Journal of Industrial Information Technology and Application *ISSN(Online): 2586-0852*
Vol.9 No. 2

Data-Driven Surrogate Model for Predicting 2D Assembly-wise Power Distribution Changes

Jung-seok Kwon, Tong-kyu Park), and Sung-kyun Zee*

Nuclear Computational Science Group. FNC Technology, Yongin-si, Korea

Abstract. This study presents a data-driven neural network surrogate model for predicting assembly-wise power distribution changes in the core of an i-SMR. The neural network was trained using a dataset generated by ASTRA, a nodal diffusion code. A convolutional neural network (CNN)-based architecture was designed to predict power distributions at target load, based on current power distributions, ramp rate, depletion time, current load and target load value. The model achieved a mean relative error of 0.833% and a peak power prediction error of 0.768%, demonstrating its ability to effectively predicting power distributions under varying load.

Keywords; Neural network; power distribution; i-SMR; surrogate model; load variation

Cite this paper as : Jung-seok Kwon, Tong-kyu Park, and Sung-kyun Zee (2025) "Data-Driven Surrogate Model for Predicting 2D Assembly-wise Power Distribution Changes", Journal of Industrial Information Technology and Application, Vol. 9. No. 2, pp. 1075 - 1080

1. Introduction

Various physics simulation approaches based on artificial neural networks have been proposed. A common approach is data-driven supervised learning, which is the

* Corresponding author: tongkyu@fnctech.com
Received: Feb. 17. 2025 Accepted: Apr. 24. 2025 Published: Jun. 30. 2025

major technique used with neural networks. Another well-known technique is the Physics-Informed Neural Network (PINN), introduced by M. Raissi [1], which employs the differential equation as loss calculation functions in the network. Several methodologies have been developed to solve time-variant state transition problem using neural networks. These models have been used in a variety of applications, including weather forecasting, collision problems, and thermos-dynamics.

Several approaches have been developed to adapt to predict time variant state transition, such as qualitative simulation, the roll-out technique, and time series models.

First, qualitative simulation [2] has been used to predict state transitions, especially in cases when precise quantitative input data is difficult to obtain. However, this approach has a disadvantage in that it struggles to predict the state with exact quantities. Additionally, unlike the original approach, which describes state transitions using predicates, the AI-based qualitative simulation approach is closer to the roll-out technique, utilizing AI for sequential state predictions.

Second, the roll-out technique uses current state information to predict the next state at a future time step. The projected next state is then fed back into the neural network to predict the following time step, and this process repeats, hence the name “roll-out technique.” For example, in weather forecasting, Google’s research team developed 'GraphCast’[3], which predicts the weather 6 hours in advance by combining weather data from the previous 6 hours with current data.

Last, time series models use sequential data to predict future values based on past observations, capturing temporal patterns and dependencies. Techniques like sliding windows are applied to analyze trends over time, such as in weather forecasting or stock price prediction, where past data is essential for forecasting upcoming conditions.

In this paper, we employ the roll-out technique to predict the next time step’s core power distribution based on the given current power distribution and the target load value to be achieved.

2. Research Methods

*A. **Data Preparation***

The dataset for training the neural network was generated using the nodal diffusion design code, ASTRA, with design features referenced from i-SMR. The in-core loading pattern (LP) was fixed, and its layout is shown in Figure 1. In total, five fuel batches were used in LP, each with fuel rods containing 4% U-235 concentration. The batches differ in the number of gadolinium burnable absorbers.

ASTRA's restart function was employed to continue the calculations from the initial input file. The subsequent file maintained the same input conditions as the initial file, except for load value and depletion time. This subsequent file was then used to calculate the power distribution at the target load.

A05	A03	A04	A02	A01
28 Gd	20 Gd	24 Gd	28 Gd	16 Gd
A03	A03	A05	A04	A02
20Gd	20 Gd	28 Gd	24 Gd	28 Gd
A04	A05	A04	A02	A01
24 Gd	28 Gd	24 Gd	28 Gd	16 Gd
A02	A04	A02	A01	
28 Gd	24 Gd	28 Gd	16 Gd	
A01	A02	A01		
16 Gd	28 Gd	16 Gd		

Figure 1. Loading Pattern

The initial file contains the assigned initial load value, and the data generation algorithm determine the next load value using the selected ramp rate and depletion time value. For example, with a negative ramp rate of 5% per minute, an initial load value of 100%, and a depletion time of 10 minutes, the next file's load value becomes 50%.

There were five ramp rate options: 1%, 2%, 3%, 4%, and 5% per minute, with depletion time expressed as an integer value in minutes. For ramp rates, both ramp-up and ramp-down cases were considered. When constructing the dataset, a value of -1 was multiplied for ramp-down cases (e.g., -4%/min for rampdown). The load limit was constrained between 100% and 20%. The xenon option was set to transient mode because the power changes occurred over a short period, and xenon had a substantial impact on power distribution. A total of 14,513 data points were generated.

The input data for the neural network model includes the 2D assembly-wise power distribution (with a symmetric quadrant layout) at the current time step, ramp rate, depletion time, current load, and target load. The target data for the neural network to predict is the 2D assembly-wise power distribution at the next time step.

B. ***Neural Network Model Configuration***

The neural network consisted of convolutional neural network (CNN) [4] layers and a fully connected layer (referred to as Dense Layer in Figure 2). The main input took assembly-wise power distribution at the current time step, while the sub-input received a 1x4 shaped array of ramp rate, depletion time, current load, and target load. These two inputs were then combined during the training process, and the network predicted the final target power distribution. The structure of the neural network is shown in Figure 2.

Various techniques have been developed to improve the performance of CNN, and the residual block[5] is one of them. In this structure, the input tensor is passed through two convolutional layers with different kernel sizes, and their outputs are then combined before being returned. The structure of the residual block can be seen in Figure 3.

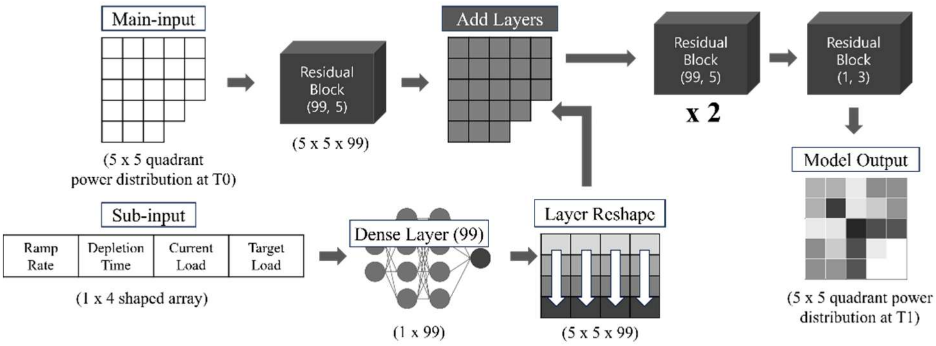

Figure 2. Neural Network Model Configuration

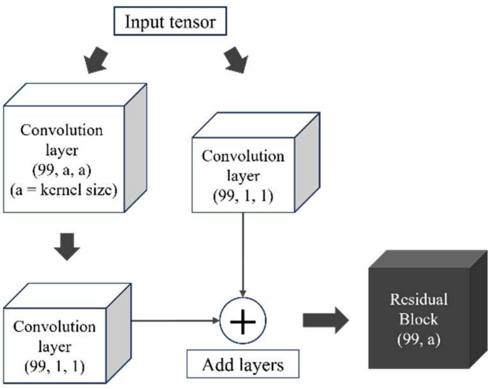

Figure 3. Residual Block

3. Results

Table 1. Model Prediction Errors

	Relative Error (%)
Mean Error	0.833%
Max Error	4.95%
Mean Error (Peak Power)	0.768%
Max Error (Peak Power)	2.84%

The total number of data samples was 14,513, with 12,407 used for training, and 1,452 used to evaluate neural network performance. The relative error was obtained by comparing the 1,452 values predicted by the model to the test data. For peak power, the error was measured between the highest power value at the lattice point in the original assembly-wise power distribution and the value predicted by the neural network.

4. Conclusion

This study validated a neural network-based methodology for predicting power distribution at target load values. First, our previous study successfully predicted pin-wise 2D power distributions and assembly-wise 3D power distributions. By combining the methodology from previous study with the current results, more comprehensive core power distributions could be obtained. Second, incorporating additional factors such as control rod positions, xenon distributions, and the location and number of burnable poison rods during training could significantly improve prediction accuracy. Finally, future research will focus on establishing and validating a methodology that provides complicate load variation histories based on operational cycles as input, rather than the simple ramping up or down used in this study.

Acknowledgment

This work was supported by the Innovative Small Modular Reactor Development Agency grant funded by the Korea Government (MOTIE) (RS-2023-00259289).

References

[1] M. Raissi, et al., "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," Journal of Computational physics, vol. 378, pp.686-707, (2019).

[2] B. Kuipers “Qualitative Simulation,” Artificial Intelligence, 29, pp. 289-338, (1986).

[3] Remi Lam et al., “Learning skillful medium-range global weather forecasting,” Science, vol. 382, pp. 1416-1421, (2023).

[4] Y. LeCun, et al., "Backpropagation Applied to Handwritten Zip Code Recognition," in Neural Computation, vol. 1, no. 4, pp. 541-551, (1989).

[5] He, Kaiming, et al. "Deep residual learning for image recognition," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, pp. 770-778, (2016).