work has been partially supported by the EPSRC Grant EP/M018717/1 (Smart on-line monitoring for nuclear power plants (SMART)).

REFERENCES

[1] Patelli, E., Panayirci, H. M., Broggi, M., Goller, B., Beaurepaire, P., Pradlwarter, H. J., & Schuëller, G. I. (2012). General purpose software for efficient uncertainty management of large finite element models. *Finite elements in analysis and design*, *51*, 31-48.

[2] Baroni, G., & Tarantola, S. (2014). A General Probabilistic Framework for uncertainty and global sensitivity analysis of deterministic models: A hydrological case study. *Environmental Modelling & Software*, *51*, 26-34.

[3] B. Efron, R.J. Thibshirani. (1993) "An introduction to the bootstrap. Monographs on statistics and applied probability 57", Chapman and Hall, New York.

[4] Saltelli, A., 2002, Making best use of model valuations to compute sensitivity indices, Comput.Phys.Commun. 145, 280–297.

[5] Sobol′, I. M. (2001). Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Mathematics and computers in simulation, 55(1-3), 271-280.

[6] Zio, E. (2006). 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), 1460-1478.

[7] Patelli, E. (2016). COSSAN: A Multidisciplinary Software Suite for Uncertainty Quantification and Risk Management. In Handbook of Uncertainty Quantification, Springer International Publishing 1-69 DOI:10.1007/978-3-319-11259-6_59-1

[8] E. Patelli and M. Broggi and d. A. Marco and M. Beer (2014) OpenCossan: An Efficient Open Tool for Dealing with Epistemic and Aleatory Uncertainties In ASCE Vulnerability, Uncertainty, and Risk 2564-2573

[9] Secchi, P., Zio, E., & Di Maio, F. (2008). Quantifying uncertainties in the estimation of safety parameters by using bootstrapped artificial neural networks. *Annals of Nuclear Energy*, *35*(12), 2338-2350.

[10] Owens, S., Higgins-Bos, M., Bankhead, M., & Austin, J. (2015). Using chemical and process modelling to design, understand and improve an effluent treatment plant. *Also in this issue*, 4.

[11] Helton, J. C., & Burmaster, D. E. (1996). Guest editorial: treatment of aleatory and epistemic uncertainty in performance assessments for complex systems. *Reliability Engineering & System Safety*, *54*(2), 91-94.

[12] Rumelhart, D. E., Hinton, G. E., Williams, R. J. (1986). Learning internal representations by error back-propagation.