Nuclear Engineering and Technology 53 (2021) 3275–3285

Contents lists available at ScienceDirect

Nuclear Engineering and Technology

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

Original Article

Artificial neural network for predicting nuclear power plant dynamic behaviors

M. El-Sefy [a, *], A. Yosri [b], W. El-Dakhakhni [c], S. Nagasaki [d], L. Wiebe [e]

[a] Department of Civil Engineering, NSERC-CREATE Program on Canadian Nuclear Energy Infrastructure Resilience Under Systemic Risk, McMaster University, Hamilton, ON, L8S 4L7, Canada
[b] Department of Civil Engineering, Institute for Multi-hazard Systemic Risk Studies (INTERFACE), McMaster University, Hamilton, ON, L8S 4L7, Canada
[c] Department of Civil Engineering, and Director, NSERC-CaNRisk-CREATE Program and teh INTERFACE Institute, McMaster University, Hamilton, ON, L8S 4L7, Canada
[d] Department of Engineering Physics, McMaster University, Hamilton, ON, L8S 4L7, Canada
[e] Department of Civil Engineering, NSERC-CREATE Program on Canadian Nuclear Energy Infrastructure Resilience Under Systemic Risk, McMaster University, Hamilton, ON, L8S 4L7, Canada

ARTICLE INFO

Article history:
Received 14 November 2020
Received in revised form
25 April 2021
Accepted 4 May 2021
Available online 20 May 2021

Keywords:
Data-driven models
Artificial intelligence
Artificial neural network
Nuclear power plant
Back-propagation algorithm

ABSTRACT

A Nuclear Power Plant (NPP) is a complex dynamic system-of-systems with highly nonlinear behaviors. In order to control the plant operation under both normal and abnormal conditions, the different systems in NPPs (e.g., the reactor core components, primary and secondary coolant systems) are usually monitored continuously, resulting in very large amounts of data. This situation makes it possible to integrate relevant qualitative and quantitative knowledge with artificial intelligence techniques to provide faster and more accurate behavior predictions, leading to more rapid decisions, based on actual NPP operation data. Data-driven models (DDM) rely on artificial intelligence to learn autonomously based on patterns in data, and they represent alternatives to physics-based models that typically require significant computational resources and might not fully represent the actual operation conditions of an NPP. In this study, a feed-forward backpropagation artificial neural network (ANN) model was trained to simulate the interaction between the reactor core and the primary and secondary coolant systems in a pressurized water reactor. The transients used for model training included perturbations in reactivity, steam valve coefficient, reactor core inlet temperature, and steam generator inlet temperature. Uncertainties of the plant physical parameters and operating conditions were also incorporated in these transients. Eight training functions were adopted during the training stage to develop the most efficient network. The developed ANN model predictions were subsequently tested successfully considering different new transients. Overall, through prompt prediction of NPP behavior under different transients, the study aims at demonstrating the potential of artificial intelligence to empower rapid emergency response planning and risk mitigation strategies.

1. Introduction

A nuclear power plant (NPP) is a complex system-of-systems that contains highly dynamic, interconnected, and interdependent subsystems such as the reactor core and primary and secondary coolant systems. Each of these subsystems consists of multiple critical components where malfunction of any has the potential to initiate an accident that can propagate throughout the entire plant causing serious negative consequences. NPP safety and performance are key concerns during the plant's service life that require a sufficient understanding of the plant's nonlinear dynamic behavior to control the reactor power, cool the fuel, and contain the radioactivity. Lessons learnt from the Fukushima Daiichi nuclear accident showed that the monitoring systems used were ineffective, leading to poor pre-accident operation and management [1]. Early warning systems are therefore essential as, in addition to monitoring, they also include analysis abilities to accurately predict the nonlinear dynamic behavior of components, subsystems, as well as the entire system under normal and transient conditions

* Corresponding author.
E-mail addresses: elsefym@mcmaster.ca (M. El-Sefy), ahmeda69@mcmaster.ca (A. Yosri), eldak@mcmaster.ca (W. El-Dakhakhni), nagasas@mcmaster.ca (S. Nagasaki), wiebel@mcmaster.ca (L. Wiebe).

https://doi.org/10.1016/j.net.2021.05.003

[2]. A rapid early warning system can contribute to effective risk mitigation strategies, based on complex considerations, to serve as a valuable decision support system (DSS) for plant operators [3,4].

Decision-making within a complex dynamic environment can be challenging, especially for highly interdependent and interconnected systems such as NPPs. As such, aside from physics-based models, each NPP must adopt an intelligent and adaptive plant-specific DSS to ensure the safety of the plant, environment, and public. An intelligent DSS aims to collect, organize, and analyze large amounts of data such that decision-makers can take informed actions during emergency situations [5]. Intuitive interpretation of such data is typically challenging, and more sophisticated tools are thus necessary to predict the system response under different operating conditions. Artificial intelligence (AI) provides faster and more effective tools that can learn autonomously based on patterns in data, and therefore have the potential to predict the behavior of complex systems through intelligent DSSs [6,7].

Several data-driven models (DDMs) have been developed based on AI, and have been rapidly progressing over the past few years due to the complex nature of real-world systems, the flourishing of database management [8], and the continuous development of powerful machine learning algorithms [9]. DDMs utilize available data for a specific system operation to obtain mathematical relationships between the system state variables (i.e., inputs and outputs), albeit with limited knowledge of the physical/mathematical interdependence between such variables [10]. *Learning from data* is the main feature of DDMs, where the mathematical interdependence between the system inputs and outputs is discovered iteratively through minimizing the deviation between observed and estimated values [11]. Additionally, DDMs can be applied to gain valuable insights from the system state variables in an unsupervised fashion (i.e., cluster analysis). DDMs provide a different concept to analyze challenging problems in science and engineering [12], and have been widely applied to simulate the dynamic behavior of different complex systems (e.g., transportation, finance, management, climate, medicine, and environment) [10,13–16]. However, the application of AI-based DDMs in the field of nuclear engineering is still limited and has been identified only recently as a critically important research area [17].

Several DDMs have been developed over the past decades, including regression models, artificial neural networks, and cluster analysis [e.g., 18–21], of which the artificial neural network (**ANN**) shows superior efficiency in uncovering complex relationships between system inputs and outputs [22,23]. In this respect, several types of artificial neural network have been developed to date. Examples include feed-forward backpropagation ANN [24], convolutional neural network (CNN) [25], and recurrent neural network (RNN) [26] with its different types including long-short-term memory (LSTM) networks. The feed-forward backpropagation ANN is the simplest type of ANN and is typically used when information is moving in one direction (i.e., inputs are used to predict a set of outputs) [27]. CNN uses different topology to mainly analyze visual datasets. RNN (and LSTM) is typically used when the order dependence of a sequence is of interest. It should be emphasized that CNN- and RNN-based models have more complex structure than that of the feed-forward backpropagation ANN, and are therefore suggested when the latter cannot capture the behavior of the underlying system. As such, the feed-forward backpropagation ANN is employed in the present study.

Training is the first stage in developing an ANN, and several training algorithms have been developed, such as gradient descent algorithms (GD), conjugate gradient algorithms (CG), and quasi-Newton algorithms (QN) [e.g., 28–32]. Training an ANN is a challenging step because: *i*) a proper combination of learning, transfer, and training functions is usually needed [33]; *ii*) different training algorithms result in different accuracy levels [34]; *iii*) training performance depends on the range and amount of data employed [35]; and, *iv*) overfitting may diminish model generalizability [36]. Once trained with enough and representative data, an ANN can be used to predict the system response under new input values. ANNs have been previously employed within the field of nuclear engineering to predict NPP response under multiple core power inputs and loss of flow accidents [17], to simulate the intermediate heat exchanger of a nuclear reactor [20], to develop a plant-wide management plan (i.e., transient identification, plant-wide monitoring, analysis of vibrations, monitoring of performance and efficiency) [37], and to model the thermal dynamic behavior of a NPP [38].

Several previous studies [39–41] have utilized the fuel temperature to estimate the probability of core damage. In addition, fuel temperature and steam pressure are the primary controllers of the reactor core and secondary coolant systems' integrity. Thus, having a model to estimate the temporal fuel temperature and steam pressure is key for effective early warning. In this respect, the present study aims at developing an intelligent DSS based on DDM to predict the critical state variables in a pressurized water reactor (PWR). This includes predicting the reactor fuel temperature and steam pressure in the PWR under four different transients (a change of reactivity, a change of steam valve coefficient, a change of reactor core inlet temperature, and a change of steam generator inlet temperature) considering uncertainty in physical parameters and system operating conditions. Each of the considered transients is represented through eight different severity levels. A feed-forward backpropagation ANN is developed based on data obtained from a previously developed system dynamics (SD) model of a PWR, including uncertainties in the physical parameters and plant operating conditions. Three algorithms (GD, CG, QN), represented by eight training functions, are tested during the training stage, and the best function is identified based on the network performance. The developed ANN model serves as a rapid early warning system and intelligent DSS that can enable the development of quick and proper risk mitigation strategies under challenging and dynamically changing operating conditions.

2. Dataset

A SD model has been previously developed by El-sefy et al. [42] to simulate the nonlinear dynamic behavior of a PWR. The developed SD model represents a single loop reactor in which the feedback mechanisms between the reactor core, the secondary coolant system, the primary coolant system, and the plenums are simulated based on the mathematical descriptions employed in previous studies [43–48]. The nominal values of PWR system parameters are adopted from those of the Palo Verde NPP. Uncertainties associated with the system physical parameters (e.g., the specific heat of primary coolant, heat transfer coefficient from fuel to coolant, coolant temperature coefficient of reactivity) and operating conditions (e.g., primary coolant mass flow rate of inside the core, steam flow rate) were also considered in the SD model. Such uncertainties were represented in terms of probability distributions similar to those employed in previous studies [49–55]. A normal distribution was assumed for 26 parameters, while the other seven parameters were assumed to follow uniform distributions. A schematic diagram of the PWR generating unit, including the feedback loops between the SD representations of the reactor core and secondary coolant system, is shown in Fig. 1. The reader is referred to El-Sefy et al. [42,56] for detailed descriptions of the SD model employed in the present study.

In order to develop an ANN-based predictive model, input-output observation pairs are required. Such information can be obtained from a monitoring system at the plant site or from a

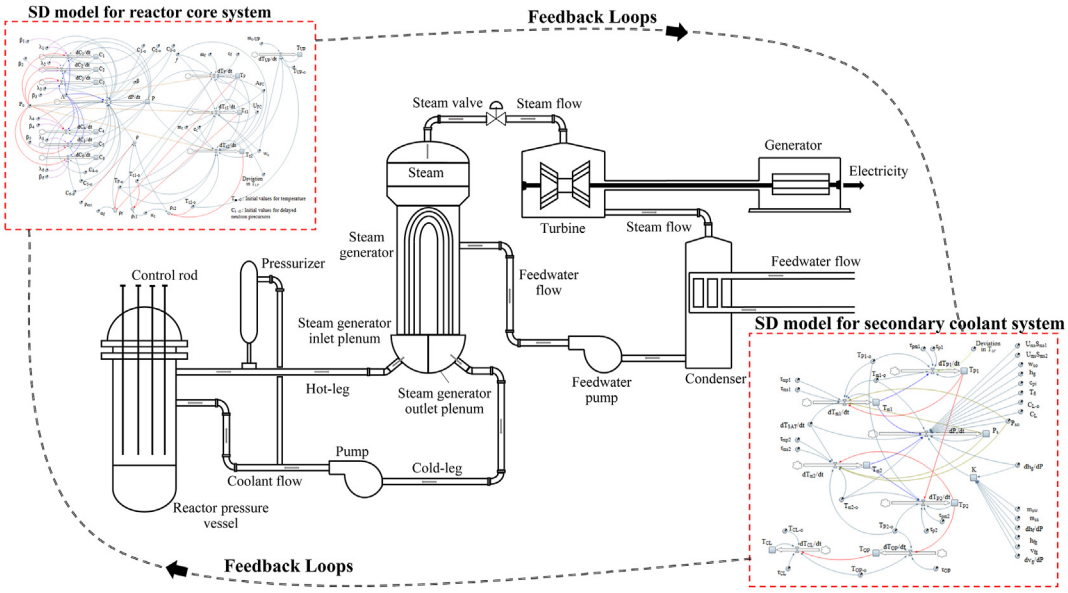

Fig. 1. Schematic diagram of a typical PWR [41].

validated physics- or data-driven model. As such, a previously developed and validated SD model [42] is utilized herein to generate the necessary data to train, validate and test the ANN model due to the lack of actual observations. The developed SD model was utilized to obtain synthetic, dynamic data corresponding to different transients. Four transients were considered, each of which was represented by eight different levels of severity, as summarized in Table 1. A total of 32 different transients were simulated, and the corresponding SD model outputs (i.e., reactor core reactivity (ρ), reactor core thermal power (P_{th}), reactor core inlet temperature (T_{IP}), steam generator inlet temperature (T_{LP}), steam valve coefficient (c_l), fuel temperature (T_f), and steam pressure (P_s)) were monitored and recorded continuously. The SD model was embedded within a Monte-Carlo framework, where 5000 realizations were employed for each transient in order to consider the impact of uncertain physical parameters and operating conditions. The outputs of the SD model were subsequently employed for the development of a corresponding ANN to investigate if the latter can reproduce the same results, albeit faster as a result of no longer needing to analyze the complex physics-based interactions within the NPP systems.

The reactor core reactivity, reactor core thermal power, reactor core inlet temperature, steam generator inlet temperature, and steam valve coefficient were selected as the ANN inputs, while the fuel temperature T_f and the steam pressure P_s were selected as the outputs because T_f provides an indication for the probability of core damage and P_s controls the secondary coolant system integrity. Each of the ANN inputs/outputs was represented by a time series over a time frame of 80 seconds. Eleven realizations corresponding to the minimum and maximum values of T_f and P_s together with those corresponding to each decile (i.e., 10th to 90th percentile range) were used to represent each transient, and were subsequently employed for the development of the ANN. Fig. 2 shows a portion of the SD dataset utilized for the development of the ANN. A total of 57,024 (32 Transients x 11 realizations x 2 outputs x 81 time-steps) samples were thus adopted for the development of the ANN using the neural network (*NN*) toolbox in MATLAB [57]. The total number of samples was randomly divided into 70% training (representing 39,916 samples), 15% validation (representing 8554 samples), and 15% testing (representing 8554 samples) subsets. The training subset was used to build the ANN through adjusting the network parameters. The validation subset was used within the training process to prevent overfitting, while the testing subset was utilized to test the trained network performance under data not employed for training [58]. It is important to note that, the ANN aims to capture the nonlinear relationship between the input and output data based on the randomly selected training data regardless of the size of the time series data. A long event with long time series data can be treated similar to a small event by randomly selecting the training, testing and validation subsets. In addition, long time series data provides larger training data points, which enhances the ability of the ANN-based model to capture more patterns in the dataset.

3. Artificial neural network

3.1. Network architecture

ANN is one of the most popular DDM tools that depends on the concept of learning to replicate the behavior of complex dynamic systems [59]. ANN was inspired by biological neural systems (e.g., the human brain) that can learn to perform tasks through exposure

Table 1
Different transients employed in the present study.

Transient description	Max.	Min.	Increment
Transient 1: Changing the reactivity	−0.006	+0.006	0.0015
Transient 2: Changing the steam generator-steam valve coefficient	−20%	+20%	5%
Transient 3: Changing the reactor core inlet temperature	−20 °F	+20 °F	5 °F
Transient 4: Changing the steam generator inlet temperature	−20 °F	+20 °F	5 °F

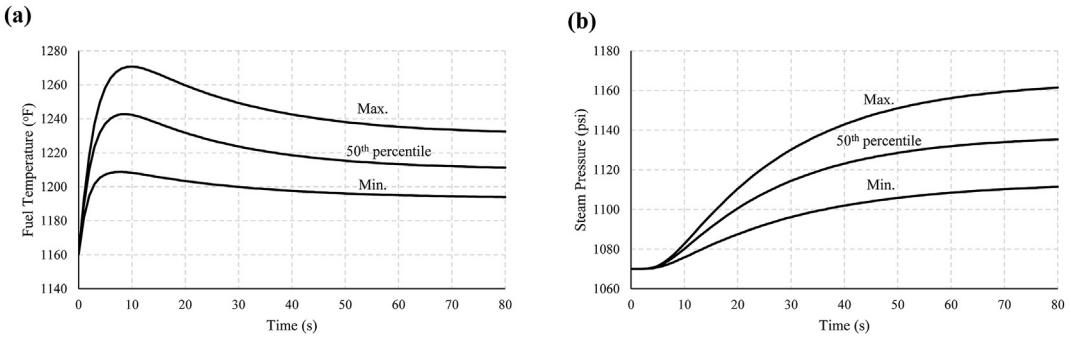

Fig. 2. (a) Sample of SD estimates of Tf due to reactivity transient of +0.0015, and (b) sample of SD estimates of Ps at the same transient level.

to different examples without being constrained to task-specific rules. Therefore, ANNs present an alternative to complex mathematical/physics models without prior knowledge of the underlying processes. In addition, ANNs typically show excellent prediction capabilities when appropriately trained. Due to their capability to simulate nonlinear behaviors, approximate input-output relationships, and recognize patterns within a reasonable amount of time, ANNs have been exhibiting an explosion of application to different research areas [60].

The feed-forward ANN (hereafter referred to simply as ANN) was employed in the present study, and a detailed description of it is provided herein. An ANN typically consists of three main components: the input layer, the hidden layer, and the output layer (Fig. 3). The input and output layers consist of a group of nodes, each of which corresponds to an input or output. The hidden layer contains a collection of artificial neurons that are interconnected to the input nodes. The links between input nodes and hidden layer neurons represent the flow of data between the two layers, where weights are assigned to represent the amount of information shared. A hidden layer is followed by an activation function (e.g., step function, ramp function, or sigmoid function) in order to limit the amplitude of a neuron output [61]. Bias is also introduced when the activation function is applied such that the hidden layer output matches the actual output. In general, in an ANN, biased weighted inputs are passed to an activation function to capture the behavior of complex systems [59]. The ANN output(s) can be represented mathematically as:

$$O = f(\mathrm{XW} + b) \tag{1}$$

where $\boldsymbol{O}$, $\boldsymbol{X}$, $\boldsymbol{W}$, and $\boldsymbol{b}$ represent the outputs, inputs, weights, and bias in matrix notation, respectively. The function f in Equation [1] represents the activation function, where the sigmoid function was utilized in the present study as follows:

$$f(z) = \frac{1}{1 + e^{-z}} \tag{2}$$

where z is an arbitrary input variable. The weights and bias (i.e., $\boldsymbol{W}$ and $\boldsymbol{b}$) are adjusted iteratively through a backpropagation algorithm such that the network output $\boldsymbol{O}$ matches the actual output ($\boldsymbol{P}$). This is referred to as the training process, as the ANN parameters (i.e., $\boldsymbol{W}$ and $\boldsymbol{b}$) are adjusted to fit the relationships inherited within the data. In this study, a backpropagation algorithm was applied using different functions (referred to as training functions), and the mean squared error (***MSE***) was utilized to evaluate the performance of corresponding networks. It is noteworthy that a more complex ANN with multiple hidden layers may be used when highly nonlinear behavior cannot be captured through simple network architecture (e.g., an ANN with a single hidden layer). A mathematical formula, similar to Equation [1], can then be applied to estimate the network output (i.e., $\boldsymbol{O}$).

One of the key challenges in the design of ANN is determining the number of hidden layers and their neurons. Increasing the number of hidden layers can significantly increase the computational time required for both model training and usage. However, based on previous studies [62,63], the use of a single hidden layer has been proven to be efficient in approximating continuous nonlinear functions according to the Universal Approximation theorem. Although multiple hidden layers might be essential when the underlying system behavior is highly complex and cannot be captured through a single hidden layer, increasing the number of hidden layers may cause overfitting problems, which limits the generalizability of the corresponding ANN predictions. In this respect, the present study employed an ANN with a single hidden layer containing n_h neurons (Fig. 4) to predict the nonlinear response of a PWR. Identifying the number of hidden layer neurons is crucial as small n_h values disable the neural network from capturing the relationship between the inputs and outputs (i.e., the model can neither fit the training data nor be generalized) whereas large n_h values may lead to the problem of overfitting [64]. Overfitting occurs when the results from an ANN cannot be generalized (i.e., the ANN's accuracy is constrained to predicting the training data only) [36]. In the present study, n_h is determined based on the number of inputs, n_i, according to Kalmogorov's theorem [65]:

$$n_h \leq 2n_i + 1 \tag{3}$$

3.2. Training algorithms

As mentioned before, different functions may be used for training an ANN. In general, the most appropriate function is that

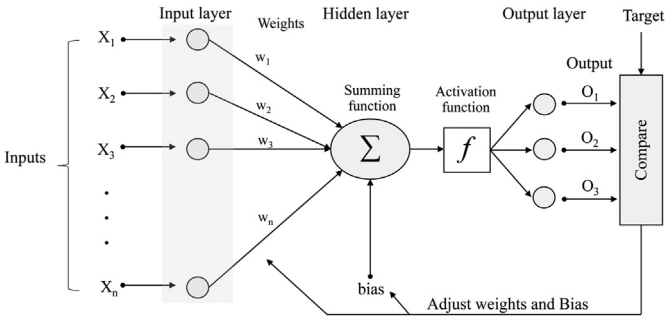

Fig. 3. Schematic diagram of the feed forward back propagation neural network with single hidden layer.

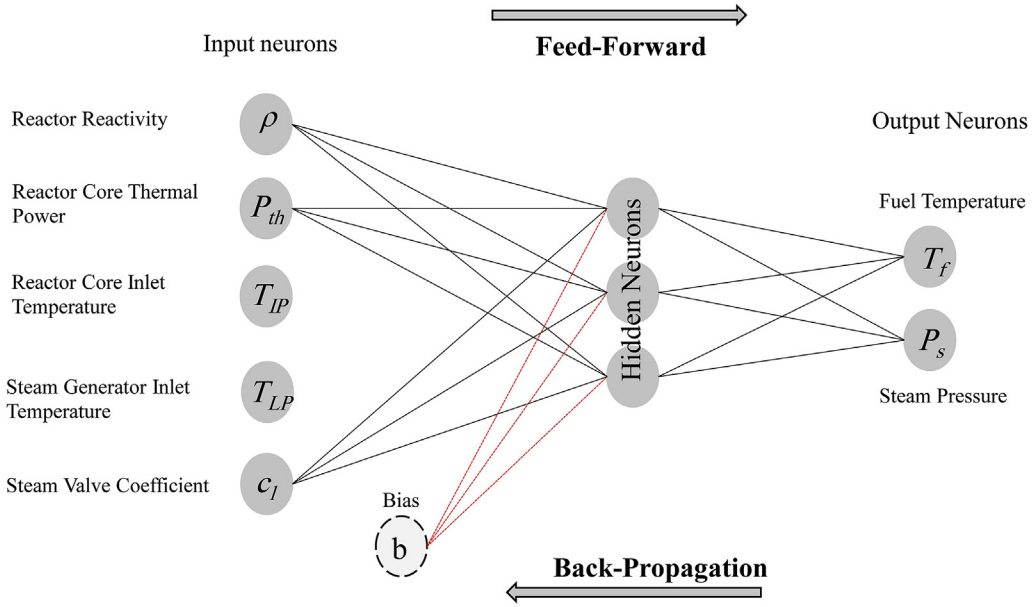

Fig. 4. Schematic Diagram of ANN employed in the present study to simulate the dynamic behavior of a PWR.

enables the network to simulate the underlying system behavior (i.e., T_f and P_s of a PWR in the context of the present study) for the training subset of the available data within a reasonable amount of time and with a minimum *MSE*. Gradient descent, conjugate gradient, and quasi-Newton are optimization algorithms used to train ANN-based models. Such algorithms, among other exist (e.g., Newton method, genetic algorithms, one-step secant method) have been proven to be highly efficient in training an ANN, and were thus used in the present study. Each of these algorithms can employ multiple functions for training. Three algorithms with eight training functions were assessed in this study (listed in Table 2), including: *i*) the GD algorithm (*traingd*, *traingdm*, and *trainrp*); *ii*) the CG algorithm (*trainscg*, *traincgp*, and *traincgf*); and *iii*) the QN algorithm (*trainbfg*, and *trainlm*). GD is the most widely applied training algorithm that adjusts $\boldsymbol{W}$ and $\boldsymbol{b}$ based on the descending gradient direction of the function [33]. The GD algorithm typically shows a fast-initial convergence rate but a slow zigzagging behavior when approaching the final solution [66]. Although the convergence of the GD algorithm is in the steepest descent direction, this may not necessarily produce the fastest convergence. The CG algorithm is, in contrast, performed along the conjugate direction, which generally provides a faster convergence rate than the steepest descent direction [67]. The QN algorithm relies on defining a better search direction based on the Hessian matrix—representing the second derivatives of the error function at the current values of the weights and biases [30]. An approximated version of

Table 2
Results of the PWR-NN for different training functions.

Algorithm	Training function	Description	n_h	Average number of epochs	R (T_f)	R (P_s)	*MSE* (T_f) $^oF^2$	*MSE* (P_s) psi^2	CPU time (s)
Gradient Descent	*traingd*	Gradient descent back propagation			did not converge				
	traingdm	Gradient descent with momentum back propagation			did not converge				
	trainrp	Resilient back propagation	4	879	0.9854	0.9985	201	22	11
			8	894	0.9886	0.9989	157	16	13
			11	904	0.9893	0.9991	147	14	15
Conjugate Gradient	*trainscg*	Scaled conjugate gradient back propagation	4	276	0.9844	0.9990	215	15	6
			8	303	0.9876	0.9991	171	13	8
			11	301	0.9881	0.9991	164	13	9
	traincgp	Conjugate gradient back propagation with Polak-Ribiére updates	4	270	0.9843	0.9991	217	14	12
			8	294	0.9874	0.9992	174	12	15
			11	332	0.9884	0.9992	159	12	21
	traincgf	Conjugate gradient back propagation with Fletcher-Reeves updates	4	264	0.9851	0.9991	206	13	12
			8	389	0.9892	0.9994	150	9	20
			11	431	0.9900	0.9994	138	8	28
Quasi-Newton	*trainbfg*	Broyden-Fletcher-Goldfarb- Shanno (BFGS) quasi-Newton back propagation	4	1000	0.9797	0.9949	279	77	65
			8	1000	0.9777	0.9926	306	112	78
			11	1000	0.9783	0.9937	298	95	95
	trainlm	Levenberg-Marquardt back propagation	4	265	0.9904	0.9995	132	7	17
			8	250	0.9934	0.9996	91	5	22
			11	336	**0.9940**	**0.9997**	**83**	**4**	38

the Hessian matrix is adopted within the QN algorithm to overcome the complexity and the large memory size that typically results from computing the exact one [68]. Although the convergence of the QN algorithm is typically faster than that of the CG algorithm, the latter is simpler and easier to apply [35]. In this study, all training functions were applied considering the different n_h values in order to obtain the best ANN architecture that resulted in the highest correlation coefficient (R) and lowest *MSE* between $\boldsymbol{O}$ and $\boldsymbol{P}$ during the training stage.

4. Results and discussion

4.1. Network training, testing and validation

The dataset obtained from the SD model of the PWR system was divided into three portions (training, validation, testing), as discussed earlier. The training and validation portions were used together for training purposes (i.e., obtaining the optimum values for $\boldsymbol{W}$ and $\boldsymbol{b}$ in Equation [1]). For training, the following parameters were fixed for all training functions within the MATLAB-*NN* toolbox: *i*) the performance function = *MSE*; *ii*) the performance goal = 0; *iii*) the adaptation learning function = LEARNGDM; *iv*) the learning rate parameter = 0.1; *v*) the activation function = TANSIG; and *vi*) the number of training iterations (max_epochs) = 1000. In addition, numerical measures haven been defined to assess the performance of each training function. Such measures include *MSE*, the Central Processing Unit (**CPU**) time elapsed at the end of the training, the number of epochs at the end of the training, and the average regression value ($\boldsymbol{R}$) over the training, validation, and testing subsets. The ANN was trained with each training function until the *MSE* remained constant for six consecutive epochs, except with the training function *trainbfg*, which reached the maximum of 1000 epochs first. As retraining the network typically results in different values of $\boldsymbol{W}$ and $\boldsymbol{b}$, the training stage was repeated 100 times using the same dataset. Table 2 shows the average training measures for each of the different ANNs that result from combining the different training functions with the different n_h values.

The training functions *traingd* and *traingdm* could not converge and therefore the corresponding ANNs were eliminated in this study. On the other hand, the ANNs corresponding to the rest of the training functions (i.e., *trainrp*, *trainscg*, *traincgp*, *traincgf*, *trainbfg*, and *trainlm*) were all adequately trained, but with differences in the CPU time consumed for training and the *MSE* values. The training function *trainlm* (a QN training algorithm) with eleven neurons showed the best performance in terms of the lowest *MSE* (Table 2), and the corresponding ANN is referred to as the developed ANN.

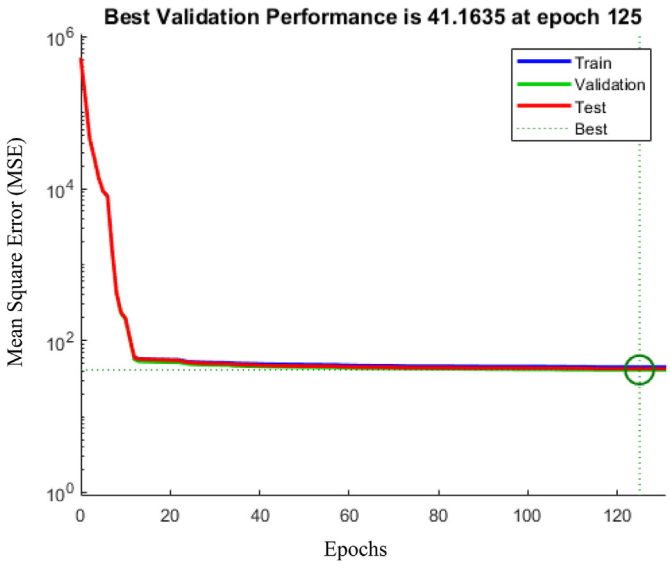

Fig. 5. MSE values under the *trainlm* training function and over the different training iterations.

Fig. 5 shows the *MSE* between the estimated and actual outputs under the *trainlm* training function. Large average *MSE* values were encountered during the first few iterations (<20) and subsequently decreased to smaller values. The results of the training stage demonstrate the ability of the developed ANN to successfully learn from the SD model-based dataset despite the complex dynamic behavior of the underlying PWR system and highlight the potential to use the developed ANN to predict the system response under new transients (i.e., new input values). Fig. 6 shows the relationship between the ANN estimated values and the SD actual outputs for the training, validation, and testing subsets when the trainlm training function was adopted with n_h equals to eleven. In Fig. 6, the symbol "Y" on the vertical axis represents the output from the developed ANN model, while the symbol "T" on the horizontal axis represents the target value that was estimated from the SD model. The line of best fit through the data for all three subsets (i.e., training, validation, and testing) has nearly a unit slope and zero intercept, reflecting the ability of the developed ANN to replicate the SD model outputs.

4.2. Additional network testing

After training, it is crucial to test the developed ANN to ensure its efficacy to predict the response of the PWR system under new transients. Therefore, the SD model was employed to simulate the PWR behavior under new transients, and the corresponding outputs were used for additionally testing the developed ANN. These new transients were: 1) a perturbation in reactivity by +0.001; 2) a perturbation in reactivity by +0.001 including uncertainties in the system physical parameters and operating conditions; 3) a perturbation in steam valve coefficient by +7.5%; 4) a deviation in steam valve coefficient by +7.5% including uncertainties in the system physical parameters and operating conditions; and, 5) a perturbation in core inlet temperature by +7.5 °F.

4.2.1. Transient 1: Increase in reactivity

In Transient 1, the performance of the developed ANN was evaluated under an increase of reactivity by +0.001. Increasing the reactivity level leads to a higher fuel temperature, which subsequently causes more heat energy to transfer to the primary cooling system. Such heat then transfers to the secondary coolant system through the metal U tubes and converts the secondary coolant into steam. As a result, additional steam is produced in the steam generators, leading to a higher steam pressure if there is no change in the steam valve opening. The developed ANN sufficiently reproduced the SD model estimates of fuel temperature and steam pressure under Transient 1, as shown in Fig. 7a and 7b, respectively. The developed ANN underestimated the fuel temperature at the beginning of the transient (t = 0 s) by merely 1.6%, and the deviations between the ANN and SD model estimates of fuel temperature decreased significantly as the reactor approached steady-state conditions. On the other hand, the developed ANN efficiently replicated the temporally changing steam pressure values estimated by the SD model with negligible deviations.

4.2.2. Transient 2: Increase in reactivity with other parameter uncertainties

Several sources of uncertainty are typically present in complex systems (e.g., uncertainty in input parameters, uncertainty in

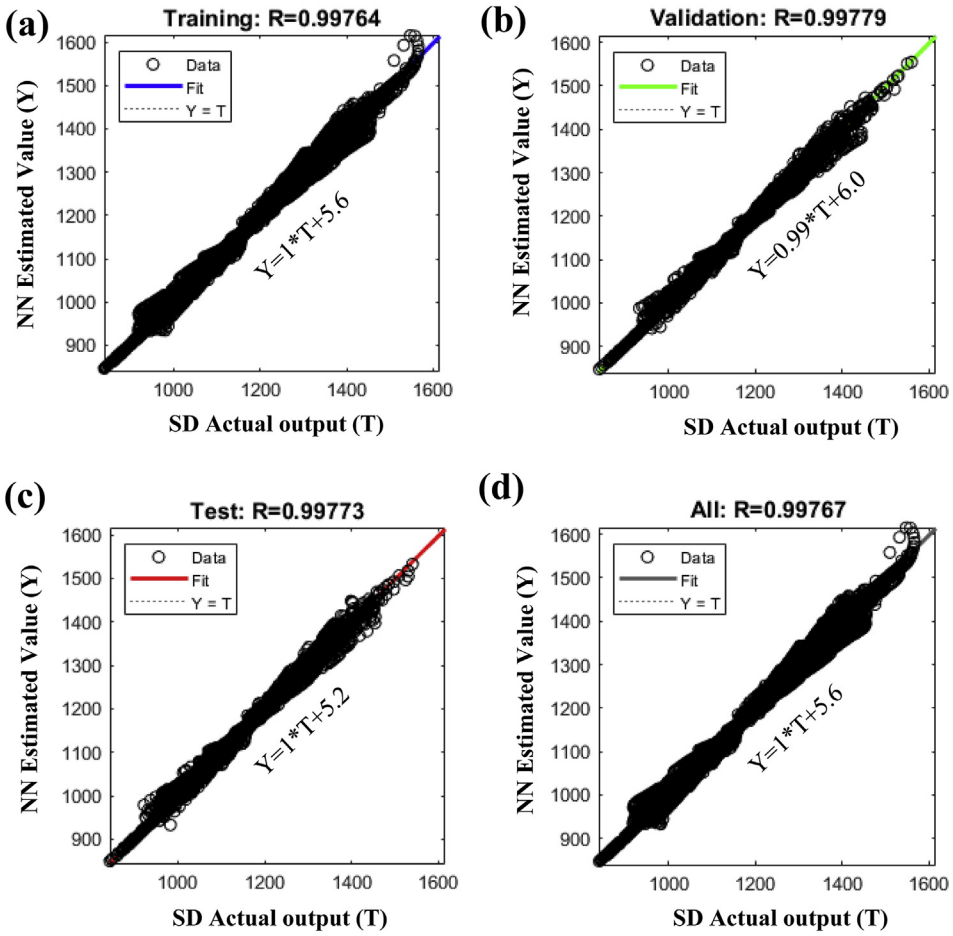

Fig. 6. Regression values of the ANN with 11 hidden layer neurons under the *trainlm* training function for the: (a) training subset, (b) validation subset, (c) testing subset, and (d) all subsets combined together.

model structure). Integrating the uncertainty of the input parameters in the developed ANN is therefore essential to reflect the real behavior inside the underlying PWR. The effect of the uncertain physical parameters and operating condition on the system response were considered during the additional testing of the developed ANN. A total of 5000 realizations of the SD model parameters were generated, and the corresponding outputs were estimated under a reactivity change of +0.001. The ANN inputs (i.e., ρ, P_{th}, T_{IP}, T_{LP}, c_l) were extracted from the SD model outputs, and were used to evaluate the uncertainty in predicting the temporal fuel temperature and steam pressure. The temporal minimum, median, and maximum fuel temperature and steam pressure were predicted, as shown in Fig. 8a and 8b, respectively. The considered uncertainty led to an increase in the fuel temperature and steam pressure by 1.2% and 1.6%, respectively, compared to the median value under steady state conditions. The developed ANN produced similar results to those of the SD model with negligible differences. In addition, the statistical distributions of fuel temperature and

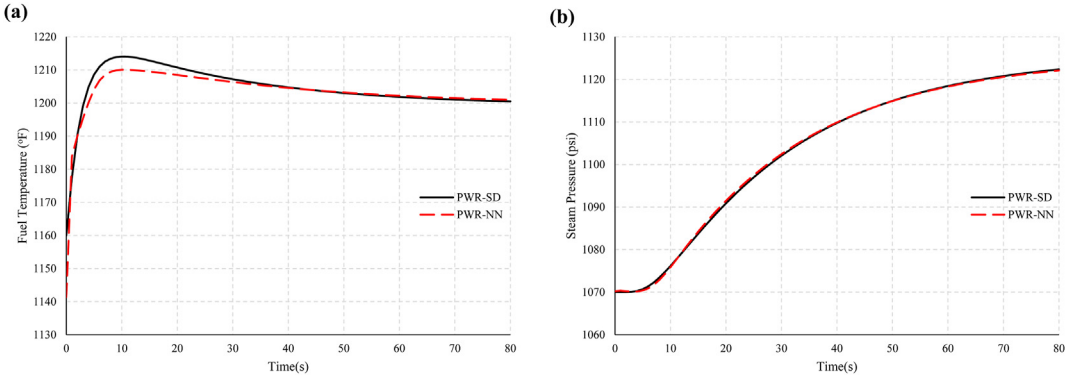

Fig. 7. (a) Comparison between NN prediction and SD estimate of fuel temperature due to an increase in reactivity level by +0.001 (Transient 1), and (b) comparison between NN prediction and SD estimate of steam pressure at the same transient level.

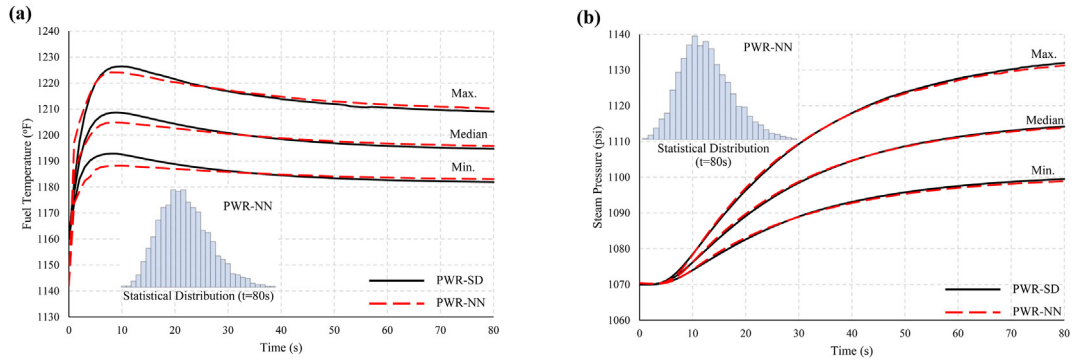

Fig. 8. (a) Comparison between NN predictions and SD estimates of uncertain fuel temperature due an increase in reactivity level by +0.001 (Transient 2), and (b) comparison between NN predictions and SD estimates of uncertain steam pressure at the same transient level.

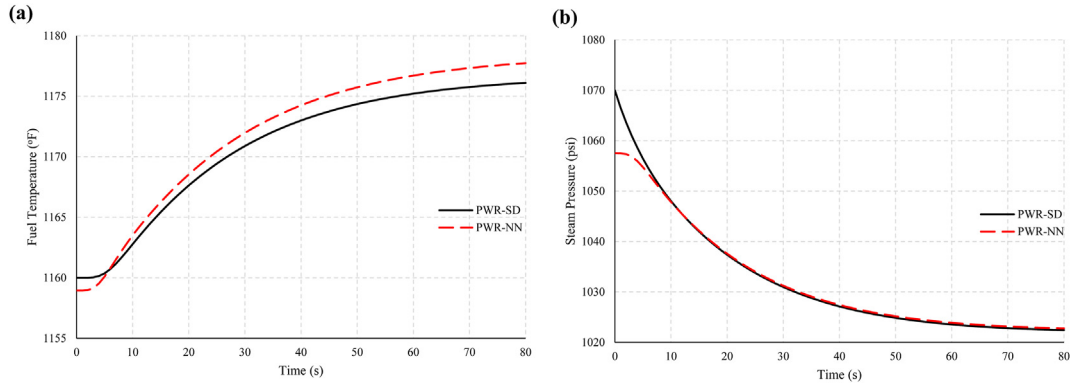

Fig. 9. (a) Comparison between NN prediction and SD estimate of fuel temperature due to an increase in steam valve coefficient by +7.5% (Transient 3), and (b) comparison between NN prediction and SD estimate of steam pressure at the same transient level.

steam pressure predicted using the developed ANN do not follow a uniform distribution but can rather be approximated by a 3-parameter lognormal distribution, where the maximum and minimum responses have a lower probability of occurrence compared to the mean response. These statistical distributions are similar to those estimated using the SD model.

4.2.3. *Transient 3: Increase in steam valve coefficient*

The developed ANN was also tested under an increase in the steam valve coefficient (corresponding to increasing the steam valve opening) by +7.5%. The steam pressure inside the steam generator decreases immediately after increasing the steam valve opening. This is followed by a reduction in the reactor core inlet temperature, which causes a positive reactivity and a subsequent increase in the fuel temperature, as shown in Fig. 9a. As a result, more heat energy is generated inside the reactor core to accommodate the reduction in steam pressure. The developed ANN sufficiently simulated this physical behavior and reproduced the SD model outputs under this transient, with maximum differences of 0.14% and 1.16% in the fuel temperature (t = 80 s) and the steam pressure (t = 0 s), respectively (Fig. 9a and 9b).

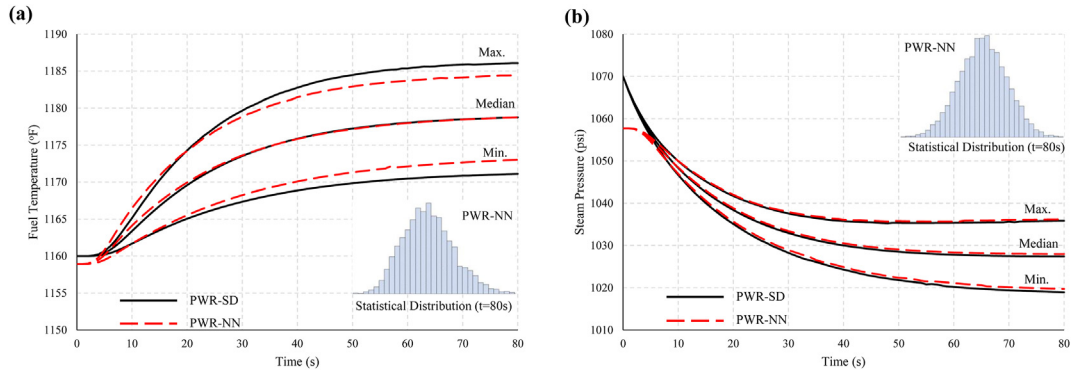

Fig. 10. (a) Comparison between NN predictions and SD estimates of uncertain fuel temperature due to an increase in steam valve coefficient by +7.5% (Transient 4), and (b) comparison between NN predictions and SD estimates of uncertain steam pressure at the same transient level.

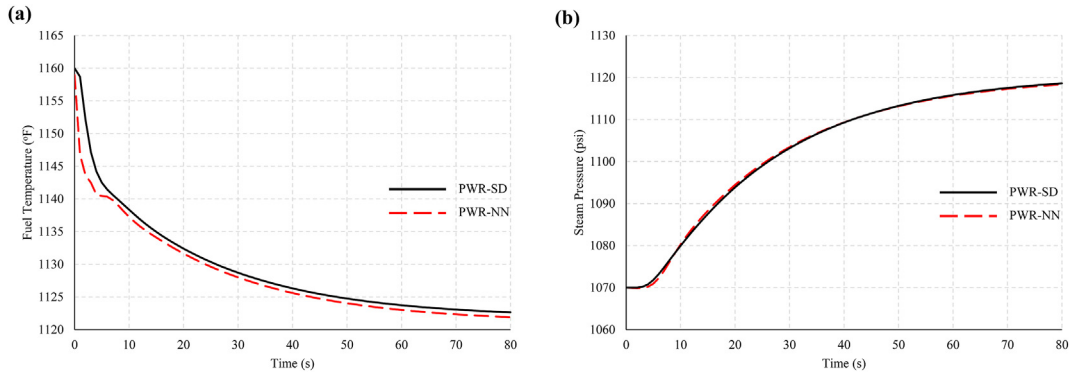

Fig. 11. (a) Comparison between NN prediction and SD estimate of fuel temperature due to an increase in the reactor core inlet temperature by +7.5 °F (Transient 5), and (b) comparison between NN prediction and SD estimate of steam pressure at the same transient level.

4.2.4. Transient 4: Increase in steam valve coefficient with other parameters uncertainties

In Transient 4, the developed ANN was tested under an increase in the steam valve coefficient by +7.5% considering uncertain physical parameters and operating conditions. Similar to Transient 2, a total of 5000 realizations were utilized as the inputs for the developed ANN and the corresponding temporal fuel temperature and steam pressure uncertainty were evaluated. The SD model estimates were efficiently reproduced using the developed ANN, with maximum differences of 0.162% and 0.137% in the minimum and maximum fuel temperature at t = 80 s, respectively, and 1.145% in the steam pressure at t = 0 s (Fig. 10). In addition, the statistical distribution of the ANN predictions under steady state conditions can be approximated by a 3-parameter lognormal distribution (Fig. 10a and 10b).

4.2.5. Transient 5: Increase in reactor Inlet temperature

Finally, the performance of the developed ANN was evaluated under an increase in the reactor core inlet temperature of 7.5 °F. Increasing the coolant temperature results in a negative reactivity feedback that reduces the fuel temperature, as shown in Fig. 11a. On the other hand, more heat energy is transferred to the secondary coolant system due to the increasing coolant temperature in the primary coolant system. In addition, the secondary coolant is converted into steam, leading to a higher steam pressure. The SD model outputs were adequately predicted using the developed ANN under this transient, with maximum difference of 1.0% in the fuel temperature at t = 1 s and negligible deviations in the steam pressure as shown in Fig. 11a and 11b, respectively.

4.2.6. Overall evaluation of the developed artifical neural network

It is important to note that other data driven models can be used to predict the response of the different subsystems in NPPs; however, the present study employed ANN due to the latter's demonstrated efficacy in predicting more than one critical parameter simultaneously [20,22,23,37,38,59]. It should thus be emphasized that the ANN developed in the present study focused on replicating the response of PWR only as a proof-of-concept demonstration, rather than showing the limit to DDM applications in other types of reactor systems per se. Additionally, the obtained results demonstrated the ability of the developed ANN model to successfully learn from the dataset despite the complex and dynamic nature of the PWR system. Therefore, other ANN architectures (i.e., CNN, RNN) were not utilized in the present study.

The main advantage of using data driven modelling is the ability to capture the input-output relationships without overburdening the model with the physical processes in the underlying system. This fact leads DDMs to be essentially black boxes, where the results may not be interpreted physically. However, training, validating, and testing DDMs using actual data (that can be interpreted either based on physics or based on operator's experience) enable such models to be used in lieu of physics-based ones in some situations. It should be noted that integrating data-driven and physics-based models can also be used to enhance the physical interpretability of the DDM results. As such, *hybrid models*— integrating data-driven and physics-based models can present the best option to develop physics-guided intelligent decision support tools to mitigate risks of rarer event scenarios. In such type of models, ANN can be trained under plant operational data, experience database, and data from rare event simulations to consider for example plant ageing with time, operational transients, and rare events in predicting the plant behavior. Such hybrid models can therefore be key for NPP operators and managers to take rapid and reliable actions under abnormal conditions.

Overall, the developed ANN has been trained under 32 different transients to simulate the dynamic interaction between complex systems inside the PWR. The ANN performance under Transients 1 through 5 supports its ability to predict PWR physical behaviors similar to a SD modeling approach, but with a lower computational cost and time. It should be noted that the analysis time including Monte-Carlo simulation by SD model is approximately 15–20 min compared to only a few seconds in case of using the ANN model (Table 2), which supports the use of the ANN model as a rapid decision support tool in lieu of physics-based models. The value of minimizing the simulation time is especially apparent when considering the complexities associated with analyzing a full NPP as a system-of-systems and the need for rapid decision making in the case of an imminent accident. Additionally, the developed ANN model has sufficiently replicated the response of two reactor critical parameters simultaneously. Other critical parameters (e.g., coolant flow in the primary heat transport system, turbine inlet steam pressure) can also be predicted when the corresponding observations become available to train the ANN. Finally, the developed ANN can be used to provide the plant operators with early warnings under the considered transients. This can reduce the likelihood of having severe accident consequences and ultimately enhance the overall safety conditions.

5. Conclusions

The present study aimed at exploring the potential of applying AI tools within the nuclear engineering field, and specifically for the prediction of NPP behavior. A previously published validated SD (physics-based) model was employed to generate data pertaining

to a reactor's dynamic behavior under different transients. The data was subsequently utilized to develop a corresponding ANN (data-driven) model. The uncertainty associated with the reactor system's physical parameters and operating conditions were also incorporated in these transient analyses. A feed-forward backpropagation ANN was trained based on 32 transients to model the interaction between complex systems inside the PWR, including the reactor core, primary and secondary cooling systems, hot and cold legs, reactor core inlet and outlet plenums, and steam generator inlet and outlet plenums. The ANN was developed with an input layer with five nodes, a single hidden layer with different numbers of neurons, and an output layer with two nodes. Three back-propagation algorithms with eight training functions were utilized during the model training stage. The ANN corresponding to the *trainlm* function, with eleven neurons in the hidden layer, showed the best performance compared to other training functions.

The developed ANN was subsequently tested under new transients representing perturbations in reactivity, steam valve coefficient, and core inlet temperature. In all cases, the developed ANN reproduced the SD model estimates of the temporal fuel temperature and steam pressure with negligible differences (no more than 1.6%). In addition, the predicted statistical distributions of fuel temperature and steam pressure using ANN are compatible with the corresponding distributions from the SD simulation model when input uncertainties are considered. In an actual NPP, the developed ANN would provide a computationally efficient alternative compared to physics-based simulators, especially for considering the uncertain system physical parameters and operation conditions. In addition, the developed ANN can be utilized as an early warning tool to enable the development of effective risk mitigation strategies under unexpected operating conditions and can therefore serve as a rapid decision support system for NPP operators and emergency managers. Moving forward, the developed ANN can be trained using real plant operation data and different transients in different systems to cover all possible scenarios that can occur during normal or abnormal operating conditions. The adoption and development of AI tools within the nuclear engineering field will enable major breakthroughs in mitigating the risk of accidents and human errors when dealing with complex, dynamic and critical systems such as NPPs. AI also has the potential of autonomously controlling and manage NPP including for example small modular reactors (SMR), both on-site and remotely. It is also expected that the next generations of NPP will include additional intelligent safety systems that enable more accurate and timely control of the plant, especially as SMR technology become mainstream.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The financial support for the study was provided through the Canadian Nuclear Energy Infrastructure Resilience under Seismic Systemic Risk (CaNRisk) – Collaborative Research and Training Experience (CREATE) program of the Natural Science and Engineering Research Council (NSERC) of Canada. Additional support from the INTERFACE Institute and the INViSiONLab is also acknowledged.

References

[1] IAEA, Accident Monitoring Systems for Nuclear Power Plants, vol. 16, IAEA Nuclear Energy series NP-T-3, 2015.
[2] J.H. Min, D. Kim, C. Park, Demonstration of the validity of the early warning in online monitoring system for nuclear power plants, Nucl. Eng. Des. 349 (2019) 56–62, https://doi.org/10.1016/j.nucengdes.2019.04.028.
[3] N. Tamimi, S. Samani, M. Minaei, F. Harirchi, An Artificial Intelligence Decision Support System for Unconventional Field Development Design, 2019, https://doi.org/10.15530/urtec-2019-249.
[4] I.S. Korovin, I.A. Kalyaev, Modern decision support systems in oil industry: types, approaches and applications, in: Int. Conf. Test, Meas. Comput. Method, 2015, pp. 141–144.
[5] S. Ahmad, S.P. Simonovic, An intelligent decision support system for management of floods, Water Resour. Manag. (2006) 391–410, https://doi.org/10.1007/s11269-006-0326-3.
[6] F.G. Filip, Decision support and control for large-scale complex systems, Annu. Rev. Contr. 32 (2008) 61–70, https://doi.org/10.1016/j.arcontrol.2008.03.002.
[7] G. Phillips-Wren, Intelligent Decision Support Systems, Wiley-Blackwell, Hoboken, NJ, USA, 2013, https://doi.org/10.1002/9781118522516.ch2.
[8] M. Rätz, A.P. Javadi, M. Baranski, K. Finkbeiner, D. Müller, Automated data-driven modeling of building energy systems via machine learning algorithms, Energy Build. 202 (2019), 109384, https://doi.org/10.1016/j.enbuild.2019.109384.
[9] H. Bao, N.T. Dinh, J.W. Lane, R.W. Youngblood, A data-driven framework for error estimation and mesh-model optimization in system-level thermal-hydraulic simulation, Nucl. Eng. Des. 349 (2019) 27–45, https://doi.org/10.1016/j.nucengdes.2019.04.023.
[10] D. Solomatine, A. Ostfeld, Data-driven modelling: some past experiences and new approaches approaches, J. Hydroinf. 10 (2008), https://doi.org/10.2166/hydro.2008.015.
[11] T.M. Mitchell, Machine Learning, McGraw-Hill, New York, 1997.
[12] F.J. Montáns, F. Chinesta, R. Gómez-Bombarelli, J.N. Kutz, Data-driven modeling and learning in science and engineering, Compt. Rendus Mec. 347 (2019) 845–855, https://doi.org/10.1016/j.crme.2019.11.009.
[13] K.R. Holdaway, Harness Oil and Gas Big Data with Analytics: Optimize Exploration and Production with Data-Driven Models, John Wiley and Sons, Hoboken, New Jersey., 2014.
[14] A. Burchard-levine, S. Liu, F. Vince, M. Li, A. Ostfeld, A hybrid evolutionary data driven model for river water quality early warning, J. Environ. Manag. 143 (2014) 8–16, https://doi.org/10.1016/j.jenvman.2014.04.017.
[15] J. Zhang, F. Wang, K. Wang, W. Lin, X. Xu, C. Chen, Data-driven intelligent transportation Systems : a survey, IEEE Trans. Intell. Transport. Syst. 12 (2011) 1624–1639, https://doi.org/10.1109/TITS.2011.2158001.
[16] N.P. Oxtoby, A.L. Young, D.M. Cash, T.L.S. Benzinger, A.M. Fagan, J.C. Morris, R.J. Bateman, N.C. Fox, J.M. Schott, D.C. Alexander, Data-driven models of dominantly-inherited Alzheimer's disease progression, Brain (2018) 1–16, https://doi.org/10.1093/brain/awy050.
[17] M. Gomez, A. Tokuhiro, K. Welter, Q. Wu, Nuclear energy system's behavior and decision making using machine learning, Nucl. Eng. Des. 324 (2017) 27–34, https://doi.org/10.1016/j.nucengdes.2017.08.020.
[18] T. Foshch, F. Portela, J. Machado, M. Maksimov, Regression models of the nuclear power unit VVER-1000 using data mining techniques, Procedia Comput. Sci. 100 (2016) 253–262, https://doi.org/10.1016/j.procs.2016.09.151.
[19] L. Fahrmeir, T. Kneib, S. Lang, B. Marx, Regression Models, Springer, Berlin, Heidelberg, 2013, https://doi.org/10.1007/978-3-642-34333-9_2.
[20] S.R. Patra, R. Jehadeesan, S. Rajeswari, S.A. Satyamurthy, Artificial neural network model for intermediate heat exchanger of nuclear reactor, Int. J. Comput. Appl. 1 (2010).
[21] D. Maljovec, S. Liu, B. Wang, D. Mandelli, P. Bremer, V. Pascucci, C. Smith, Analyzing simulation-based PRA data through traditional and topological clustering : a BWR station blackout case study, Reliab. Eng. Syst. Saf. 145 (2016) 262–276, https://doi.org/10.1016/j.ress.2015.07.001.
[22] O.I. Abiodun, A. Jantan, A.E. Omolara, K.V. Dada, N.A. Mohamed, H. Arshad, State-of-the-art in artificial neural network applications: a survey, Heliyon (2018), e00938, https://doi.org/10.1016/j.heliyon.2018.
[23] H.H. Kang, M. Kaya, S. Hajimirza, A data driven artificial neural network model for predicting radiative properties of metallic packed beds, Quant. Spectrosc. Radiat. Transf. (2019) 66–72, https://doi.org/10.1016/j.jqsrt.2019.01.013.
[24] J. Li, J. Cheng, J. Shi, F. Huang, Brief introduction of back propagation (BP) neural network algorithm and its improvement, Adv. Comput. Sci. Inf. Eng. 169 (2012), https://doi.org/10.1007/978-3-642-30223-7_87.
[25] K. O'Shea, R. Nash, An introduction to convolutional neural networks, 2015. https://arxiv.org/abs/1511.08458.
[26] T. Mikolov, M. Karafiat, L. Burget, J. Cernock, S. Khudanpur, Recurrent neural network based language model, in: Interspeech, 2010.
[27] Y. Singh, A.S. Chauhan, Neural networks in data mining, J. Theor. Appl. Inf. Technol. 5 (2009) 36–42.
[28] N.M. Nawi, R.S. Ransing, M.R. Ransing, An improved conjugate gradient based learning algorithm for back propagation neural networks, Int. J. Comput. Intell. (2008) 46–55.
[29] M.F. Moller, A scaled conjugate gradient algorithm for fast supervised learning, Neural Network. 6 (1993) 525–533.

[30] M. Moller, Efficient Training of Feed-Forward Neural Networks, Aarhus University, Computer Science Department, Ph.D. Thesis, 1997.
[31] R. Battiti, First- and second-order methods for learning: between steepest descent and Newton's method, Neural Comput. 4 (1992) 141–166.
[32] M.T. Hagan, M.B. Menhaj, Training feedforward networks with the marquardt algorithm, IEEE Trans. Neural Network. 5 (1994) 2–6.
[33] B. Sharma, P.K. Venugopalan, Comparison of neural network training functions for hematoma classification in brain CT images, IOSR J. Comput. Eng. 16 (2014) 31–35.
[34] H. Mustafidah, S. Hartati, R. Wardoyo, A. Harjoko, Selection of most appropriate backpropagation training algorithm in data pattern recognition, Int. J. Comput. Trends Technol. 14 (2014) 92–95.
[35] H. Sug, The effect of training set size for the performance of neural networks of classification, WSEAS Trans. Comput. 9 (2010) 1297–1306.
[36] S. Lawrence, C.L. Giles, A.C. Tsoi, Lessons in neural network training: overfitting may be harder than expected, in: Proc. Fourteenth Natl. Conf. Artif. Intell. AAAI-97, AAAI Press, Menlo Park, California, 1997, pp. 540–545.
[37] R.E. Uhrig, Use of neural networks in nuclear power plants, ISA Trans. 32 (1993) 139–145.
[38] Z. Guo, R.E. Uhrig, Use of artificial neural networks to analyze nuclear power plant performance, Nucl. Technol. 5450 (2017), https://doi.org/10.13182/NT92-A34701.
[39] A. Varuttamaseni, Bayesian Network Representing System Dynamics in Risk Analysis of Nuclear Systems, Ph.D. Thesis, University of Michigan, 2011.
[40] M. Knochenhauer, J.E. Holmberg, Guidance for the Definition and Application of Probabilistic Safety Criteria, Swedish Radiation Safety Authority, 2011.
[41] United States Nuclear Regulatory Commission, Acceptance criteria for emergency core cooling systems for light-water nuclear power reactors, 2017. https://www.nrc.gov/reading-rm/doc-collections/cfr/part050/part050-0046.html.
[42] M. El-Sefy, M. Ezzeldin, W. El-Dakhakhni, L. Wiebe, S. Nagasaki, System dynamics simulation of the thermal dynamic processes in nuclear power plants, Nucl. Eng. Technol. 51 (2019) 1540–1553, https://doi.org/10.1016/j.net.2019.04.017.
[43] J.G. Thakkar, Correlation of Theory and Experiment for the Dynamics of a Pressurized Water Reactor, University of Tennessee, 1975. https://trace.tennessee.edu/utk_gradthes/2696%0A%0A.
[44] T.W. Kerlin, E.M. Katz, J.G. Thakkar, J.E. Strange, Theoretical and experimental dynamic analysis of the HB Robinson nuclear plant, Nucl. Technol. 30 (1976) 299–316, https://doi.org/10.13182/NT76-A31645.
[45] M. Ali, Lumped Parameter, State Variable Dynamic Models for U-Tube Recirculation Type Nuclear Steam Generators, Ph.D. Thesis, University of Tennessee, 1976, https://trace.tennessee.edu/utk_graddiss/2548%0A%0A.
[46] S. Arda, K.E. Holbert, J. Undrill, Development of a linearized model of a pressurized water reactor generating station for power system dynamic simulations, in: 45th North Am. Power Symp, NAPS, 2013, https://doi.org/10.1109/NAPS.2013.6666832, 2013.
[47] S. Arda, Implementing a Nuclear Power Plant Model for Evaluating Load-Following Capability on a Small Grid, MASc Thesis, Arizona State University, 2013.
[48] B. Puchalski, T.A. Rutkowski, K. Duzinkiewicz, Nodal models of Pressurized Water Reactor core for control purposes – a comparison study, Nucl. Eng. Des. 322 (2017) 444–463, https://doi.org/10.1016/j.nucengdes.2017.07.005.
[49] A.I. 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, Reliab. Eng. Syst. Saf. 174 (2018) 19–28, https://doi.org/10.1016/j.ress.2018.02.005.
[50] Y. Perin, J. Jimenez, Application of the best-estimate plus uncertainty approach on a BWR ATWS transient using the NURESIM European code platform, Nucl. Eng. Des. 321 (2017) 48–56, https://doi.org/10.1016/j.nucengdes.2017.05.018.
[51] M.I. Radaideh, W.A. Wieselquist, O. Ridge, T. Kozlowski, A new framework for sampling-based uncertainty quantification of the six-group reactor kinetic parameters, Ann. Nucl. Energy (2018), https://doi.org/10.1016/j.anucene.2018.11.043.
[52] C.S. Brown, H. Zhang, Uncertainty quantification and sensitivity analysis with CASL Core Simulator VERA-CS, Ann. Nucl. Energy J. 95 (2016) 188–201, https://doi.org/10.1016/j.anucene.2016.05.016.
[53] C. Demazière, I. Pázsit, Evaluation of the boron dilution method for moderator temperature coefficient measurements, Nucl. Technol. 140 (2002) 147–163.
[54] S.G. Zimmerman, J.C. Brittingham, M.L. Reed, R.P. Bandera, P.F. Crawley, PWR Reactor Physics Methodology Using CASMO-4/SIMULATE-3, Arizona Public Service Company, 1999.
[55] P. Romojaro, F. Álvarez-Velarde, N. García-Herranz, Sensitivity methods for effective delayed neutron fraction and neutron generation time with summon, Ann. Nucl. Energy 126 (2019) 410–418, https://doi.org/10.1016/j.anucene.2018.11.042.
[56] M. El-Sefy, M. Ezzeldin, W. El-Dakhakhni, S. Nagasaki, L. Wiebe, Dynamic probabilistic risk assessment of core damage under different transients using system dynamics simulation approach, Chapter 4, Ph.D. Thesis, 2021, http://hdl.handle.net/11375/26246.
[57] MATLAB and Statistics Toolbox Release 2018a, The MathWorks, Inc., Natick, Massachusetts, United States, 2018.
[58] C. Perez, Neural Networks Using Matlab. Cluster Analysis and Classification, Lulu Press, Inc, 2019.
[59] E. Arce-Medina, J.I. Paz-Paredes, Artificial neural network modeling techniques applied to the hydrodesulfurization process, Math. Comput. Model. 49 (2009) 207–214, https://doi.org/10.1016/j.mcm.2008.05.010.
[60] R. Rallo, A. Arenas, F. Giralt, Neural virtual sensor for the inferential prediction of product quality from process variables, Comput. Chem. Eng. 26 (2002) 1735–1754, 26.
[61] S. Haykin, Neural Networks, second ed., A Comprehensive Foundation, Prentice Hall PTR, Upper Saddle River, NJ, United States, 1999.
[62] G. Cybenkot, Approximation by superpositions of a sigmoidal function, Math. Control. Signals, Syst. 2 (1989) 303–314.
[63] A.H. Gandomi, D.A. Roke, Advances in Engineering Software Assessment of artificial neural network and genetic programming as predictive tools, Adv. Eng. Software 88 (2015) 63–72, https://doi.org/10.1016/j.advengsoft.2015.05.007.
[64] S. Xu, L. Chen, A novel approach for determining the optimal number of hidden layer neurons for FNN's and its application in data mining, in: 5th Int. Conf. Inf. Technol. Appl, 2008, pp. 683–686.
[65] R. Hecht-Nielsen, Kolmogorov's mapping neural network existence theorem, in: IEEE First Int. Conf, Neural Networks, San Diego, CA, 1987.
[66] A.Y. Krylatov, A.P. Hirokolobova, Projection approach versus gradient descent for network' s flows assignment problem, Learn. Intell. Optim. (2017) 7–69404, https://doi.org/10.1007/978-3-319-.
[67] E.M. Johansson, F.U. Dowla, D.M. Goodman, Backpropagation learning for multilayer feed-forward neural networks using the conjugate gradient method, Int. J. Neural Syst. 2 (1992) 291–301.
[68] R. Setiono, L.C.K. Hui, Use of a quasi-Newton method in a feedforward neural network construction algorithm, IEEE Trans. Neural Network. 6 (1995), 0–4.