ISSN 0951-8320
ELSEVIER
RELIABILITY ENGINEERING & SYSTEM SAFETY
Editor-in-Chief
Carlos Guedes Soares
Available online at www.sciencedirect.com
ScienceDirect

Robust on-line diagnosis tool for the early accident detection in nuclear power plants

Silvia Tolo, Xiange Tian, Nils Bausch, Victor Becerra, T.V. Santhosh, G. Vinod, Edoardo Patelli

PII: S0951-8320(18)30425-3
DOI: https://doi.org/10.1016/j.ress.2019.02.015
Reference: RESS 6383

To appear in: *Reliability Engineering and System Safety*

Received date: 5 April 2018
Revised date: 4 January 2019
Accepted date: 12 February 2019

Highlights

- A robust on-line monitor tools that absorb different sources of uncertainty is proposed
- Based on a set of artificial neural network architectures through the use of Bayesian statistics
- Provides the quantification of the output confidence bounds
- Enhances of the model response accuracy and relax the need for model selection
- Tested on a 220 MWe heavy-water nuclear reactor case-study

Robust on-line diagnosis tool for the early accident detection in nuclear power plants

Silvia Tolo[a], Xiange Tian[b], Nils Bausch[b], Victor Becerra[b], T. V. Santhosh[c], G. Vinod[c], Edoardo Patelli[a,1,]

[a] *Institute for Risk and Uncertainty, University of Liverpool, UK*
[b] *School of Engineering, University of Portsmouth, UK*
[c] *Reactor Safety Division, Bhabha Atomic Research Centre, Mumbai, India*

Abstract

Any loss of coolant accident mitigation strategy is necessarily bound by the promptness of the break detection as well as the accuracy of its diagnosis. The availability of on-line monitoring tools is then crucial for enhancing safety of nuclear facilities. The requirements of robustness and short latency implied by the necessity for fast and effective actions are undermined by the challenges associated with break prediction during transients.

This study presents a novel approach to tackle the challenges associated with the on-line diagnostics of loss of coolant accidents and the limitations of the current state of the art. Based on the combination of a set of artificial neural network architectures through the use of Bayesian statistics, it allows to robustly absorb different sources of uncertainty without requiring their explicit characterization in input. It provides the quantification of the output confidence bounds but also enhances of the model response accuracy. The implemented methodology allows to relax the need for model selection as well as to limit the demand for user-defined analysis parameters. A numerical case-study entailing a 220 MWe heavy-water reactor is analysed in order to test the efficiency of the developed computational tool.

Keywords: LOCA, Neural Networks, Pattern Recognition, Bayesian Statistics, Fault Diagnostics, On-line Condition Monitoring

1. Introduction

The attractiveness of nuclear energy in the market and among the public is generally challenged by its operation and maintenance costs (equal to 73% of the total nuclear power generation cost in contrast with 15% of the cost of electricity generation from fossil sources [1]) and by the numerous concerns regarding the system safety and reliability. Hence, the challenges for the growth of the nuclear sector involve the enhancement of safety, the preservation of availability and the reduction of the costs associated with the operation and maintenance of the plants.

The possibility of pipe breaks resulting in refrigerant loss in the primary cooling system, generally referred as LOCA (Loss of Coolant Accident), is an integral part of the design of pressurized water reactors (PWRs). Indeed, due to the dramatic consequences that this type of incident may have on

[1] Corresponding author

the structural safety of the reactor, the latter must be designed and built with the purpose of withstanding any related accident scenario avoiding the release of radioactive material to the external environment. As for any kind of emergency, the key to the effectiveness of the system response to failure lies primarily with the prompt and accurate diagnosis of the incident occurred. In the specific case of LOCAs, the severity of the break defines the time until the core will be uncovered and thus is an essential information to predict the behaviour of the system during the accident scenario and plan for timely action.

In the case of an accident, this task is allocated to the plant operators: they are required to forecast the progression of the accident through the observation of the abnormal trends of plant parameters displayed in the control room (e.g. inlet and outlet temperature, pressure in hot headers etc.) and their similarity with the typical patterns of specific break sizes and locations. Taking into account even only the time constraint and the hundreds of instrument readings and alarms the operators would be faced with, it is not difficult to deduce how overwhelming such task may result and how much room it leaves to potential errors. On the other hand, LOCA can occur without triggering drastic transients, i.e. small breaks may result in primary pressure remaining high up for few hours, making the detection and diagnosis of such occurrence difficult for the operator and hence putting the system in danger on the long term.

To enhance reactor safety and overcome the limitations and risks associated with the reliance on operators'response, several efforts towards the automatization of fault detection and diagnostics have been made. On-line monitoring techniques are commonly regarded as promising tools for tackling these issues through the continuous supervision of the ongoing processes, behaviour and overall system health. This is achieved through the acquisition of data while the plant is operating (i.e. through the use of sensors) on one hand, and on the adoption of low-cost computational methodologies that ensure real-time computation response on the other.

This latter requirement has progressively led the implementation of such techniques towards the field of machine learning and, more specifically, towards the use of artificial neural networks (ANNs). Indeed, this methodology not only ensures fast response in mapping data, which makes it particularly attractive for on-line monitoring purposes, but has been proven able to satisfactorily capture the non-linear behaviour typical of complex systems [2]. For these reasons, ANNs solutions are not new to the nuclear industry: successful application include the determination of two-phase mixture density [3] and flow regime [4], the prediction of the critical heat flux and heat transfer [5] [6] [7], the calculation of maximum fuel cladding during complex transients [8] [9], neutron transport [10], reactor reactivity [11] and, more generally, the implementation of control systems [12] [13] [14] [15]. Focusing exclusively on the use of ANNs for LOCAs monitoring, the available literature narrows significantly. First attempts date back to the early 90s [16], when ANNs started acquiring popularity exponentially thanks to the re-discovery of the backpropagation method [17]. Beyond the work of Bartlett and Uhrig, that embraced LOCAs detection in a wider-purpose plant transient diagnostics systems, few other studies have focused specifically on tailoring neural network models on LOCAs applications [18] [19].

Although these studies have highlighted the great potential of this approach, the reliance on measured data and the black-box nature of ANNs raise reasonable questions about the impact of the input and model uncertainty on the analysis accuracy and hence on the credibility of the provided point prediction. Indeed, measured data possess an intrinsic level of uncertainty, due to its evolutionary nature (i.e. time-space variability) and unavoidable measurement imprecision or even errors [20]. In the case of real-time monitoring, the degree of uncertainty affecting the prediction is further exacerbated by the restriction to low computational complexity models, which generally comes at the cost of lower accuracy. In light of this, the quantification of the uncertainty in output, and hence the robustness of the tools adopted for the analysis, play a role as crucial as the primary diagnostic analysis itself. In other words, any robust on-line monitoring methodology should characterize the output of interest with margins of uncertainty in order to provide a fully risk-informed decision support.

The present study aims to address the limitations of the current literature, providing a novel solution for enhancing the robustness of the adopted models on the one hand and to quantify the accuracy of the diagnostic response on the other. Differently from existent strategies, the proposed approach allows to take into account several sources of uncertainty in input (e.g. data noise, model uncertainty etc.) without requiring their characterization and to quantify the uncertainty affecting the model in output. This is achieved thanks to the use of a novel methodology that relies on the integration, through Bayesian statistics, of diverse ANN architectures for shaping the model response on the basis of the individual models'credibility along the output domain. Section 2 provides a brief overview of the theoretical background associated with the ANN architectures developed while 3 describes thoroughly the novel methodology proposed for the robustness enhancement and uncertainty quantification of the model. Details regarding the specific computational tools adopted and implemented are provided in Section 4. Finally, the effectiveness of the novel methodology has been verified through the analysis of a real-world case-study focusing on a 220 MWe pressurized heavy water reactor system, discussed in detail in Section 5.

2. Background

Multilayer perceptron (MLP) is the most common class of feedforward artificial neural networks, which create a functional mapping from the input space to the output target space. This is realised adapting the weights connecting the networks nodes, generally referred as neurons, according to a predefined target through the training of the network. The most widely used training approach for adapting the weights is back-propagation which is based on gradient descent techniques. In this study, four different ANN models have been employed. Three of these are MLP models trained adopting the Levenberg-Marquardt algorithm, as it is the fastest method for moderate-sized feedforward ANNs (up to several hundred weights) [21]. The three MLP architectures differ for number of layers (e.g. one and two hidden layer networks are considered) and for number of connections, which were reduced adopting the optimal brain surgeon method, briefly introduced in section 2.0.1. The fourth model adopted falls in the class of group method of data handling, described in Section 2.0.2.

2.0.1. Optimal Brain Surgeon (OBS)

In neural networks, the regularization problem is often cast as minimizing the number of connection weights. When there are too many weights, over-fitting problems can occur which result in poor generalization performance. Conversely, if there are too few weights, the network might not be capable to adequately learn the key aspects of the training data. The optimal architecture should be large enough to learn the problem and small enough to generalize well. When the training set is very limited, it is increasingly important to choose a network architecture wisely in that it should only contain the most essential weights. The optimal architecture selection can also be increasingly difficult, if a limited number of available data sets causes difficulties to set aside a suitable data set for testing purposes. The OBS algorithm proposed by Hassibi et al. [22] is an effective method to optimize the network architecture, allowing to improve the generalization ability of the network and hence its robustness.

The OBS algorithm judges the importance of a weight by its saliency which represents the increase in error that is introduced by eliminating it. Then, the OBS algorithm deletes the weights with the weakest saliency, which means they are insignificant. This is an iterative process which continues until a stopping criterion is satisfied [23], as shown in Figure 1.

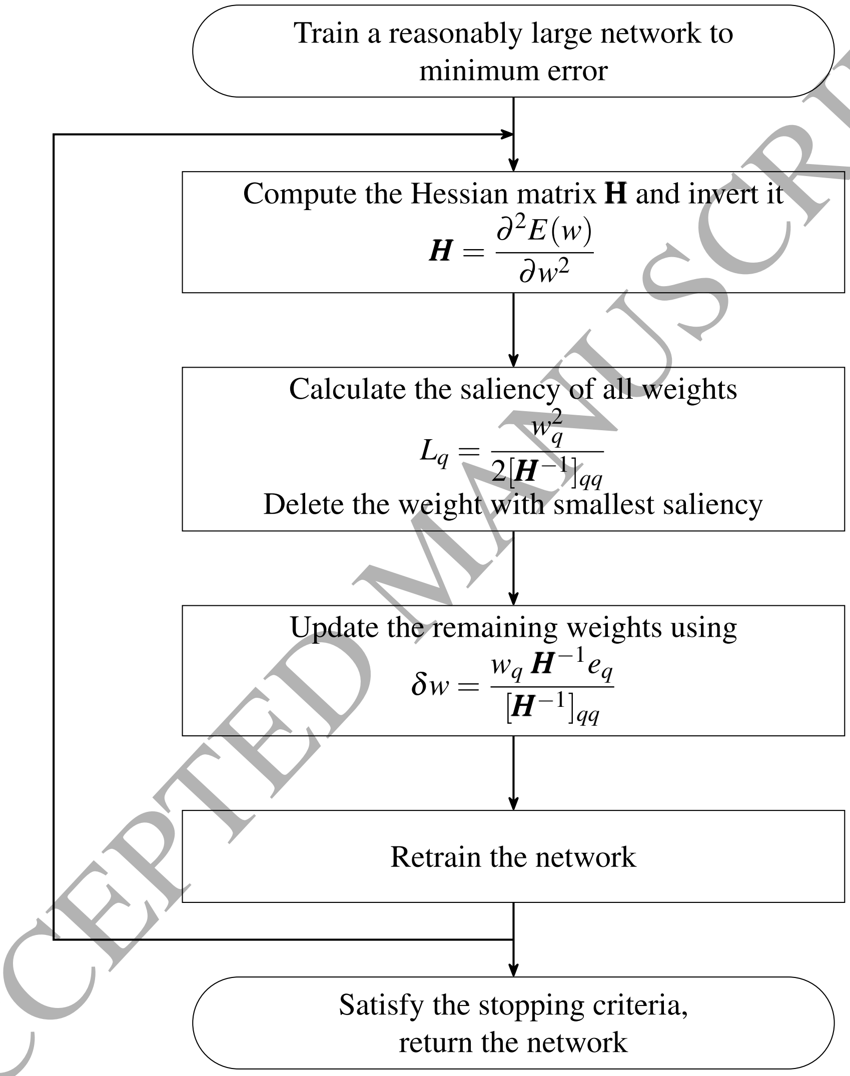
Train a reasonably large network to minimum error
Compute the Hessian matrix **H** and invert it
$\boldsymbol{H} = \frac{\partial^2 E(w)}{\partial w^2}$
Calculate the saliency of all weights
$L_q = \frac{w_q^2}{2[\boldsymbol{H}^{-1}]_{qq}}$
Delete the weight with smallest saliency
Update the remaining weights using
$\delta w = \frac{w_q \boldsymbol{H}^{-1} e_q}{[\boldsymbol{H}^{-1}]_{qq}}$
Retrain the network
Satisfy the stopping criteria, return the network

Figure 1 – The process flow chart of OBS

2.0.2. *Group method of data handling (GMDH)*

GMDH [24] is an approach that involves growing a neural network which has a self-organized structure. There is no need to specify the number of layers and neurons in each layer. The GMDH algorithm replaces the neuron activation functions by the set of hierarchically connected partial models. The coefficients of these models are estimated by the least squares method. GMDH networks gradually increase the number of partial model components and find a model structure with optimal complexity. Thus this approach avoids the tedious work of architecture selection, and allows focusing on the organization of effective model inputs based on physical and statistical considerations. Figure 2 shows the structure of a GMDH network with three inputs and one output. The algorithm is based on a multilayer structure using the general form, which is referred to as the Kolmogorov-Gabor polynomial (Volterra functional series) [25]. The output of a neuron with three inputs shown in Figure 2 can be expressed by a partial polynomial, for example as given in Eq. (1)

$$\begin{aligned} y = a_0 + a_1x_1 + a_2x_2 + a_3x_3 + a_4x_1x_2 + a_5x_1x_3 + a_6x_2x_3 \\ + a_7x_1^2 + a_8x_2^2 + a_9x_3^2 + a_{10}x_1x_2x_3 + a_{11}x_1^3 + a_{12}x_2^3 + a_{13}x_3^3 \ldots \end{aligned} \tag{1}$$

where x_i denotes the ith input, y is the corresponding output, and a_i, a_{ij} and a_{ijk} are the coefficients.

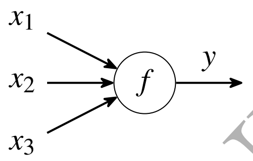

Figure 2 – The structure of a GMDH network with three inputs and one output

The architectures of the neural networks applied in this study are from reference [26]. For the 1-hidden layer MLP, there are 15 neurons in the hidden layer. For the 2-hidden layer MLP, there are 19 neurons in the first hidden layer and 26 neurons in the second hidden layer. There are 586 weights in the fully-connected 1-hidden layer MLP. The minimum number of weights is set to 400, so there are 186 weights pruned at most to guarantee the accuracy of modelling. For the GMDH method, the maximum number of inputs for each neuron is 3, the degree of polynomials is 3, the maximum number of neurons in a layer is 7 and the structure is optimized by the method.

2.1. Interpolation pre-processing

Generally, the training data sets are only available for a limited number of break sizes. This causes robustness issues, as the resulting neural network models are not very accurate for predicting unseen break sizes. With the aim of improving robustness in this study, interpolation was applied to generate data sets for additional break sizes for neural network training. Overall, the criterium for selecting the interpolated sizes is that a newly generated break size can split the interval between two adjacent break sizes equally. If the break sizes are $B_s(1), B_s(2), ..., B_s(n)$, the interpolated break size can be calculated as

$$B_{sn}(i) = [B_s(i-1) + B_s(i+1)]/2, \quad i = 1, 2, ..., n-1 \tag{2}$$

Therefore, the new break size list is $B_s(1)$, $B_{sn}(1)$, $B_s(2)$, $B_{sn}(2), ...,$ $B_s(n-1)$, $B_{sn}(n-1)$, $B_s(n)$.

Two well-known interpolation methods (linear interpolation and cubic spline interpolation) were applied in this study.

3. Proposed approach

The ANN implementation process allows to capture the underlying relationship existing between a set of input factors and an output. This capability is exploited to draw further predictions, which are provided as crisp estimates of the system response. In spite of the large popularity of ANNs in a wide range of applications, the lack of transparency associated with their black-box nature, together with the stochastic character of training process, raises reasonable questions on the credibility of the model output when addressing real-world problems, which are unavoidably affected by inherent variability and uncertainty.

The uncertainty affecting the model prediction can be mainly traced back to the input, network parameters and model structure. Indeed, for the first case, any available dataset will be inevitably affected by measurement and sampling errors to a certain extent, potentially resulting in misleading model outputs when the degree of uncertainty becomes significant. Moreover, the overall accuracy of the diagnosis depends not only on the quality of the available data but also on how much the input differs from the initial dataset the network was trained on. This is due to the data-driven nature of the ANN training process, which allows to identify the parameters of the model on the basis of the information available for calibration. Such mechanism is intrinsically stochastic: due to the random initialization of the network weights in the training process, even the adoption of identical architectures and training strategies would result in ANNs characterized by different parameters and hence performance. In other words, it is not possible to identify an unique set of global optimal parameters for the model, adding further uncertainty on how well this will later approximate the true system. Finally, it is worth to stress out that ANNs only approximate the true process or system under study, implying uncertainty on the model structure due to the simplification and inadequacy of the approximated model.

In order to deal with the above drawbacks, several numerical techniques aiming to characterize the uncertainty affecting the network prediction have been proposed in the literature. Probabilistic based methods, including Bayesian neural networks [27]), require the characterization of the uncertainty affecting the variables involved in the analysis as probability distribution functions. In this case, the credibility of the model output distribution depends on the accuracy of the input distribution and on the number of random samples computed [28]. The first of these requirements implies the capability of accurately capturing the uncertainty in input while the second may hinder the real-time feasibility of the computation due to the problem size. Similarly, fuzzy logic based methods [29] are based on the representation of the uncertain model inputs or parameters as fuzzy numbers to quantify the prediction uncertainty in output. Also in this case, the computational costs result higher than other more traditional approaches and the methodology requires the characterization of the uncertainty in input. Another common category of techniques adopted for ANN uncertainty quantification embraces error prediction methods: these rely on the analysis of the model error in reproducing observed data [30]. Due to the low computational costs, this approach results quite attractive for real-time purposes but lacks any guarantee on the prediction accuracy outside of the training domain. Furthermore, a previous comparison between the error estimation by series association technique and the Bayesian model selection method (which has been demonstrated to be outperformed by the method proposed in the current study [31]) has highlighted the superiority of this latter, at least for the application of interest in the current study [32]. Finally, a further sampling based approach such as bootstrapped techniques [33] rely on the assumption that the bootstrapped samples follow the statistical characteristics of population data and the resamples mimic the random component of the process to be modelled.

All the mentioned strategies require the introduction of several assumptions (e.g. related to the prior distributions of the uncertain input parameters and data to be propagated or to the membership function of the above uncertain quantities), on the validity of which depends the accuracy of the analysis. Moreover, most of the mentioned techniques imply the need for sampling and optimization, increasing the computational costs of the overall analysis which is a crucial aspect for real-time applications.

The current study aims to overcome these limitations, bypassing the need for user-defined input uncertainty and ensuring the computational feasibility of the real-time analysis without affecting the accuracy of the response, as already highlighted in previous studies [31][32]. This is achieved through the combination of multiple ANN models responses on the basis of Bayesian statistics, relaxing the need for model selection. The relevance of this aspect is crucial since different ANNs may outperform each other in different parts of the domain. In this case, the adoption of one model rather than the other, implies a decrease of the model performance in such regions. Furthermore, while it is relatively easy to assess the accuracy of each model over the known dataset (even if the limitations of accuracy metrics adopted in common practice has been often highlighted [34]), the same cannot be said with regards to unexplored data regions.

On the basis of Bayesian statistics, the proposed approach allows to predict automatically the accuracy of each model output on the basis of its behaviour over the training dataset and to predict its reliability outside the known domain. In light of this information, which is obtained avoiding reliance on user's judgements, the models'output are combined in a single response whose accuracy results enhanced compared to the individual ANN contributions. Moreover, it allows to quantify the uncertainty affecting the output according to the specific area of the domain where the output lies. This section is dedicated to provide an overall description of the approach adopted, which is referred to as Adaptive Bayesian Model Selection (ABMS).

3.1. Adaptive Bayesian Model Selection (ABMS)

In order to meet the requirements imposed by the specific application proposed in the current study, both in terms of robustness of the response and computational time, a novel technique based on the Bayesian model averaging approach, has been developed.

ANNs sharing identical architectures and training process result in different models (i.e. ANNs with identical structure but different weights). This is mainly due to the random initialization of the weights in the training process and leads to the implementation of ANNs characterized by different performance and hence output accuracy, introducing a degree of uncertainty in selecting the best performing ANN. Although it is common practice to select the best performing ANN based on cross-validation test (i.e. *k-fold*), this selection strategy does not take into account the eventual noise and imprecision affecting the validation data available and the impossibility to determine the ANN performance when considering unseen data. In order to tackle this issue, a novel methodology for the enhancement of ANN response robustness has been developed to exploit the inner variability of the training process through a Bayesian model averaging technique.

The proposed adaptive Bayesian model selection method computes the posterior probability associated to each network of the ANN set in the form of a continuous distribution over the output domain. Once the continuous posterior distribution has been computed, the robust response as well as the prediction confidence bounds are obtained from the individual networks output adopting as weights posterior probability values according to the distributions previously built,

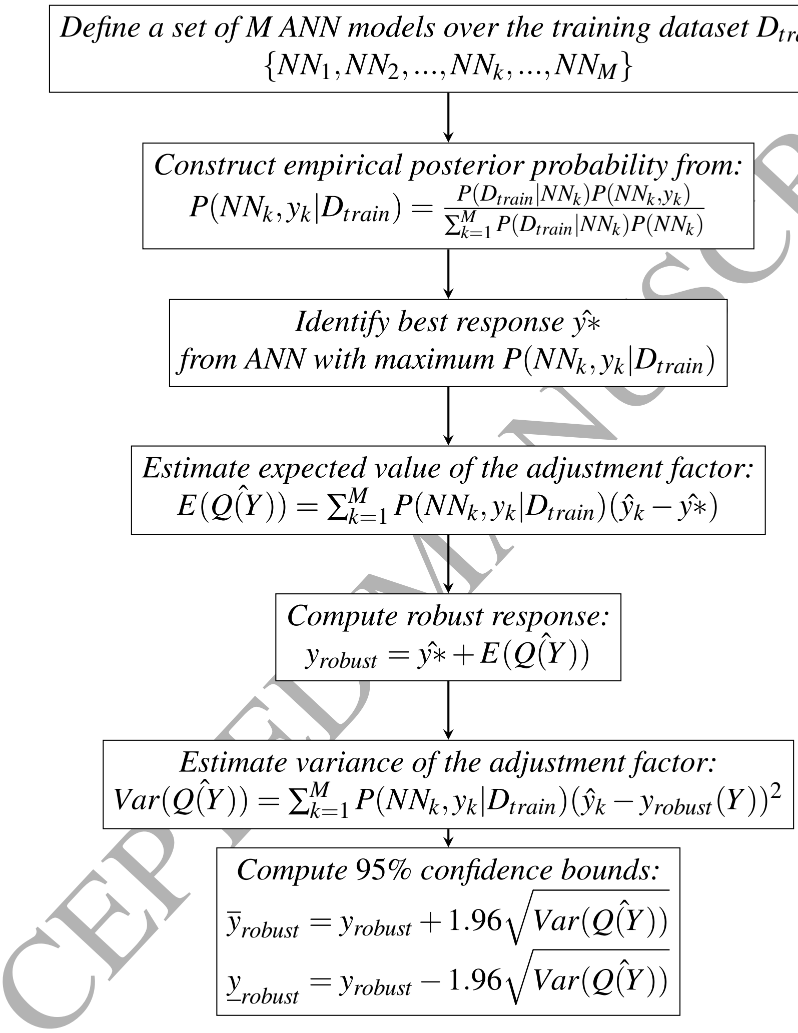
Define a set of M ANN models over the training dataset D_{train}:
$\{NN_1, NN_2, ..., NN_k, ..., NN_M\}$
Construct empirical posterior probability from:
$P(NN_k, y_k | D_{train}) = \frac{P(D_{train}|NN_k)P(NN_k, y_k)}{\sum_{k=1}^{M} P(D_{train}|NN_k)P(NN_k)}$
Identify best response $\hat{y*}$
from ANN with maximum $P(NN_k, y_k | D_{train})$
Estimate expected value of the adjustment factor:
$E(Q(\hat{Y})) = \sum_{k=1}^{M} P(NN_k, y_k | D_{train})(\hat{y}_k - \hat{y*})$
Compute robust response:
$y_{robust} = \hat{y*} + E(Q(\hat{Y}))$
Estimate variance of the adjustment factor:
$Var(Q(\hat{Y})) = \sum_{k=1}^{M} P(NN_k, y_k | D_{train})(\hat{y}_k - y_{robust}(Y))^2$
Compute 95% confidence bounds:
$\overline{y}_{robust} = y_{robust} + 1.96\sqrt{Var(Q(\hat{Y}))}$
$\underline{y}_{robust} = y_{robust} - 1.96\sqrt{Var(Q(\hat{Y}))}$

Figure 3 – Overview of the Adaptive Bayesian Model Averaging approach.

for each network, over the available calibration dataset. Therefore, given a set of identical (i.e. the same model structure) ANNs $\{NN_1, NN_2, ..., NN_k, ..., NN_M\}$ trained with the same data set $D_{\text{train}} = (X,Y) = (x_1,y_1)...(x_i,y_i)...(x_n,y_n)$, Bayes'theorem can be applied to build a continuous posterior probability distribution for the k^{th} ANN in the set which is defined as shown in Eq. 3:

$$P(NN_k, y_k|(X,Y)) = \frac{P((X,Y)|NN_k)P(y_k, NN_k)}{\sum_{q=1}^{M} P(D_{\text{train}}|NN_q)P(NN_q)} \tag{3}$$

where $P((X,Y)|NN_k)$ is the likelihood associated with the NN_k for the given training dataset. For any k^{th} network the true value y_i of the prediction for the i-th sample of the training dataset can be rewritten as $y = \hat{y_{ki}} + \varepsilon_k$, where $\hat{y_{ki}}$ is the i-th output sample provided by the k-th network while ε_k is the associated bias, due to data noise and model error. Assuming the bias for each sample being independent and normally distributed so that $\varepsilon_{ki} = \varepsilon_k - N(0, \sigma_k^2)$, the likelihood in Eq.3 can be built over the calibration domain on the basis of the probability points obtained for each i^{th} data sample as:

$$P((x_i, y_i)|NN_k) = \frac{1}{2\pi\sigma_k^2} \cdot e^{-\frac{(y_i - \hat{y}_{ki})^2}{2\sigma_k^2}} \tag{4}$$

where the variance σ_k^2 for the k-th network is computed from the training data according to the maximum likelihood estimation approach.

On this basis, the posterior probability distribution for each network is built over the response dataset domain and stored. When the model is run in order to provide a prediction of the system state, each output $\hat{y}_{qk}, k = 1...M$ is associated with the approximate value of the probability $P((x_q, \hat{y}_q)|NN_k)$ obtained from the posterior distribution. This shall be understood as a measure of the accuracy of the specific output value for the network of reference. On this basis, the *best output* $\hat{y}*_q$ among those provided by each ANN in the training set, is selected, identifying the maximum value of the posterior probability, so that

$$P((x_q, \hat{y}*_q)|NN*) = max(P((x_q, \hat{y}*_q)|NN_k)), k = 1, ..., M. \tag{5}$$

The main idea behind the implemented approach is that the information provided by each network in the set, and not only the best output, is valuable, but its importance is proportional to its quality. Hence, the robust response, y_{robust}, is computed adjusting the best output on the basis of all the M network outputs but weighting their contribution on the basis of their associated posterior probability, as shown in Eq.6

$$y_{\text{robust}} = \hat{y}*_q + \sum_{k=1}^{M} P(NN_k, \hat{y}_{qk}|(X,Y))(\hat{y}_{qk} - \hat{y}*_q) \tag{6}$$

The variability of the answer obtained from the M networks contains also valuable information that can provide a measure of the uncertainty affecting the output. Indeed, assuming the model uncertainty to follow a normal distribution, the 95% confidence bounds on the robust response, express in terms of upper and lower bounds ($\overline{y_{\text{robust}}}$ and $\underline{y_{\text{robust}}}$ respectively), can finally be computed as:

$$\overline{y_{\text{robust}}} = y_{\text{robust}} + 1.96\sqrt{\sum_{k=1}^{M} P(NN_k, \hat{y}_{qk}|(X,Y))(\hat{y}_{qk} - y_{q\,\text{robust}})^2} \tag{7}$$

$$\underline{y_{\text{robust}}} = y_{\text{robust}} - 1.96\sqrt{\sum_{k=1}^{M} P(NN_k, \hat{y}_{qk}|(X,Y))(\hat{y}_{qk} - y_{q\,\text{robust}})^2} \tag{8}$$

4. Computational Tool: SMARTool toolbox

All the presented methods and the novel approach proposed in the current study have been implemented in an open source MATLAB toolbox named SMARTool. The SMARTool contains procedures and scripts used to identify and train ANN architectures adopting the Machine Learning Toolbox for MATLAB, while the pruning of the one hidden layer architecture has been performed adopting the NNSYSID toolbox [23]. The NNSYSID is a toolbox for the identification of non-linear dynamic systems with artificial neural networks which implements several algorithms for ANN training and pruning.

The SMARTool toolbox provides the adaptive Bayesian model averaging method [31, 35], dataset organization and re-population methods (i.e. for linear and cubic spline interpolation as well as for Gaussian mixture sampling) and dedicated uncertainty quantification techniques, such as the error estimation for series association [32]. In addition, the SMARTool can also access the advanced uncertainty quantification techniques provided by OpenCossan [36]. The methods available in the toolbox include:

- *createSMARTdataset*: this method allows to organize the available experimental data into subsets for calibration, test and validation purposes, according to user's size preferences. Further options provide the enrichment of the available dataset through interpolation (either linear or cubic spline) or sampling techniques adopting Gaussian mixture random models.

- *applyBMS*: this method applies the Bayesian Model Selection technique. This latter can be regarded as a particular case of the ABMS methodology: it relies on the use of crisp posterior probability values instead of continuous distributions for the characterization of the networks performance and hence the definition of the robust model response as well as for the identification of the associated confidence values, which can be specified by the user [37]. The algorithm performs the implementation of the ANN models through the Levenberg-Marquardt training algorithm when provided with calibration dataset, while it directly applies the Adaptive Bayesian Model Selection approach to the data of interest if the ANN set is already available.

- *applyEESA*: this method allows the application of the Error Estimation by Series Association methodology [38] Such approach allows to predict the error associated with the output of a primary ANN through the use of a secondary ANN specifically trained for such task. The tool provides the implementation of the primary and secondary network upon the specification of the model architecture and calibration dataset, as well as the application of existent models to the data of interest. For further details about the implementation of the method and the comparison between the EESA and BMS approach performance, please refer to [31].

- *applyABMS*: this algorithm implements the methodology discussed in Section 2. User defined options include the choice of model for prior and posterior probability distributions (i.e. uniform or empirical and Gaussian mixture or empirical respectively). As for the *applyBMS* method, the algorithm can either provide the training and implementation of the ANNs according to the calibration dataset and model architecture provided or use existent ANN sets.

5. Robust On-line monitoring of a Nuclear Power Plant

The simulation setting was referred from Santhosh et al. [39] and consists of the primary heat transport system of a 220 MWe pressurised heavy water reactor, whose design has a double containment with a vapour suppression pool. The main aim of the containment is to limit the release of radioactivity under normal and accident conditions, both at the ground level and through the stack. The accident scenario contemplated by design is a Loss Of Coolant Accident (LOCA) involving a double ended guillotine rupture of the reactor inlet header. In case of such accident, the vapour suppression pool is designed to limit the peak pressure and temperature in the containment, allowing the complete condensation of the incoming steam and limiting the leakage of fission products to the surrounding environment. In addition to this, several strategies (e.g. dissolving, trapping, entraining mechanisms) are in place to perform the removal of the fission products that reach the pool. All the instrumentation and control parameters are continuously displayed in the reactor control room.
The aim of the analysis is to identify the severity of the LOCA events on the basis of selected instrument signals: for this purpose, the main task of the implemented ANN is to recognize the pattern drawn by the input signals in time and to provide, on the basis of this information, the severity of the break in output. This is quantified in terms of break size, expressed in comparison with the double-ended rupture of the largest pipe in the reactor coolant system. For instance, a 200% break indicates the free discharge of the primary coolant from both the broken ends of the main pipe (this is generally considered the worst accident that can occur in a water circuit).

5.1. Dataset and Models

The proposed approach has been applied to the case study previously described adopting two different sets of ANNs: the first set includes the four specified architectures (i.e. two layers MLP, one layer MLP fully connected, one layer MLP pruned, GMDH network) trained on each defined dataset (i.e. original, with linear and with cubic spline interpolation) for a total set size equal to twelve networks (SET I); conversely, the second set (SET II) includes only four networks, one for each architecture, selected from the previous set on the basis of the model performance in terms of mean square error: for all architectures the best performance was achieved adopting the dataset enriched through cubic spline interpolation, with the only exception of the 2-hidden layer model for which the best results were obtained adopting the linearly interpolated dataset. Both the ANN sets have been tested on two different datasets: the first includes unseen data falling within the training domain (i.e. associated with 20%, 60%, 100%, 120% or 200% break size), the second refers to blind cases where data lie outside the adopted training domain (i.e. break sizes equal to 70%, 50% and 160%).

5.2. Results

The performance of the models implemented has been evaluated in terms of response accuracy as well as confidence interval efficiency. The first is estimated in terms of mean and standard deviation of the squared error (i.e. MSE and STDSE), while the accuracy of the confidence bounds was computed as the overall percentage of experimental data falling within the defined interval. The width of the confidence interval has also been taken into account in the comparison in order to evaluate the degree of precision of the overall analysis. In order to verify the validity of the hypothesis underlying the construction of the confidence bounds for the robust response of the model, goodness of fit tests have been carried out over the test dataset. Both KolmogorovâĂŞSmirnov and Chi-square sets of tests resulted in the acceptance of the Gaussian distribution hypothesis for the models response error, with a confidence of 95% and 90% respectively for all the cases analysed. For both SET I and II, 4 different representations of the prior and posterior probability distributions have been considered and compared, namely:

a Uniform prior and empirical posterior

b Gaussian mixture prior and empirical posterior

c Gaussian mixture prior and Gaussian mixture posterior

d Uniform prior and Gaussian mixture posterior

For all the cases considered, the adoption of different combinations seems not to significantly affect the performance of the models and hence the quality of the prediction.

5.2.1. Test Dataset

All the tests performed over the test dataset resulted in bound accuracy well over 95%. In particular, with regards to SET I, all the output computed fell within the defined confidence bounds, with the only exception of the model adopting uniform distributions as priors and Gaussian mixture as posteriors: nevertheless, also in this case the bounds accuracy is satisfactory, being over 99% (see Table1). As shown in Figure 4, the accuracy of the prediction, as well as the width of the uncertainty intervals in output, is consistent among all the models tested, with a mean value of the square error around 0.0002 and an interval length of about 13 for all the options considered.

ABMS ANN SET	**MSE**	**STDSE**	**Bounds Accuracy**	**Interval Width** [mean (std)]
SET I (a)	0.000218	0.000938	100%	13.0137 (8.9663)
SET I (b)	0.000218	0.000938	100%	13.0137 (8.9667)
SET I (c)	0.000220	0.000926	100%	12.9747 (8.9726)
SET I (d)	0.000227	0.000955	99.79%	13.0935 (8.9834)

Table 1 – Results for the SET I (12 ANNs) over the test dataset.

The use of 4 ANNs instead of 12 slightly decreased the quality of the results, increasing the MSE by over 0.0005 and lowering the bounds accuracy to about 96%. However, this loss of accuracy is at least partly compensated by narrower intervals, as shown in Table 2. As for SET I,

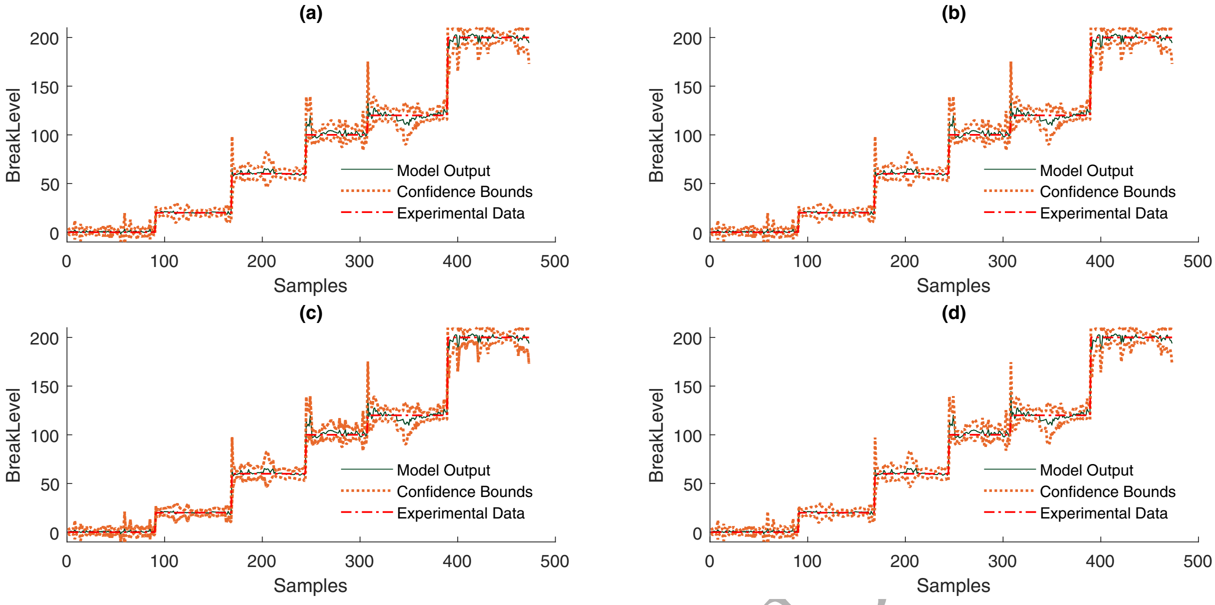

Figure 4 – Model response and computed confidence bounds for SET I (12-ANN set) over the test dataset.

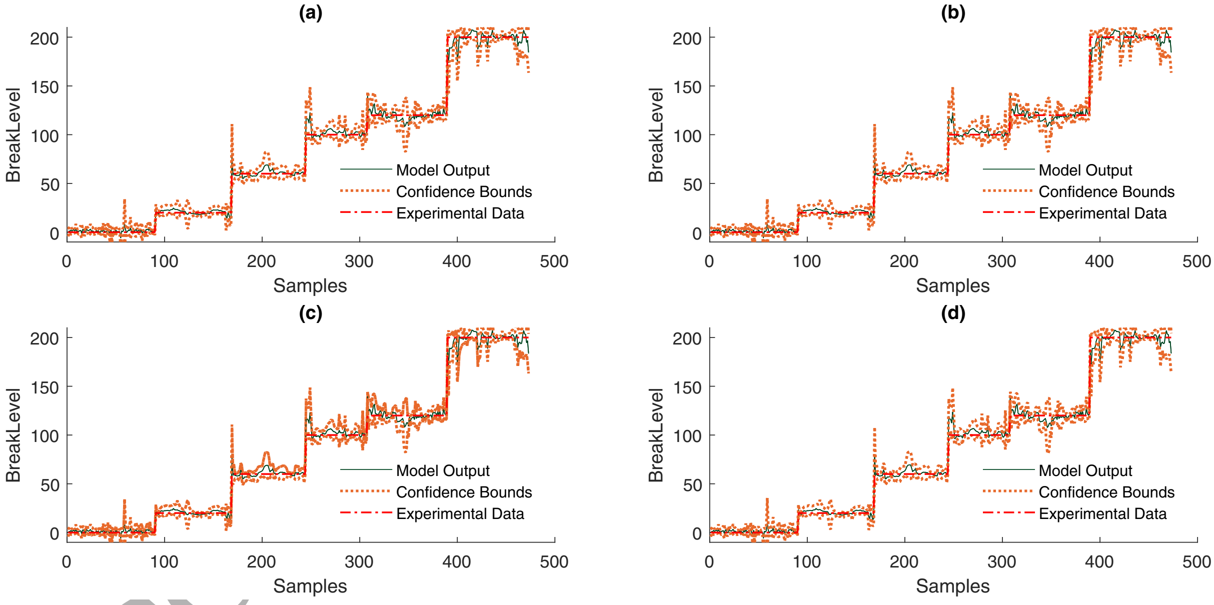

Figure 5 – Model response and computed confidence bounds for SET II (4 ANN set) over the test dataset

the adoption of different probability representations does not significantly affect the overall performance of the models. On the other hand, the quality of the prediction, as well as the associated uncertainty, seems to change within the domain: the trend shown in Fig.4 and 5 suggests a higher performance of the models for small LOCAs and a decrement of the response quality along with the increment of the break size. Furthermore, the output associated with the 200% break size seems to strongly affect the overall error mean value that results instead of an order of magnitude of 10^{-4} for all the lower break sizes. This tendency is common to all the single ANNs included in the sets:

ABMS ANN SET	MSE	STDSE	Bounds Accuracy	Interval Width [mean (std)]
SET II(a)	0.000508	0.001606	96.41%	12.4727 (9.8989)
SET II(b)	0.000508	0.001606	96.41%	12.4729 (9.8992)
SET II(c)	0.000528	0.001729	95.98%	12.4221 (9.8504)
SET II(d)	0.000537	0.001709	96.19%	12.5103 (9.9045)

Table 2 – Results for the SET II over the test dataset

as shown in Fig.6, the models'responses is affected by fluctuations for bigger LOCAs, leading to higher prediction errors.

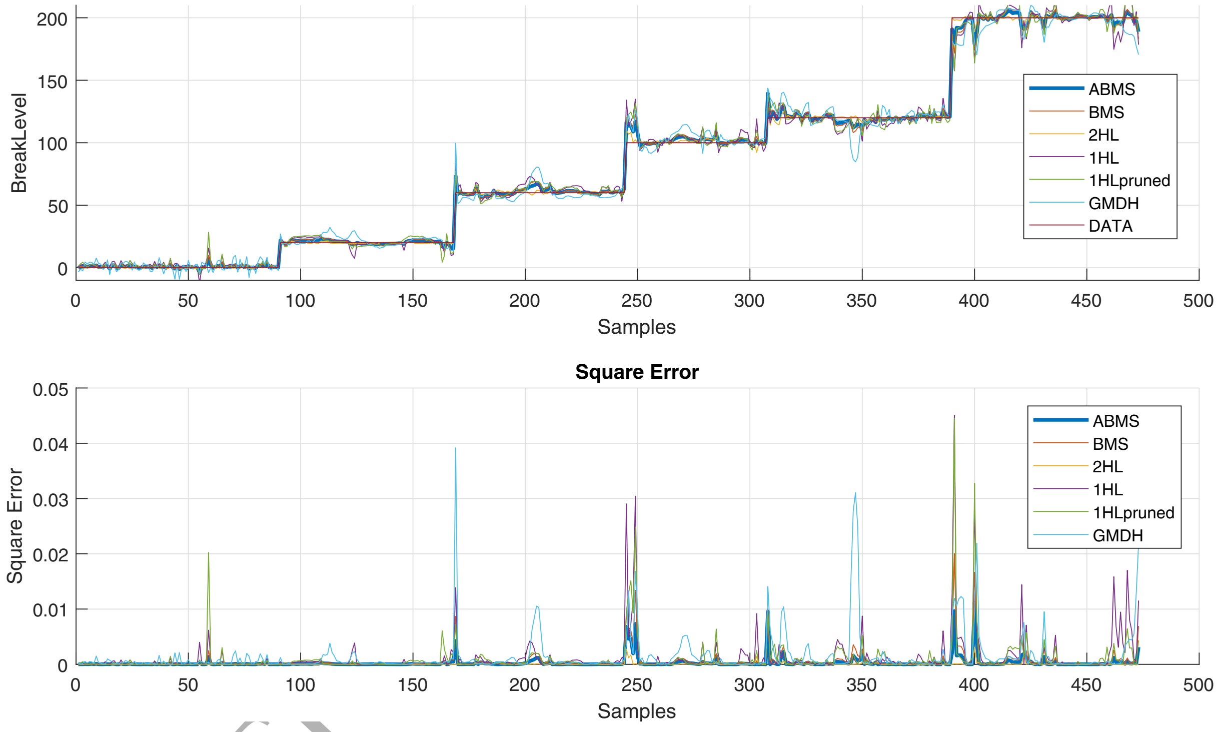

Figure 6 – Comparison between SET I accuracy and the individual models in the set, i.e. two layers MLP (2HL), group method of data handling (GMDH), one layer MLP fully connected (1HL fully connected) and pruned (1HL pruned), over the test dataset

5.2.2. Blind Cases (Validation)

In order to test the generalization capability of the models implemented, both SET I and SET II have been applied to a further unseen dataset including break size values not previously analysed by the networks and hence lying outside the training domain. Also in this case, the models'performance results were satisfactory: as seen in the results for SET I in Table 3, the bounds accuracy is over 95% and the MSE under 0.003 for all the options adopted. As highlighted in Fig.7 and coherently with the previous tests, the choice of distribution results in negligible performance differences.

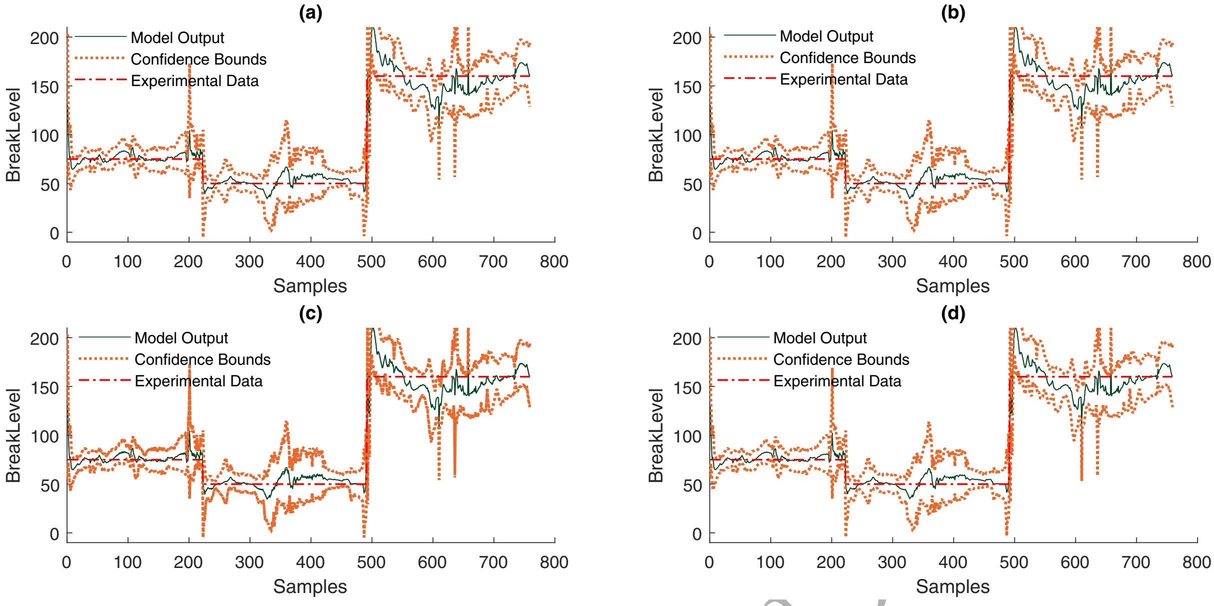

Figure 7 – Model response and computed confidence bounds for SET I (12-ANN set)

ABMS ANN SET	MSE	STDSE	Bounds Accuracy	Interval Width [mean (std)]
SET I(a)	0.002941	0.0084079	96.05%	36.2437 (21.4466)
SET I(b)	0.002941	0.0084079	96.05%	36.2437 (21.4466)
SET I(c)	0.002865	0.0081679	96.18%	35.7369 (20.6085)
SET I(d)	0.002925	0.0083353	96.05%	36.1851 (21.2125)

Table 3 – Results for the ANN sets of Case I

Similar considerations are valid also for the results for SET II (see Fig. 8), shown in Table 4. However, the comparison of the response obtained adopting the two ANN sets highlights the higher accuracy of the confidence interval computed for SET I: for all the possible options considered, the confidence intervals computed for SET II contain the true value of the prediction only about 81% of the times against the 96% registered for the previous case. This can be traced back to the higher variability of the prediction values computed by each network due to the larger number of ANNs in the set and comes at the cost of a larger width of the confidence bounds, which increases by about 60% with regards to the SET II output. On the other hand, the selection of networks performed for the definition of the SET II, ensures a higher accuracy of the model response in this case, lowering the MSE value by almost 30% in comparison to the values obtained by the larger ANN set. The overall best result in terms of response accuracy was obtained by adopting the five-ANN set analysed in SET II and using Gaussian mixture models for posterior probability and uniform distributions as prior. Such best case was hence compared to the single ANN models included in the set (see Fig. 9), highlighting the advantages in terms of overall prediction accuracy as shown in Table 5. Moreover, the response obtained through the combination of the five ANN models resulted in an MSE equal to about 70% of the value obtained with the best performing single ANN (i.e. GMDH) and 57% of the worst performing one (i.e. one hidden layer fully connected).

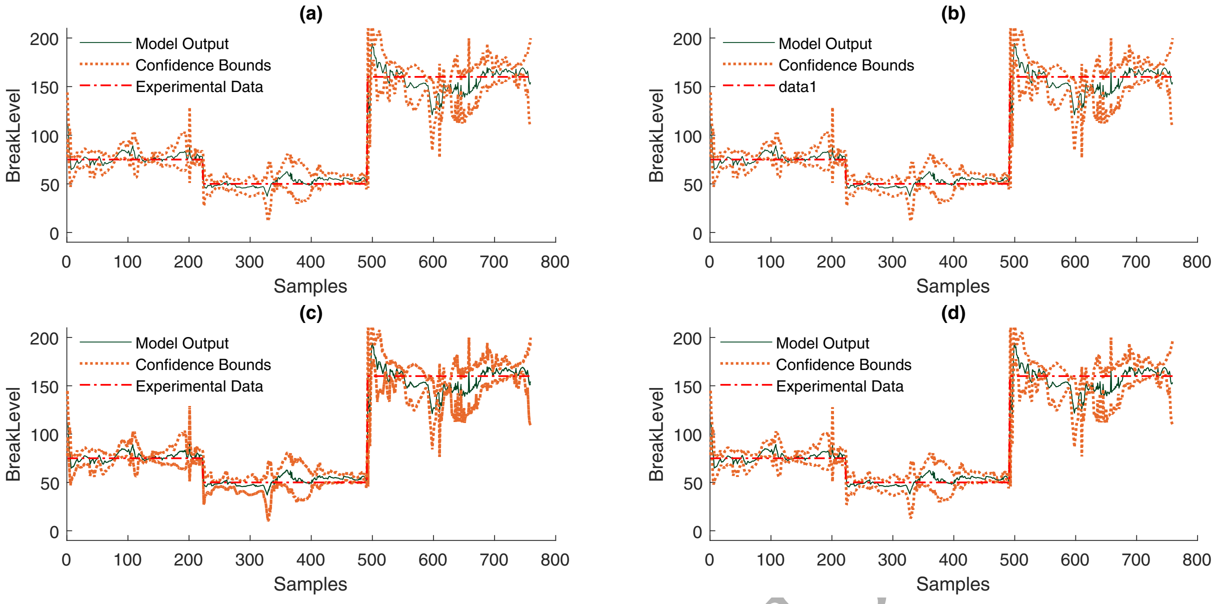

Figure 8 – Model response and computed confidence bounds for the case B (four ANN set)

ABMS ANN SET	MSE	STDSE	Bound Accuracy	Interval Width [mean (std)]
SET II(a)	0.0021509	0.0060499	81.55%	22.2611 (17.0164)
SET II(b)	0.0021507	0.0060490	81.55%	22.2604 (17.0155)
SET II(c)	0.0021550	0.0060395	81.42%	22.1101 (16.5398)
SET II(d)	0.0021261	0.0060162	81.29%	22.1664 (16.8253)

Table 4 – Result for the ANN sets of Case II

Model	MSE	STDSE
ABMS (I c)	0.002126	0.0009
2HL	0.003609	0.0124
1HL fully connected	0.003716	0.0079
1HL pruned	0.003578	0.0083
GMDH	0.002952	0.0094

Table 5 – Response accuracy for the best case in SET II (ABMS Ic) and the individual models in the set, i.e. two layers MLP (2HL), group method of data handling (GMDH), one layer MLP fully connected (1HL fully connected) and pruned (1HL pruned)

Conversely, the adoption of the larger ANN set seems generally to lead to a higher MSE than that obtained with the one layer architectures, but still lower than the other single models analysed.

5.2.3. Discussion

The numerical case-studies analysed demonstrate the suitability of the methodology adopted in terms of response accuracy and reliability of the uncertainty bounds identified. Indeed, even for untrained areas of the output domain (i.e. for the blind cases considered) the true value of the break size falls within the confidence bounds in more than 80% of the cases for the worst-performing

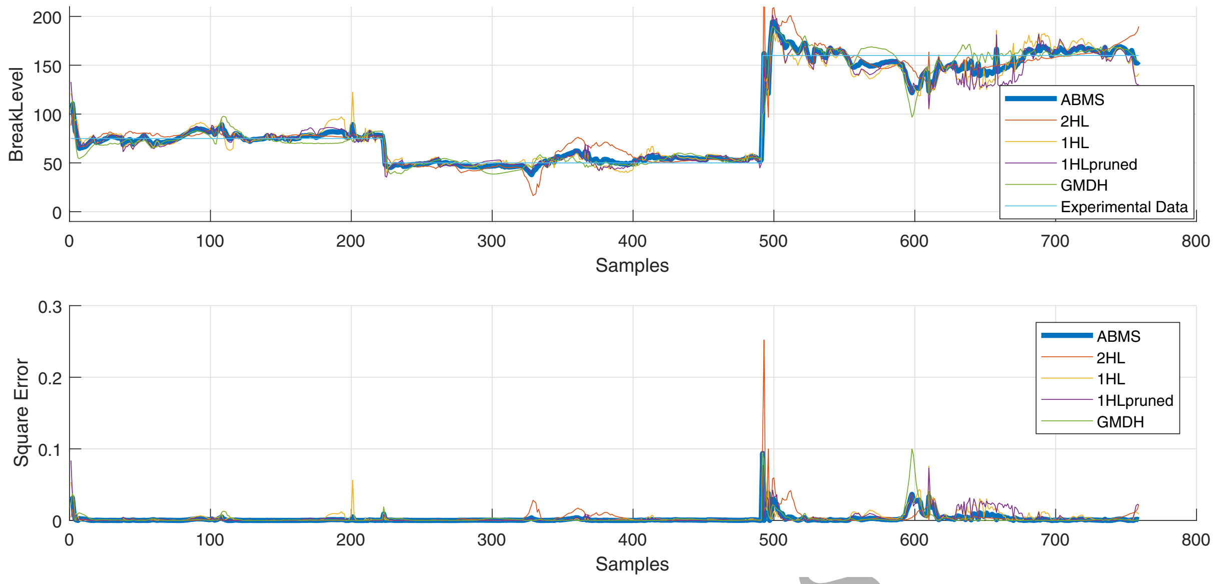

Figure 9 – Comparison between the proposed method (ABMS) accuracy and the single models in the set, i.e. two layers MLP (2HL), group method of data handling (GMDH), one layer MLP fully connected (1HL fully connected) and pruned (1HL pruned)

model. Moreover, in all the cases considered, the accuracy of the overall response results higher than that achieved by the individual ANN architectures separately: this can be interpreted as proof of the efficiency of the method and of the generalization capability of the approach. Furthermore, as anticipated in the introductory sections, the analysis parameters to be defined by the user are extremely limited: in all the settings analysed, the quality of the results appears to be insensitive to the type of representation adopted for both the prior and posterior distributions, providing further proof of the robustness of the methodology. Overall, the definition of the analysis, e.g. in terms of stopping criteria and parameters identification, lies with the design of the single ANN models to be included in the network rather than with their combination according to the proposed technique.
This is true with the exception of the ANN set size which affects the output in terms of response accuracy, confidence bounds effectiveness and confidence interval width. A higher numbers of ANN models in the set ensure lower mean square error values and higher accuracy of the confidence bounds at the cost of wider intervals. Generally, bigger ANN sets are preferable since ensure more accurate responses without significantly increasing the burden of the computation. Nevertheless, when applied to strongly uncertain domain (e.g. far from the support points of the prior distributions), the amount of uncertainty affecting the output may grow significantly leading to not informative response. This drawback is mainly associated with the eventual nature of the application and its high level of uncertainty rather than with the ability of the methodology to depict it. Indeed, this does not apply to the diagnostics of LOCAs since such application is characterized by a finite output domain and the possibility of enriching the training dataset through the use of simulators. On the contrary, the results obtained demonstrate the potential of such technique for LOCAs diagnostics and point to the extension of the current implementation beyond the break size prediction, that is to the inclusion in the analysis of break location detection. However, to extent the application of the proposed methodology to applications characterized by extremely uncertain domains, further research effort are required in order to refine the computation of the output

confidence avoiding non-informative interval width.

6. Conclusions

A novel methodology for the on-line detection and diagnostics of loss of coolant accident has been proposed. The adopted approach relies on the selection of high-performing artificial neural networks and their combination through the use of a Bayesian approach. The methodology adopted is proven to enhance the robustness of the response as well as to reliably quantify the confidence bounds associated with the computed prediction. The main advantages introduced by the proposed solution lie with its capability to relax the need for model selection, limit the reliance on userâĂŹs judgement (in terms of analysis parameters definition), to enhance the response accuracy with regards to the single models adopted and to provide the domain-wise quantification of the output uncertainty.

The methodology was implemented as the diagnostics system of the primary cooling system of a 220 MWe heavy-water pressurized reactor. The numerical application was performed considering two cases: the first includes the use of 12 ANNs sharing 4 architectures (including group method of data handling, fully connected and pruned multilayer perceptron networks) and trained on both experimental and enriched datasets. In the second case, the best performing models among the former networks have been selected, obtaining a set of 4 models. In both cases, the results highlight the suitability of the selected approach as uncertainty quantification technique, with a confidence interval accuracy well over 80% for all the analysed models. The number of the ANNs adopted in the analysis results to be proportional to such accuracy: the use of a higher number of networks results in wider confidence bounds and hence in a higher percentage of success for the uncertainty interval to include the true answer. Nevertheless, this comes at the cost of wider confidence intervals. The novel technique provides better results than any single best-performing network, both in terms of mean and standard deviation of the prediction squared error. The application analysed focuses exclusively on the prediction of break sizes, future work will be dedicated to extent the current diagnostic system to include the break location, which plays a crucial role for the effectiveness of accident mitigation strategies.

The proposed methodology has been implemented and tested in MATLAB and is available in the SMARToolbox, which includes several computational tools for the robust on-line diagnosis of loss of coolant accident in nuclear facilities previously implemented in the context of the SMART project.

Acknowledgement

This work has been supported by the UK Engineering and Physical Sciences Research Council with the project "Smart on-line monitoring for nuclear power plants (SMART)", Grant number EP/M018415/1 and EP/M018709/1.

7. References

References

[1] H. M. Hashemian, On-line monitoring applications in nuclear power plants, Progress in Nuclear Energy 53 (2011) 167–181.

[2] S. G. Vinod, A. Babar, H. Kushwaha, V. V. Raj, Symptom based diagnostic system for nuclear power plant operations using artificial neural networks, Reliability Engineering & System Safety 82 (2003) 33–40.

[3] C. Lombardi, A. Mazzola, Prediction of two-phase mixture density using artificial neural networks, Annals of Nuclear Energy 24 (1997) 1373–1387.

[4] T. Tambouratzis, I. Pàzsit, A general regression artificial neural network for two-phase flow regime identification, Annals of Nuclear Energy 37 (2010) 672–680.

[5] Y. Huang, J. Shan, B. Chen, X. Lang, D. Jia, X. Wang, Application of artificial neural networks in analysis of chf experimental data in round tubes, Nuclear Science and Techniques 15 (2004) 236–242.

[6] J. Shan, J. Zhu, Y. Huang, B. Chen, Application of artificial neural networks in critical heat flux prediction, Nuclear Power Engineering 20 (1999) 182–185.

[7] A. Ridluan, M. Manic, A. Tokuhiro, Ebalm-thp–a neural network thermohydraulic prediction model of advanced nuclear system components, Nuclear engineering and design 239 (2009) 308–319.

[8] F. Cadini, E. Zio, V. Kopustinskas, R. Urbonas, A model based on bootstrapped neural networks for computing the maximum fuel cladding temperature in an rmbk-1500 nuclear reactor accident, Nuclear Engineering and Design 238 (2008) 2165–2172.

[9] J. L. Montes, J. L. François, J. J. Ortiz, C. Martín-del Campo, R. Perusquía, Local power peaking factor estimation in nuclear fuel by artificial neural networks, Annals of nuclear energy 36 (2009) 121–130.

[10] J. J. Ortiz, I. Requena, Using a multi-state recurrent neural network to optimize loading patterns in bwrs, Annals of Nuclear Energy 31 (2004) 789–803.

[11] F. Li, D. Zhang, G. Lu, S. Yang, Reactivity identification during physics start-up of a reactor based on neural network, Nuclear Techniques 30 (2007) 78–80.

[12] M. Boroushaki, M. B. Ghofrani, C. Lucas, M. J. Yazdanpanah, An intelligent nuclear reactor core controller for load following operations, using recurrent neural networks and fuzzy systems, Annals of Nuclear Energy 30 (2003) 63–80.

[13] K. Nabeshima, T. Suzudo, T. Ohno, K. Kudo, Nuclear reactor monitoring with the combination of neural network and expert system, Mathematics and Computers in Simulation 60 (2002) 233–244.

[14] K. Nabeshima, T. Suzudo, K. Suzuki, E. TÜRKCAN, Real-time nuclear power plant monitoring with neural network, Journal of nuclear science and technology 35 (1998) 93–100.

[15] K. Nabeshima, K. Inoue, K. Kudo, K. Suzuki, Nuclear power plant monitoring with recurrent neural network, INTERNATIONAL JOURNAL OF KNOWLEDGE BASED INTELLIGENT ENGINEERING SYSTEMS 4 (2000) 208–212.

[16] E. B. Bartlett, R. E. Uhrig, Nuclear power plant status diagnostics using an artificial neural network, Nuclear Technology 97 (1992) 272–281.

[17] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning representations by back-propagating errors, nature 323 (1986) 533.

[18] T. Santosh, A. Srivastava, V. S. Rao, A. Ghosh, H. Kushwaha, Diagnostic system for identification of accident scenarios in nuclear power plants using artificial neural networks, Reliability Engineering & System Safety 94 (2009) 759–762.

[19] M. G. Na, S. H. Shin, D. W. Jung, S. P. Kim, J. H. Jeong, B. C. Lee, Estimation of break location and size for loss of coolant accidents using neural networks, Nuclear engineering and design 232 (2004) 289–300.

[20] F. Sanchez-Saez, A. Sánchez, J. F. Villanueva, S. Carlos, S. Martorell, Uncertainty analysis of a large break loss of coolant accident in a pressurized water reactor using non-parametric methods, Reliability Engineering & System Safety 174 (2018) 19–28.

[21] L. V. Fausett, Fundamentals of Neural Networks: Architectures, Algorithms and Applications, 1st ed., Prentice Hall International, Inc., 1994.

[22] B. Hassibi, D. G. Stork, G. J. Wolff, Optimal brain surgeon and general network pruning, in: IEEE International Conference on Neural Networks, 1993, pp. 293–299 vol.1. doi:10.1109/ICNN.1993.298572.

[23] M. NâŁŸrgaard, O. Ravn, N. K. Poulsen, NNSYSID-Toolbox for System Identification with Neural Networks, Mathematical and Computer Modelling of Dynamical Systems 8 (2002) 1–20.

[24] M. Witczak, J. Korbicz, M. Mrugalski, R. J. Patton, A GMDH neural network-based approach to robust fault diagnosis: Application to the DAMADICS benchmark problem, Control Engineering Practice 14 (2006) 671–683.

[25] G. Onwubolu, GMDH-Methodology and Implementation in MATLAB, Imperial College Press, 2016. URL: http://www.worldscientific.com/worldscibooks/10.1142/p982. doi:10.1142/p982.

[26] X. Tian, V. Becerra, N. Bausch, G. Vinod, T. Santhosh, A method for measuring the robustness of diagnostic models for predicting the break size during loca, in: Proceedings of the Annual Conference of the Prognostics and Health Management Society 2017 (PHM2017), 2017, pp. 2–10.

[27] D. J. MacKay, A practical bayesian framework for backpropagation networks, Neural computation 4 (1992) 448–472.

[28] K. Kasiviswanathan, K. Sudheer, J. He, Quantification of prediction uncertainty in artificial neural network models, in: Artificial neural network modelling, Springer, 2016, pp. 145–159.

[29] S. Alvisi, M. Franchini, Fuzzy neural networks for water level and discharge forecasting with uncertainty, Environmental Modelling & Software 26 (2011) 523–537.

[30] D. L. Shrestha, D. P. Solomatine, Machine learning approaches for estimation of prediction interval for the model output, Neural Networks 19 (2006) 225–235.

[31] S. Tolo, T. V. Santhosh, G. Vinod, U. Oparaji, E. Patelli, Uncertainty quantification methods for robust break diagnostics in nuclear facilities, in: Proceedings of the 2018 Best Estimate plus Uncertainty International Conference, BEPU2018, 2018.

[32] S. Tolo, T. V. Santhosh, G. Vinod, U. Oparaji, E. Patelli, Uncertainty quantification methods for neural networks pattern recognition, in: Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, 2017, pp. 2715–2722.

[33] 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 (2006) 1460–1478.

[34] D. Alexander, A. Tropsha, D. A. Winkler, Beware of r 2: simple, unambiguous assessment of the prediction accuracy of qsar and qspr models, Journal of chemical information and modeling 55 (2015) 1316–1322.

[35] U. Oparaji, R.-J. Sheu, M. Bankhead, J. Austin, E. Patelli, Robust artificial neural network for reliability and sensitivity analysis of complex non-linear systems, Neural Networks (2017).

[36] E. Patelli, Handbook of Uncertainty Quantification, Springer International Publishing, Cham, 2016, pp. 1–69. doi:10.1007/978-3-319-11259-6_59-1.

[37] U. Oparaji, R.-J. Sheu, E. Patelli, Robust artificial neural network for reliability analysis, in: Proceedings of the 2nd International Conference on Uncertainty Quantification in Computational Sciences and Engineering, UNCECOMP 2017, 2017.

[38] K. Kim, E. B. Bartlett, Nuclear power plant fault diagnosis using neural networks with error estimation by series association, IEEE Transactions on Nuclear Science 43 (1996) 2373–2388.

[39] T. V. Santhosh, A. Srivastava, V. V. S. Sanyasi Rao, A. K. Ghosh, H. S. Kushwaha, Diagnostic system for identification of accident scenarios in nuclear power plants, Reliability Engineering and System Safety (2009) 759–762.