Nuclear Engineering and Technology 55 (2023) 3409–3416

Contents lists available at ScienceDirect

Nuclear Engineering and Technology

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

Original Article

Physics informed neural networks for surrogate modeling of accidental scenarios in nuclear power plants

Federico Antonello [a, *], Jacopo Buongiorno [a], Enrico Zio [b, c]

[a] Department of Nuclear Science and Engineering, Massachusetts Institute of Technology (MIT), Cambridge, MA, 02139, USA
[b] Mines Paris-PSL University, Centre de Recherche sur les Risques et les Crises, 06904, Sophie Antipolis, France
[c] Energy Department, Politecnico di Milano, Via La Masa 34, 20156, Milan, Italy

ARTICLE INFO

Article history:
Received 10 January 2023
Received in revised form
18 April 2023
Accepted 16 June 2023
Available online 18 June 2023

Keywords:
Nuclear power plant
Accidental scenario
Modeling and simulation
Deep-learning
Physics informed neural network
Surrogate model
Metamodel
Nuclear battery
Nuclear microreactor

ABSTRACT

Licensing the next-generation of nuclear reactor designs requires extensive use of Modeling and Simulation (M&S) to investigate system response to many operational conditions, identify possible accidental scenarios and predict their evolution to undesirable consequences that are to be prevented or mitigated via the deployment of adequate safety barriers. Deep Learning (DL) and Artificial Intelligence (AI) can support M&S computationally by providing surrogates of the complex multi-physics high-fidelity models used for design. However, DL and AI are, generally, low-fidelity 'black-box' models that do not assure any structure based on physical laws and constraints, and may, thus, lack interpretability and accuracy of the results. This poses limitations on their credibility and doubts about their adoption for the safety assessment and licensing of novel reactor designs.

In this regard, Physics Informed Neural Networks (PINNs) are receiving growing attention for their ability to integrate fundamental physics laws and domain knowledge in the neural networks, thus assuring credible generalization capabilities and credible predictions. This paper presents the use of PINNs as surrogate models for accidental scenarios simulation in Nuclear Power Plants (NPPs). A case study of a Loss of Heat Sink (LOHS) accidental scenario in a Nuclear Battery (NB), a unique class of transportable, plug-and-play microreactors, is considered. A PINN is developed and compared with a Deep Neural Network (DNN). The results show the advantages of PINNs in providing accurate solutions, avoiding overfitting, underfitting and intrinsically ensuring physics-consistent results.

1. Introduction

The licensing of future-generation Nuclear Power Plants (NPPs) requires extensive development of Modeling and Simulation (M&S) to investigate the system response to various operational and environmental conditions for identifying possible accident scenarios, predicting their consequences and assessing the effectiveness of safety systems and barriers [1]. Safety analysis frameworks based on Best Estimate Plus Uncertainty (BEPU) are becoming popular, as they allow computing Quantities of Interest (QoI) and their safety margins (e.g., peak cladding temperature and fuel pellet maximum temperature) by considering the response to accidental scenarios modeled without excessive conservativism [2]. Uncertainty of design parameters, model input variables and operating conditions are propagated through high-fidelity Best Estimate (BE) models, typically via a Monte-Carlo approach, to quantify the uncertainty in the QoIs and provide a knowledgeable characterization of the NPP risk profile [3]. However, BE models are complex and include a large number of parametrized quantities (i.e., material properties, initial conditions, boundary conditions, constitutive laws, etc.), which make them computationally expensive so that their direct use for UQ, which could require a large number of lengthy simulations, is often prohibitive [4]. Therefore, computationally cheap surrogates must be built to approximate the input-output response of the original BE models [5].

The existing surrogate modeling approaches can be categorized into reduced order models (ROMs) and data-driven models [6]. The former category reduces the degrees of freedom and the accuracy of the physical models to alleviate the computational burden. However, ROMs are code-intrusive and their potential to speed up the simulations is limited when strong nonlinearity exists [7]. The latter category uses a limited number of data and trajectories

* Corresponding author.
E-mail addresses: afederic@mit.edu, federico.antonello@polimi.it (F. Antonello).

https://doi.org/10.1016/j.net.2023.06.027

simulated by the BE model to learn, replicate and generalize the input-output relation [8]. Examples are Machine Learning (ML) models such as Support Vector Machine (SVM) [9], Gaussian Processes (GP) [10–12], Polynomial Chaos Expansion (PCE) [13,14], Proper Orthogonal Decomposition (POD) (Kang, Tian, Chen, Li, & Wang, 2022), Finite Mixture Models (FMMs) [15], Artificial Neural Networks (ANNs) (Kang, Tian, Chen, Li, & Chu, 2022 [16]; and Radial Basis Functions (RBF) [17], and Deep Learning (DL) models, such as Deep Neural Networks (DNNs) [18] and Deep GP [1]. ML and DL metamodels have been widely applied in the nuclear industry, e.g., to replace multi-physics complex codes [19], to accelerate thermal-hydraulic analyses [20], to quantify safety margins [21], to perform uncertainty analysis [3], sensitivity analysis [22], neutron transport predictions [23] and autonomous control of next-generation reactors [24]. [16] used an ANN to create a surrogate thermal-hydraulic model of a NPP assessing the failure probability of a passive safety system. In Ref. [18], a Deep Neural Network (DNN) is used to simulate a postulated loss-of-coolant accident (LOCA), and support uncertainty quantification and probabilistic decision analysis [25]. developed a surrogate model based on GP to assess failure probabilities in radioactive waste repositories [26]. used a DNN as a surrogate model of normal and abnormal conditions of a microreactor to perform multi-objective optimization of control drum operation [8]. examined the application of twenty-five ML surrogates to model a fire hazard in a nuclear power plant [15]. developed a surrogate model based on FMM and adaptive GP to simulate the decay heat removal of a passive safety system in a NPP [23]. used DNN to estimate reactor core parameters in a Pressurized Water Reactor (PWR) to optimize its design [27]. developed a DL-based surrogate to emulate thermal-hydraulic (TH) codes, whereas [28] considered Long-Short Term Memory (LSTM) networks to substitute massive simulations of TH dynamics and performed the Probabilistic Safety Assessment (PSA) of a NPP [29]. proposed a ML-based surrogate to optimize the configuration of a spent fuel cask in a seismic environment. In Ref. [30], a surrogate is used to perform UQ and inverse UQ in order to quantify the uncertainty in input parameters, and reduce the discrepancies between code output and experimental data, whereas [1] investigated the use of Deep-GP to substitute a computationally expensive BE model and perform advanced UQ tasks.

However, DL and ML models are 'black-box' models that do not contain any structure based on physics and, thus, are difficult to interpret and it is difficult to quantify their output uncertainty. This limits the credibility of their use in safety assessment and licensing, especially in a tightly regulated industry such as the nuclear industry [31]. Moreover, the training of DL models can require large amounts of data, which may be unfeasible to obtain experimentally or by simulation with the computationally expensive BE models used in nuclear technology. In this regard, desirable characteristics of a surrogate model are *i*) accuracy with respect to the original BE model; *ii*) consistency with the physical laws governing the simulated phenomena; *iii*) capability to be trained in low data regime, when only a few simulations are available for the training; *iv*) the ability to deal with a large amount of data, if many simulations of long transients were available; and *v*) capability to run tens of thousands of simulations in a reasonable computational time.

Physics Informed Neural Networks (PINNs) allow integrating physics laws and domain knowledge, to yield more physically consistent outputs, and assure sound generalization capabilities [32]. When no data are available, PINNs use specific governing Ordinary Differential Equations (ODEs) or Partial Differential Equations (PDEs), and the related boundary conditions and initial conditions, to mimic a process. On the other hand, when simulation or experimental data are available, PINNs can integrate them with physics knowledge and governing physics laws (usually in the form of PDEs) [33]. With regards to the above, PINNs are expected to *a*) provide accurate solutions, intrinsically consistent with the physical laws describing the phenomena of interest; *b*) limit underfitting (i.e., the situation where the model fails to both learn the training data and generalize), thanks to the integration of the physical laws in the training; *c*) avoid overfitting (i.e., the situation where the model performs well on the training data but is not able to generalize and accurately predict previously unseen data); *d*) be able to handle both situations of small or large amount of data for the network training. PINNs have been used in various domains such as Computational Fluid Dynamic (CFD) [20], thermal-hydraulics [34], nuclear physics [35] and solid mechanics [36]. [20] developed a PINN that, without relying on any simulation data, uses the initial and boundary conditions to solve Navier–Stokes equations [37]. encoded hydro-fracture physical knowledge, in the form of PDEs, in order to develop a surrogate model to optimize hydro-fracture geometries during construction [35]. proposed the use of PINNs to solve Point Kinetics Equations (PKEs) with temperature feedback [34]. used PINNs in the thermal design of power electronics and for inverse problems of heat transfer, given forced and mixed convection. In Ref. [36], a PINN was used to incorporate the momentum balance and constitutive relations in solid mechanics, for linear and nonlinear elasticity applications. Despite the use of PINNs to solve specific PDEs or ODEs, their application to surrogate multi-physics BE models representing the overall response of complex systems, such as nuclear reactors, has not been investigated yet. Moreover, to the best of the authors' knowledge, most of PINN applications in the nuclear domain solve PDEs without relying on any simulation data and the knowledge therein, and none of them considers the use of partial domain knowledge (only a part of the governing PDEs) in a low data regime.

This work describes the use of PINNs for surrogate modeling of accidental scenarios in NPPs. PINN, including partial domain knowledge, is compared with its DNN counterpart. Moreover, the influence of the numerosity of the training data points is investigated to highlight the advantages of PINNs when a small amount of data is available. Also, the capabilities of the PINN to avoid overfitting and underfitting are explored. The results are validated by considering the simulation of the Loss of Heat Sink (LOHS) accidental scenario in a Nuclear Battery (NB), which is a unique class of transportable, plug-and-play microreactors [38].

The contributions of the work can be summarized as follows:

- It proposes and investigates the use of PINNs as surrogate models of multi-physics BE models of complex systems.
- It proposes the use of PINNs with partial domain knowledge in a low data regime.
- It investigates the influence of the amount of training data on PINNs and compares the results with DNNs.
- It develops an accurate, physics-consistent surrogate model of the LOHS accidental scenario in a NB.

The remainder of the paper is organized as follows: Section 2 presents the case study of the nuclear battery; Section 3 describes the developed PINN; Section 4 presents the results and the comparison with Deep NNs; finally, Section 5 draws some conclusions and points to potential future lines of work.

2. Case study: LOHS in a nuclear battery

Section 2.1 describes the considered nuclear battery design, Section 2.2 presents the LOHS accident scenario and Section 2.3 illustrates the BE model used to simulate the behavior of the NB during the LOHS.

2.1. Nuclear battery design

NBs are standardized plug-and-play micro-reactors that generate 1–20 MW of heat and/or electricity, which can provide low-carbon energy for various applications, including chemical processes, manufacturing, desalinition of water, hydrogen, food production, ship propulsion, to mention a few. They are envisioned as reliable, flexible, resilient, semi-autonomous, decentralized, affordable and transportable installations [38]. The NB design considered here is a 5 MW (thermal) high-temperature heat pipe reactor designed at MIT. Fig. 1 shows the core cross-section and the hexagonal pitch of the Fuel Assemblies (FAs). The core is characterized by solid monolithic blocks of graphite with three types of channels that encapsulate the yttrium hydride (YtH) neutron moderator, the fuel and the heat pipes. The core is surrounded by a neutron reflector, which accommodates Control Drums (CDs) to provide reactivity control during normal operation. Core and reflector are encased in a Canister Containment Subsystem (CCS), which is itself surrounded by a Reactor Vessel Auxiliary Cooling System (RVACS) for decay-heat removal, whereas Shutdown Rods (SRs) provide rapid shutdown. During normal operation, heat pipes remove the heat from the core to the heat exchanger and, then, to the Power Conversion Unit (PCU). Finally, the NB is designed to operate autonomously: The SRs provide rapid reactivity control, whereas a remote operator monitors operations and actuates a manual shutdown, should the protection system fail to do so. More details of the design are given in Ref. [39].

2.2. Loss of Heat Sink accident scenario

The scenario considered in this work is the unprotected Loss of Heat Sink (LOHS). This event leads to a sudden loss of heat transfer to the secondary side PCU, and to the inability to control the reactivity with the CDs and SRs. Thus, the reactivity control relies on the Reactivity Feedback (RF) only, whereas the heat is removed radially through the core by conduction and eventually by the RVACS.

2.3. NB simulation model

The BE model has been developed using the RELAP-5 3D code. Heat Structures (HSs) are used to model the system elements and the heat transfer among structures is modeled using the gap conductance approach. Fig. 2 represents the general high-level nodalization of the NB. Heat is transferred among the various core regions to the RVACS, which is represented using a pipe element with Time-Dependent Volumes (TDV) to model the air source and sink.

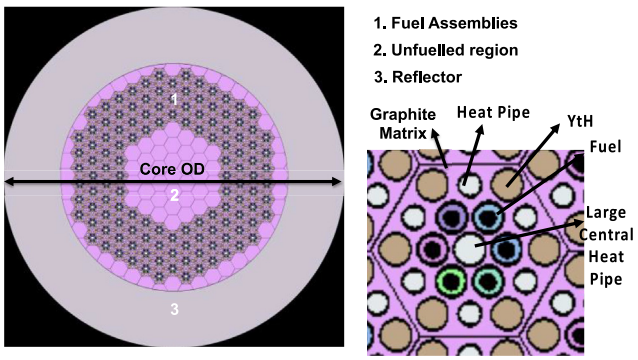

Fig. 1. Core cross-section and hexagonal pitch of the FAs.

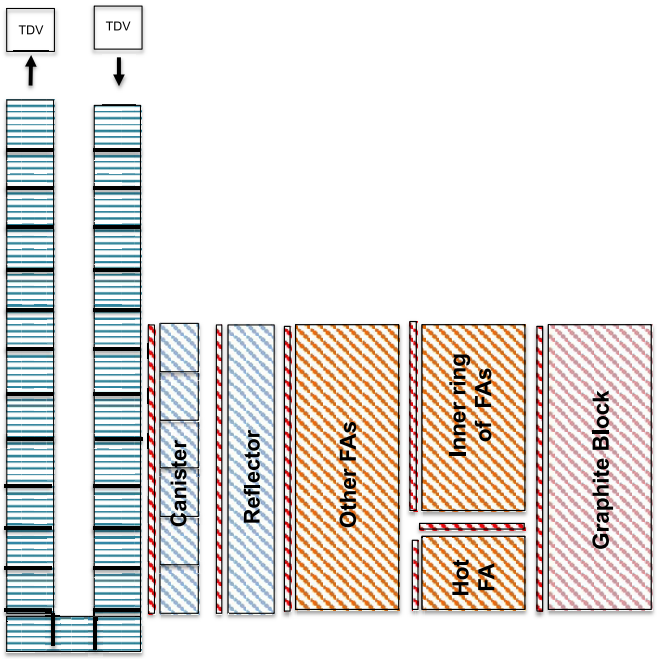

Fig. 2. High-level nodalization of the NB model.

Table 1 gives the list of the model input parameters considered in the analysis. Notice that the values and distributions have been taken according to Ref. [40]. In particular, the nominal values have been: *a*) obtained from literature data, for example for the thermal conductivity and wall roughness, *b*) computed based on literature models, such as for the gap conductance, or *c*) assessed via dedicated reactor core analyses, for example for the reactivity feedback coefficient. The ranges of variability have been conservatively considered large enough to account for design changes, for example of the RVACS flow area and initial power, and for the lack of knowledge on the actual values, for example of the thermal conductivity and the gap conductance. Notice also that initial power, RVACS flow rate and input air temperature can be considered initial conditions of the accident scenarios. Further details about the modeling strategies and the model parameters can be found in Ref. [40].

The unprotected LOHS is simulated through the following phases: 1) an initial normal operation condition, 2) the sudden loss of heat transfer from the heat pipes to the heat sink and 3) the simulation of the transient for 5,000 s. The key model output variables considered here are: fuel average temperature in the hot FA, net reactivity, peak cladding temperature and RVACS heat removal rate (or power). Fig. 3 shows an example of the behavior of the fuel temperature in the hot FA and of the reactivity during the first 2,500 s of a transient. The temperature increases following the occurrence of the LOHS and reaches its maximum value after 150 s. The increase of the temperature leads the RF to shut down the NB by decreasing reactivity. Notice that the objective of the work is to investigate the use of PINNs for the surrogate modeling of a generic accident scenario, and, for the sake of simplicity, the unprotected LOHS accident is here modeled without assuming additional initiating events, control actions and degrees of severity. A comprehensive analysis of other accidental scenarios can be found in Ref. [40], within a safety analysis framework.

Multiple simulations by the PINN surrogate model can be then used to perform uncertainty quantification in support to BE safety evaluations. As an example, Fig. 4 shows the probability density

Table 1
Simulation model input parameters and ranges of variation.

Parameter	Nominal Value [Range] [Distribution]
1. NB Power	5 MW [∓15%] [Uniform]
2. Reactivity feedback coefficients	Heat pipes: 5.6 pcm/(kg/m3) Fuel coefficient: 1.5 pcm/K Moderator coefficient: +0.1 pcm/K Range: [∓30%] [Uniform]
3. Graphite thermal conductivity	Temperature dependent function [∓10%] [Uniform]
Gap conductance between:	10^4 W/m2-K [5×10^3-10^5] [LogUniform]
4. Fuel and cladding	
5. Cladding and graphite	
6. Adjacent fuel assemblies	
7. Gap conductance between canister and reflector	70 W/m^2-K [40–120] [Uniform]
8. RVACS flow area	0.3 m [0.2–0.4] [Uniform]
9. RVACS input air temperature	300 K [270–320] [Uniform]
10. RVACS wall roughness	10^{-4} [10^{-3}-10^5] [LogUniform]

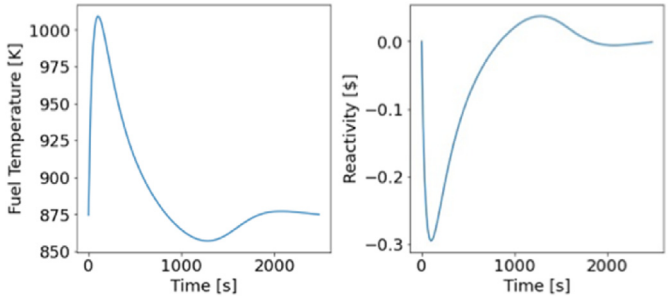

Fig. 3. Example of the behavior of the fuel temperature in the hot FA and the reactivity during a transient.

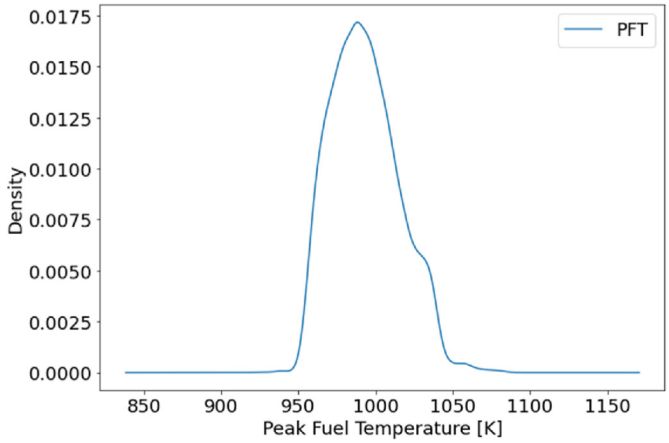

Fig. 4. Peak Fuel Temperature distribution.

function (pdf) of the Peak Fuel Temperature (PFT), which is the most safety-relevant parameter among the four model outputs. The results are obtained with the PINN described in Section 4, trained using 180 simulations.

3. Physics informed neural network-based surrogate model

The objective of the proposed surrogate model is to mimic the BE simulation model. Let $\overline{X} = [x_1, x_2, \ldots, x_d] \in R^d$ be an input parameter vector of the BE model, where each x_i corresponds to one of the parameters in Table 1. The BE model computes, at every time step t, the values of the output variables $\overline{y_t} = [y_{t,1}, y_{t,2}, \ldots, y_{t,N}] \in R^N$, where $y_{t,i}$ is the observation of the i-th output variable, y_i, at time t. The simulation time is here considered relative to the beginning of the LOHS accident scenario. Eventually, the BE model provides the overall transient simulation $Y = [\overline{y_1}, \overline{y_2}, \ldots, \overline{y_T}] \in R^{TxN}$.

Given the large dimensionality of the output Y, particularly when long transients are simulated, traditional surrogate models, such as SVMs, GPs, PCE and simple NNs, are not suitable for learning the input-output relations between $\overline{X}$ and Y [10]. Dimensionality reduction techniques can be used to reduce the output dimension. Another solution is reconstructing only a limited number of critical values of the output parameters, e.g., the maximum temperatures reached by the fuel during the simulation, $y_{t^*,i}$. Thus, the surrogate models learn the mapping between $\overline{X}$ and $y_{t^*,i}$. However, the above-mentioned solutions are not particularly accurate and do not replicate the overall transient, which is particularly important for inverse UQ [10] and prognostics and health management [41]. In light of this, advanced DL models can handle the large dimensionality of the BE output and be used to mimic the overall BE simulations [28]. In this work, we consider the use of PINNs to surrogate the simulation of the overall BE model transient Y. In particular, given the input vector $\overline{X}$, the PINN is trained to map, for each time step t, the pair $(\overline{X}, t)$ with the corresponding $\overline{y_t}$. Table 2 reports the considered input and output parameters.

3.1. Physics informed neural network

Fig. 5 displays the PINN architecture considered in this study. The main difference from a traditional DNN is the loss function. DNNs train on losses which are functions of the difference between the actual values of the output vector, $\overline{y_t}$, and the values predicted by the network, $\overline{y_t}^*$. Instead, PINNs make use of loss functions that combine traditional losses with a physics-based loss, which forces the network to satisfy domain knowledge and constraints, usually in the form of PDEs.

In this work we use the point kinetic equation and, in particular, the following reactivity feedback equation [42] as a physics-constraint for the PINN:

$$\frac{\partial R}{\partial t} = \alpha \bullet \frac{\partial T_f}{\partial t} \tag{1}$$

The equation gives a simplified representation of the time-dependent behavior of reactivity feedback in response to fuel temperature changes during a transient, where the reactivity coefficient α describes the variation of the reactivity that corresponds to the variation of the average fuel temperature in the core of 1 K. The equation considers only the response of the reactivity due to

Table 2
Simulation model input and output parameters.

Input parameter		Output parameters	
x_1	NB power (P_{NB})	y_1	Average fuel temperature (T_f)
x_2	Fuel temperature reactivity feedback coefficient (α)	y_2	Net reactivity (R)
x_3	Graphite thermal conductivity (K_g)	y_3	Peak cladding temperature (T_c)
x_4	Gap conductance between fuel and cladding (GC_{fc})	y_4	RVACS power (P_R)
x_5	Gap conductance between adjacent fuel assemblies (GC_{fa})		
x_6	Gap conductance between cladding and graphite (GC_{cg})		
x_7	Gap conductance between canister and reflector (GC_{NB})		
x_8	RVACS flow area (A_R)		
x_9	RVACS input air temperature (T_R)		
x_{10}	RVACS wall roughness (k_R)		

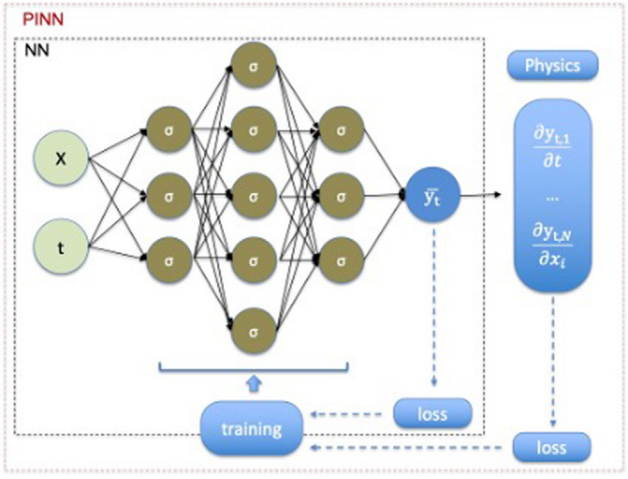

Fig. 5. Architecture of the considered PINN.

temperature variations and ignores other effects and the reactivity manipulation via CDs and CRs. Notice that, the present work aims at investigating the performance of PINNs with partial knowledge, which is here considered as the implementation into the network loss function of only a small subset of the physical laws governing the process under analysis. During the considered LOHS accident scenario, the NB is shut down only through the RF, which reduces reactivity and, consequently, the reactor power, as the fuel temperature increases. Thus, among the thousands of physical equations and correlations comprised in the RELAP-5 3D code, the above-mentioned reactivity feedback equation, which drives the dynamic of the NB during the LOHS, is here selected to investigate the potential of using PINNs with partial knowledge.

The corresponding physics-based loss for a batch of data of size M is:

$$loss_P = \frac{\sum_{i=1}^{M}\left(\frac{\partial y_{t,2}^{*}}{\partial t} - x_{2,i} \bullet \frac{\partial y_{t,1}^{*}}{\partial t}\right)^2}{M} \tag{2}$$

Whereas, the traditional loss is computed using the Mean Squared Error (MSE):

$$MSE(\overline{y}_t, \overline{y}_t^{*}) = \frac{\sum_{i=1}^{N}\left(y_{t,i} - y_{t,i}^{*}\right)^2}{N} \tag{3}$$

Then, the overall loss function is the combination of the physics-based loss and the traditional loss:

$$Loss = loss_P + MSE \tag{4}$$

Notice that, given a PINN, its DNN counterpart is the network with the same architecture, but which uses only the traditional loss and not also the physics-based loss. The details of the architecture of the PINN and its DNN counterpart, and the hyperparameters selected following a trial-and-error procedure, are given in Table 3. The data have been scaled using a traditional min-max scaler and the networks have been trained on 70%–80% of the available simulations, validated on 10%–20% to calibrate the hyperparameters, and tested on the remaining 10%–20% simulations. The training has been performed for 2000 epochs and a callback method has been used to select the network performing the best on the validation dataset. For the optimization, the first-order gradient-based optimizer Adam, based on adaptive estimates of lower-order moments, has been selected for its straightforward implementation and computational efficiency [43]. Finally, Equation (5) gives the considered Exponential Linear Unit (ELU) activation function, whose characteristic of decreasing the bias by driving the mean activation towards zero has been proven effective, compared to other non-linear activation functions [44].

$$\begin{cases} elu(x) = \alpha(\exp(x) - 1) & if\ x \le 0 \\ elu(x) = x & if\ x > 0 \end{cases} \tag{5}$$

where α is here set equal to 1.

4. Results

In this work, we implement and compare a PINN with its DNN counterpart to create a surrogate of the BE model simulations of an unprotected LOHS accidental scenario. We consider the development of surrogate models using different amounts of training data. In particular, we initially sampled 250 sets of input parameters using the Latin Hypercube Strategy (LHS) strategy. These sets are used by the RELAP5-3D BE model to simulate the corresponding transients. Among the 250 sets of input-output samples, 180 are used for the training of the PINN and the DNN, 30 for the validation of the results and 40 for the testing. Note that, given the complexity of the dataset and the nonlinear relationships driving the process, the accuracy of the networks is affected by the randomness in the initialization of the network weights and parameters. Thus, the PINN and the DNN are trained, validated and tested considering 10 different random seeds. The networks are, then, analyzed considering the MSE and, in particular, comparing the minimum, the maximum, the mean and the median MSE values among the 10 training seeds. Then, we reduce the number of training trajectories to 50 and, eventually, to 10 samples, considering the same 30 sets for validation and 40 for testing.

Table 4 reports the results for the PINN and DNN trained with 180 samples. Note that the relative errors of the four model outputs

Table 3
PINN and DNN architecture details and hyperparameters.

Number of layers	6
Number of neurons per layer	128, 64, 32, 32, 14, 4
Optimizer	Adam
Activation function	Elu
Batch size	128
Learning rate	0.001
Maximum training epochs	2000

are comparable. Thus, for the sake of conciseness, only the results and errors related to the fuel temperature, which is the most safety-relevant parameter in our application, are displayed here. Notice that the best PINN (the one with the minimum MSE) outperforms the best DNN, both for the validation set and for the test set. In particular, the PINN normalized MSE on the validation set is 8.3×10^{-6} and corresponds to a rescaled MSE of 0.4 K^2, which is 30% lower than the DNN one. Moreover, the worst PINN is characterized by an MSE that is two orders of magnitude lower than that of the worst DNN, in which the large error is due to network overfitting. DNN underfitting is also shown by the mean MSE, which is much larger than that of the PINN. Similar considerations can be derived for the test set. Fig. 6 shows the MSE on the train and validation sets at each of the training epochs for the DNN and PINN. The comparison highlights the capability of the PINN to avoid overfitting and provide more regularized training than DNN. Finally, it is worth noticing that the computational time required to train the PINN increases with respect to the time required by the DNN, from 363 s to 2345 s on an Intel core (TM) i7-4790 CPU@ 3.6 GHz, 16 GB RAM. On the other hand, given a trained model, the computational time required to simulate the entire transient, which is the term of reference for the performance of a surrogate model, is a few seconds for both the PINN and the DNN.

The results are, then, compared with a DNN which integrates the l2 regularization in the loss function, a widely used technique to avoid overfitting [45]. The l2 regularization term has been set equal to 10^{-5} following a trial-and-error procedure considering values in the range [10^{-1},10^{-6}]. Fig. 7 shows the MSE values on the training and validation datasets at each of the training epochs, and highlights the capability of the l2 regularizer to avoid overfitting. However, the MSE is larger than that of the DNN which does not use regularization. In this light, it is worth underlining the capabilities of the PINN to avoid overfitting and reducing the MSE, at the same time.

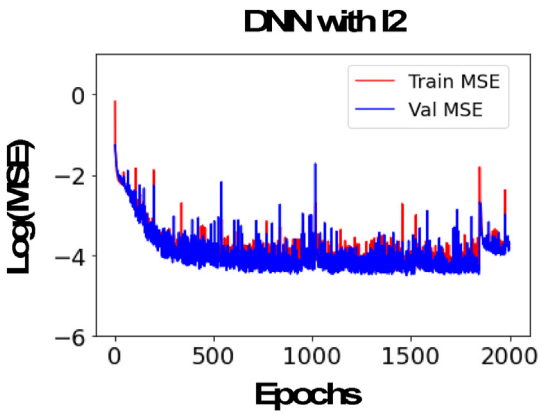

Fig. 7. *Comparison of the MSE values for training and validation at each of the training epochs for the DNN* with l2 regularizer.

Table 5 reports the results for the PINN and DNN trained with 50 samples, drawn with the LHS strategy. The best PINN outperforms the best DNN both for the validation set and for the test set. Also, the worst PINN has a lower MSE than the worst DNN, and the mean and median values among all the training seeds are better for the PINN than for the DNN. This reflects the good regularization properties of the PINN and its ability to avoid underfitting.

Finally, Table 6 reports the results for the PINN and DNN trained with 10 samples, drawn with the LHS strategy. Note that the best PINN (the one with the minimum MSE) again outperforms the best

Table 4
Results for the PINN and DNN trained with 180 samples.

		DNN	PINN
Validation set	Min MSE (rescaled)	1.18×10^{-5} (0.58)	8.27×10^{-6} (0.41)
	Max MSE (rescaled)	0.00421 (208.55)	3.63×10^{-5} (1.80)
	Mean MSE (rescaled)	0.000449 (22.22)	1.72×10^{-5} (0.85)
	Median MSE (rescaled)	1.70e-05 (0.84)	1.36×10^{-5} (0.67)
Test set	Min MSE (rescaled)	2.20e-05 (1.09)	1.69×10^{-5} (0.84)
	Max MSE (rescaled)	0.00396 (196.11)	9.21×10^{-5} (4.56)
	Mean MSE (rescaled)	0.000436 (21.58)	3.63×10^{-5} (1.80)
	Median MSE (rescaled)	3.10×10^{-5} (1.54)	2.64×10^{-5} (1.31)

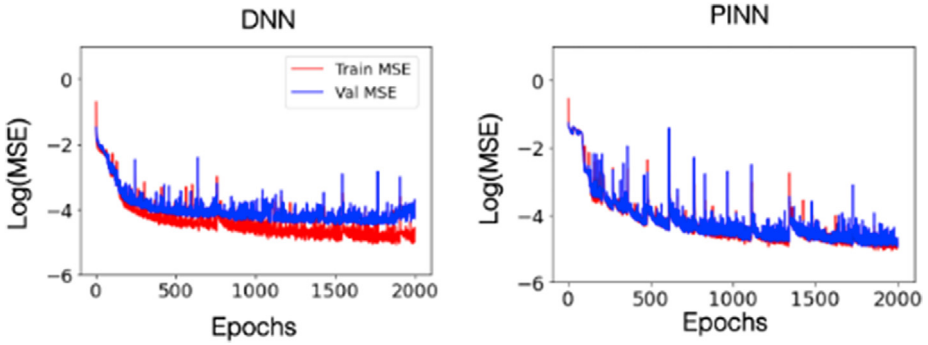

Fig. 6. Comparison of the MSE values for training and validation at each of the training epochs for the DNN and PINN.

Table 5
Results for the PINN and DNN trained with 50 samples.

		DNN	PINN
Val set	Min MSE (rescaled)	0.00265 (12.03)	0.000243 (8.21)
	Max MSE (rescaled)	0.0163 (73.77)	0.00149 (70.15)
	Mean MSE (rescaled)	0.000601 (24.82)	0.00562 (19.36)
	Median MSE (rescaled)	0.000597 (19.84)	0.00401 (13.37)
Test set	Min MSE (rescaled)	0.000344 (17.05)	0.000268 (13.25)
	Max MSE (rescaled)	0.00175 (86.40)	0.00141 (69.75)
	Mean MSE (rescaled)	0.000759 (37.59)	0.000659 (32.64)
	Median MSE (rescaled)	0.000617 (30.51)	0.000517 (25.60)

Table 6
Results for the PINN and DNN trained with 10 samples.

		DNN	PINN
Val set	Min MSE (rescaled)	0.00265 (131.36)	0.00208 (103.10)
	Max MSE (rescaled)	0.0163 (809.17)	0.00820 (405.98)
	Mean MSE (rescaled)	0.00662 (328.07)	0.00393 (194.70)
	Median MSE (rescaled)	0.00597 (295.58)	0.00354 (175.25)
Test set	Min MSE (rescaled)	0.00467 (231.11)	0.00263 (130.04)
	Max MSE (rescaled)	0.141 (6957.77)	0.0118 (584.65)
	Mean MSE (rescaled)	0.0204 (1010.36)	0.00535 (264.64)
	Median MSE (rescaled)	0.00638 (315.81)	0.00434 (214.92)

DNN both for the validation set and for the test set. In particular, for the test set, the rescaled errors of the best PINN and DNN correspond to an absolute error of 10.4 K for the PINN and 15.1 for the DNN. Despite both the PINN and the DNN are characterized by larger errors than networks trained with more data, the MSE of the PINN is 44% lower than that of the DNN. This underlines the significant advantages of using PINNs in low data regimes. Moreover, the worst PINN is characterized by a much lower MSE than the worst DNN, which underfits the training data. This highlights, again, the capability of the PINN to avoid underfitting and provide more regularized training than DNN.

5. Conclusion

Developing more reliable, physics-consistent and accurate surrogate models is crucial for the safe licensing of novel designs of nuclear reactors. This work has developed a surrogate model of accidental scenarios in nuclear reactors using PINNs, which, embedding fundamental physics laws and domain knowledge in the networks, assures good generalization, and reliable results. In particular, considering the simulation of the LOHS accident scenario in a nuclear battery, a PINN is developed including domain knowledge, in the form of a differential equation, in the neural network. The PINN is compared with a traditional DNN and the results show that it *i)* outperforms traditional DNN, *ii)* reduces the network error, *iii) avoids* overfitting and underfitting, and *iv)* provides more regularized and consistent simulations. This highlights the benefit of including part of the governing fundamental laws in the neural network training. The influence of the numerosity of training data points has also been investigated. The results have highlighted the particularly significant advantages of adding physics domain knowledge in the PINNs when a small amount of data is available.

For future work, one direction lies in the development of different physics-informed deep learning network architectures, such as recurrent and LSTM networks. Also, since only one physic-law equation is considered in the developed PINN, the integration of other laws and additional physical knowledge will be investigated.

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.

Acknowledgements

The research has been funded by the “Progetto Rocca Post-Doctoral Fellowship”.

References

[1] M.I. Radaideh, T. Kozlowski, Surrogate modeling of advanced computer simulations using deep Gaussian processes, Reliab. Eng. Syst. Saf. 195 (October 2019) (2020), 106731, https://doi.org/10.1016/j.ress.2019.106731.
[2] F. D'Auria, C. Camargo, O. Mazzantini, The Best Estimate Plus Uncertainty (BEPU) approach in licensing of current nuclear reactors, Nucl. Eng. Des. 248 (2012) 317–328, https://doi.org/10.1016/j.nucengdes.2012.04.002.
[3] X. Wu, T. Kozlowski, Inverse uncertainty quantification of reactor simulations under the Bayesian framework using surrogate models constructed by polynomial chaos expansion, Nucl. Eng. Des. 313 (2017) 29–52, https://doi.org/10.1016/j.nucengdes.2016.11.032.
[4] R.K. Tripathy, I. Bilionis, Deep UQ: learning deep neural network surrogate models for high dimensional uncertainty quantification, J. Comput. Phys. 375 (2018) 565–588, https://doi.org/10.1016/j.jcp.2018.08.036.
[5] M.I. Radaideh, T. Kozlowski, Analyzing nuclear reactor simulation data and uncertainty with the group method of data handling, Nucl. Eng. Technol. 52 (2) (2020) 287–295, https://doi.org/10.1016/j.net.2019.07.023.
[6] P. Benner, S. Gugercin, K. Willcox, A survey of projection-based model reduction methods for parametric dynamical systems, SIAM Rev. 57 (4) (2015) 483–531, https://doi.org/10.1137/130932715.
[7] S. Chaturantabut, D.C. Sorensen, Nonlinear model reduction via discrete empirical interpolation, SIAM J. Sci. Comput. 32 (5) (2010) 2737–2764.
[8] C. Worrell, L. Luangkesorn, J. Haight, T. Congedo, Machine learning of fire hazard model simulations for use in probabilistic safety assessments at nuclear power plants, Reliab. Eng. Syst. Saf. 183 (February 2018) (2019) 128–142, https://doi.org/10.1016/j.ress.2018.11.014.
[9] J.E. Hurtado, Filtered importance sampling with support vector margin: a powerful method for structural reliability analysis, Struct. Saf. 29 (1) (2007) 2–15, https://doi.org/10.1016/j.strusafe.2005.12.002.
[10] G. Roma, F. Antonello, F. Di Maio, N. Pedroni, E. Zio, A. Bersano, C. Bertani, F. Mascari, Passive safety systems analysis: a novel approach for inverse uncertainty quantification based on Stacked Sparse Autoencoders and Kriging metamodeling, Prog. Nucl. Energy 148 (September 2021) (2022), 104209, https://doi.org/10.1016/j.pnucene.2022.104209.
[11] S. Yoon, M.J. Kim, S. Park, G.Y. Kim, Thermal conductivity prediction model for compacted bentonites considering temperature variations, Nucl. Eng. Technol. 53 (10) (2021) 3359–3366, https://doi.org/10.1016/j.net.2021.05.001.

[12] J.P. Yurko, J. Buongiorno, R. Youngblood, Demonstration of emulator-based bayesian calibration of safety analysis codes: theory and formulation, Sci. Technol. Nucl. Instal. 2015 (2015), https://doi.org/10.1155/2015/839249. Mcmc).
[13] B. Ebiwonjumi, C. Kong, P. Zhang, A. Cherezov, D. Lee, Uncertainty quantification of PWR spent fuel due to nuclear data and modeling parameters, Nucl. Eng. Technol. 53 (3) (2021) 715–731, https://doi.org/10.1016/j.net.2020.07.012.
[14] B. Sudret, C.V. Mai, Computing derivative-based global sensitivity measures using polynomial chaos expansions, Reliab. Eng. Syst. Saf. 134 (2015) 241–250, https://doi.org/10.1016/j.ress.2014.07.009.
[15] L. Puppo, N. Pedroni, F. Di Maio, A. Bersano, C. Bertani, E. Zio, A framework based on finite mixture models and adaptive kriging for characterizing non-smooth and multimodal failure regions in a nuclear passive safety system, Reliab. Eng. Syst. Saf. 216 (August) (2021), 107963, https://doi.org/10.1016/j.ress.2021.107963.
[16] N. Pedroni, E. Zio, An Adaptive Metamodel-Based Subset Importance Sampling approach for the assessment of the functional failure probability of a thermal-hydraulic passive system, Appl. Math. Model. 48 (2017) 269–288, https://doi.org/10.1016/j.apm.2017.04.003.
[17] X. Song, L. Lv, W. Sun, J. Zhang, A radial basis function-based multi-fidelity surrogate model: exploring correlation between high-fidelity and low-fidelity models, Struct. Multidiscip. Optim. 60 (3) (2019) 965–981, https://doi.org/10.1007/s00158-019-02248-0.
[18] L. Conner, C.L. Worrell, J.P. Spring, J. Liao, Machine learned metamodeling of a computationally intensive accident simulation code, Int. Conf. Nucl. Eng. Proc., ICONE 1 (2021) 1–6, https://doi.org/10.1115/ICONE28-66619.
[19] R. Abu Saleem, M.I. Radaideh, T. Kozlowski, Application of deep neural networks for high-dimensional large BWR core neutronics, Nucl. Eng. Technol. 52 (12) (2020) 2709–2716, https://doi.org/10.1016/j.net.2020.05.010.
[20] L. Sun, H. Gao, S. Pan, J.X. Wang, Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data, Comput. Methods Appl. Mech. Eng. 361 (2020), 112732, https://doi.org/10.1016/j.cma.2019.112732.
[21] A. Ayodeji, M.A. Amidu, S.A. Olatubosun, Y. Addad, H. Ahmed, Deep learning for safety assessment of nuclear power reactors: reliability, explainability, and research opportunities, Prog. Nucl. Energy 151 (August) (2022), 104339, https://doi.org/10.1016/j.pnucene.2022.104339.
[22] V. Ciriello, V. Di Federico, M. Riva, F. Cadini, J. De Sanctis, E. Zio, A. Guadagnini, Polynomial chaos expansion for global sensitivity analysis applied to a model of radionuclide migration in a randomly heterogeneous aquifer, Stoch. Environ. Res. Risk Assess. 27 (4) (2013) 945–954, https://doi.org/10.1007/s00477-012-0616-7.
[23] F. Shriver, C. Gentry, J. Watson, Prediction of neutronics parameters within a two-dimensional reflective PWR assembly using deep learning, Nucl. Sci. Eng. 195 (6) (2021) 626–647, https://doi.org/10.1080/00295639.2020.1852021.
[24] M.I. Radaideh, D. Price, T. Kozlowski, Modeling nuclear data uncertainties using deep neural networks, in: International Conference on Physics of Reactors: Transition to a Scalable Nuclear Future, PHYSOR 2020, 2020-March, 2020, pp. 2583–2590, https://doi.org/10.1051/epjconf/202124715016.
[25] F. Cadini, A. Gioletta, E. Zio, Improved metamodel-based importance sampling for the performance assessment of radioactive waste repositories, Reliab. Eng. Syst. Saf. 134 (2015) 188–197, https://doi.org/10.1016/j.ress.2014.10.018.
[26] D. Price, M.I. Radaideh, B. Kochunas, Multiobjective optimization of nuclear microreactor reactivity control system operation with swarm and evolutionary algorithms, Nucl. Eng. Des. 393 (January) (2022), 111776, https://doi.org/10.1016/j.nucengdes.2022.111776.
[27] H. Kim, J. Cho, J. Park, Application of a deep learning technique to the development of a fast accident scenario identifier, IEEE Access 8 (2020) 177363–177373, https://doi.org/10.1109/ACCESS.2020.3026104.
[28] S. Ryu, H. Kim, S.G. Kim, K. Jin, J. Cho, J. Park, Probabilistic deep learning model as a tool for supporting the fast simulation of a thermal–hydraulic code, Expert Syst. Appl. 200 (March) (2022), 116966, https://doi.org/10.1016/j.eswa.2022.116966.
[29] M. Ebad Sichani, J.E. Padgett, V. Bisadi, Probabilistic seismic analysis of concrete dry cask structures, Struct. Saf. 73 (June 2017) (2018) 87–98, https://doi.org/10.1016/j.strusafe.2018.03.001.
[30] R. Shrestha, T. Kozlowski, Inverse uncertainty quantification of input model parameters for thermal-hydraulics simulations using expectation–maximization under Bayesian framework, J. Appl. Stat. 43 (6) (2016) 1011–1026, https://doi.org/10.1080/02664763.2015.1089220.
[31] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, F. Herrera, Explainable Artificial Intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI, Inf. Fusion 58 (October 2019) (2020) 82–115, https://doi.org/10.1016/j.inffus.2019.12.012.
[32] G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, Nat. Rev. Phys. 3 (6) (2021) 422–440, https://doi.org/10.1038/s42254-021-00314-5.
[33] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. 378 (2019) 686–707, https://doi.org/10.1016/j.jcp.2018.10.045.
[34] S. Cai, Z. Wang, S. Wang, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks for heat transfer problems, J. Heat Tran. 143 (6) (2021), https://doi.org/10.1115/1.4050542.
[35] E. Schiassi, M. De Florio, B.D. Ganapol, P. Picca, R. Furfaro, Physics-informed neural networks for the point kinetics equations for nuclear reactor dynamics, Ann. Nucl. Energy 167 (2022), 108833, https://doi.org/10.1016/j.anucene.2021.108833.
[36] E. Haghighat, M. Raissi, A. Moure, H. Gomez, R. Juanes, A physics-informed deep learning framework for inversion and surrogate modeling in solid mechanics, Comput. Methods Appl. Mech. Eng. 379 (2021), 113741, https://doi.org/10.1016/j.cma.2021.113741.
[37] Y. Lu, B. Wang, Y. Zhao, X. Yang, L. Li, M. Dong, Q. Lv, F. Zhou, N. Gu, L. Shang, Physics-informed surrogate modeling for hydro-fracture geometry prediction based on deep learning, Energy 253 (2022), 124139, https://doi.org/10.1016/j.energy.2022.124139.
[38] J. Buongiorno, B. Carmichael, B. Dunkin, J. Parsons, D. Smit, Can nuclear batteries Be economically competitive in large markets? Energies 14 (14) (2021) 4385, https://doi.org/10.3390/en14144385.
[39] F. Antonello, J. Buongiorno, E. Zio, A methodology to perform dynamic risk assessment using system theory and modeling and simulation: application to nuclear batteries, Reliab. Eng. Syst. Saf. 228 (August) (2022), 108769, https://doi.org/10.1016/j.ress.2022.108769.
[40] F. Antonello, J. Buongiorno, E. Zio, Insights in the safety analysis of an early microreactor design, Nucl. Eng. Des. 404 (September 2022) (2023), 112203, https://doi.org/10.1016/j.nucengdes.2023.112203.
[41] E. Zio, Prognostics and Health Management (PHM): where are we and where do we (need to) go in theory and practice, Reliab. Eng. Syst. Saf. 218 (October 2021) (2022), https://doi.org/10.1016/j.ress.2021.108119.
[42] A.F. Henry, The application of reactor kinetics to the analysis of experiments, Nucl. Sci. Eng. 3 (1) (1958) 52–70, https://doi.org/10.13182/nse58-1.
[43] D.P. Kingma, J.L. Ba, Adam: a method for stochastic optimization, 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf.Track Proc. (2015) 1–15.
[44] D. Pedamonti, Comparison of Non-linear Activation Functions for Deep Neural Networks on MNIST Classification Task, vol. 3, 2018. http://arxiv.org/abs/1804.02763.
[45] C. Cortes, G. Research, N. York, L 2 regularization for learning kernels, in: Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, 2004, pp. 109–116.