

# A Comparative Study of Neural Network-based Core Analysis - Surrogate Models to Reduce Prediction Error of Axial Shape Index

Jungseok Kwon<sup>a</sup>, Tongkyu Park<sup>a\*</sup>, Sung-kyun Zee<sup>a</sup>  
<sup>a</sup>FNC Technology, Headquarters, 13 Heungdeok 1-ro,  
Giheung-gu, Yongin-si, Gyeonggi-do, 16954, Republic of Korea  
<sup>\*</sup>Corresponding author: tongkyu@fnctech.com

**\*Keywords :** nuclear core analysis, neural network, surrogate model, ASI (axial shape index), i-SMR (innovative small modular reactor)

## 1. Introduction

With the rapid advancement and widespread adoption of artificial intelligence (AI), the number of applications for AI-based approaches in nuclear engineering is steadily increasing. Particularly for nuclear core analysis, several AI-based surrogate models were proposed using data from the results of core design code or data constructed based on the physical equation. Examples of these models include the reconstruction of in-core power distribution based on the signals from neutron detectors[1], a core analysis surrogate model trained on data calculated from core design codes[2], and others.

In this study, we compared the prediction errors of neural network models trained on the calculation results of the design code, distinguished by data shape and model structure. As a baseline for comparison, we employed a convolutional neural network (CNN)[3]-based model. This model has demonstrated the effectiveness of AI-based surrogate models in predicting core parameters and power distribution at the beginning of the cycle (BOC) [2][4].

However, when estimating the axial shape index (ASI), a crucial metric used in flexible operation, relatively high errors were observed compared to other parameters. The base model attempted to predict the scalar value ASI directly from the input loading pattern data. Since ASI represents the axial power difference between the upper and lower regions of the core, it compresses three-dimensional information into a single value. This compression likely leads to information loss, making it inherently difficult for the model to make accurate predictions.

To improve accuracy, this study explored three approaches: (1) providing additional information related to reactor core control, (2) applying a fine-tuning strategy, and (3) recalculating ASI from the predicted 3D power distribution instead of using ASI directly as a target data. The effectiveness of these methods was evaluated by comparing them with the model used in the previous study.

## 2. Methods

In this section, we describe the methodology used to compare the baseline model with four additional improved models. The first part discusses how the dataset is constructed, including its shape and features. The second part presents the input/output shapes and architectures of the models. Additionally, we provide specifications for the computational server used to train the models.

### 2.1 Data Preparation

The models in this study employed a common quadrant core loading pattern as input data, but the loading patterns were processed differently depending on the architecture of each model. The loading patterns were constructed by randomly placing fuel assemblies. Each assembly differs in terms of fuel and burnable absorber concentration, number of burnable absorber rods, and their configuration. Based on a total of 37,532 loading patterns, core analysis was conducted at the Beginning of Cycle (BOC) using an in-house code, and the results were obtained.

The common quadrant core loading pattern can be represented in both 2D and 3D forms. The data shape in the 2D representation is a 5 x 5 matrix, and there is only one axial plane. In the 3D model, the composition of fuel assemblies in the loading pattern remains constant throughout the axial direction, and the included data information is identical to that of the 2D representation. However, it consists of 24 axial planes, and at the topmost plane, there is a cutback region composed solely of rods enriched with 4 w/o U-235. Consequently, the data shape for the 3D representation is 24 x 5 x 5.

The loading pattern was originally constructed using fuel assembly batch symbols, such as 'A2' and 'B0'. However, since the input data must be in numerical rather than character form, we converted the original data into numerical form using seven two-group macroscopic cross-section data obtained by homogenizing the cross-section information of the constituent materials in the fuel assembly. Using these cross-section data, we transformed the 2D and 3D loading patterns into 3D and 4D cross-section representations, respectively. Figure 1 shows an example of this conversion procedure.

For the macroscopic cross sections, we used nu-fission ( $\nu\Sigma_f$ ), capture ( $\Sigma_c$ ), transport ( $\Sigma_{tr}$ ), and scattering ( $\Sigma_s$ ) cross sections, excluding up-scattering in the second group. Since we assumed that up-scattering does not occur in the second group, we did not include the second group's scattering cross section. Consequently, the processed 2D loading pattern is represented as a  $5 \times 5 \times 7$  three-dimensional (3D) matrix, with the last dimension (7) representing the macroscopic cross sections. Following processing, the 3D loading pattern is represented as a  $24 \times 5 \times 5 \times 7$  four-dimensional (4D) matrix.

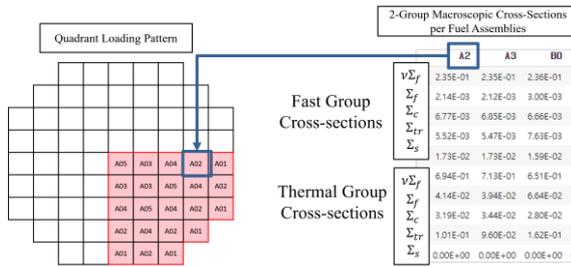


Fig. 1. Illustration of a step in converting a 2D loading pattern to a macroscopic cross-section matrix

We considered the critical control rod position as an additional input. These rod positions are part of the results obtained from the core analysis conducted using the in-house code on the generated random loading patterns.

Both 2D and 3D representations of the control rod position data initially contained the same information. The  $5 \times 5$  matrix was used to depict the 2D control rod location. For the 3D representation, the radial 2D control rod position was mapped across 24 axial planes based on the actual insertion depth, with values ranging between 0 and 1, resulting in a  $24 \times 5 \times 5$  matrix.

In the 2D representation as a  $5 \times 5$  matrix, the control rod position values range from 0 to 100 (%), indicating the insertion ratio. A value of 100 represents the fully withdrawn position at the top of the core, while a value of 0 indicates the fully inserted position at the bottom. The matrix includes four regulating control rod banks: R1, R2, R3, and R4, each with 50% overlap.

When converting the 2D control rod position ( $5 \times 5$ ) to a 3D representation ( $24 \times 5 \times 5$ ), the insertion depth is mapped onto a 24-layer axial structure. Each axial node is assigned a value between 0 and 1 based on the insertion depth. If a control rod completely occupies an axial node, a value of 1 is assigned. If it just partially occupies the node, a value between 0 and 1 is assigned. If the node is vacant, a value of 0 is assigned.

Through this process, the original 2D control rod representation ( $5 \times 5$ ) is transformed into a 3D matrix ( $24 \times 5 \times 5$ ), correctly capturing the control rods' axial insertion within the core. Fig. 2 depicts both the 2D

matrix form of the control rod position representation and its transformed 3D matrix form.

The output data serve as ground truth during the training process. Since the Axial Shape Index (ASI) is defined as the power generated in the lower half of the core less the power in the upper half of the core divided by the sum of these powers, we directly converted the 3D power distribution into ASI and used it for error calculation.

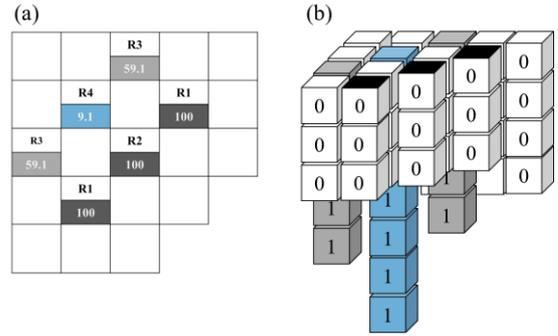


Fig. 2. Visualization of control rod position data: (a) 2D matrix representation ( $5 \times 5$ ), (b) 3D matrix representation ( $24 \times 5 \times 5$ ) with mapping.

## 2.2 Constructed Models

We compared five different models in this study, assigning them model numbers from 1 to 5. First, Models 1 and 2 have the identical neural network backbone structure. They both use a 2D loading pattern, which is then transformed into a 3D macroscopic cross-section that serves as input. Both models generate ASI, a value with a single scalar form. However, Model 2 takes critical control rod positions as additional input data.

The number of parameters, which is widely used as a measure of model size, refers to the total count of trainable variables in the neural network, including weights and biases. Model 1 has 250,797 parameters, and Model 2 has 251,817.

Despite sharing the same backbone structure, Model 2 has a slight variation in parameter count due to the additional layer required to process the (5,5) control rod input. Table I summarizes the inputs, outputs, and the number of parameters for Models 1 and Model 2.

Table I: Input and Output Shape of the Model 1 and Model 2

	Model 1 (baseline, 2D loading pattern)	Model 2 (2D loading pattern)
# of Parameters	250,797	251,817
1 <sup>st</sup> Input shape	(5, 5, 7)	(5, 5, 7)

2 <sup>nd</sup> Input shape	-	(5, 5)
Output shape	(1)	(1)

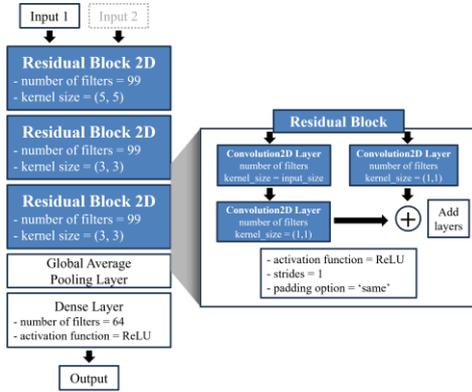


Fig. 3. Shared backbone network architecture of Model 1 and Model 2

Models 3, 4, and 5 utilize a 3D loading pattern, which is then transformed into a 4D macroscopic cross-section with a shape of (24, 5, 5, 7) as input. These models share the backbone structure of Model 3, which differs from Models 1 and 2.

Table II: Input and Output Shape of the Model 3, 4 and 5

	Model 3 (3D loading pattern)	Model 4 (3D loading pattern)	Model 5 (Fine tuned, 3D loading pattern)
# of Parameters	1,901,002	1,960,802	1,969,379
1 <sup>st</sup> Input shape	(24, 5, 5, 7)	(24, 5, 5, 7)	(24, 5, 5, 7)
2 <sup>nd</sup> Input shape	-	(24, 5, 5)	(24, 5, 5)
Output shape	(24, 5, 5)	(24, 5, 5)	(1)

Model 3 has 1,901,002 parameters and requires no more input beyond the macroscopic cross-section data. The model generates assembly-wise 3D power distribution in the shape of (24 × 5 × 5). The 3D power distribution is then converted to a single scalar value of ASI.

Model 4 is based on Model 3 but includes additional input representing control rod positions, with the shape of (24, 5, 5). This additional input contains the same information as that Models 2, but in a different format. Extra layers were added to process this new input, which resulted in a slight increase in the number of parameters to 1,960,802 over Model 3. Similarly to Model 3, Model 4's output of 3D power distribution is likewise converted into a single scalar ASI value.

Model 5 is a fine-tuned variant of Model 4's backbone structure. Fine-tuning is the process of adapting a pre-trained model with already optimized weights to a similar but distinct task. We fine-tuned Model 4, which was originally trained to predict the assembly-wise 3D power distribution, by modifying its structure to predict ASI. After these modifications, we retrained the model (Model 5) using a dataset that includes ASI values as output data. Model 5 uses the same input as Model 4 but differs in that it predicts a single scalar value as its output. Due to the additional layers used to construct the fine-tuned model, the number of parameters increased to 1,969,379. Table II provides summary on the input, output, and the number of parameters for Models 3, 4 and 5.

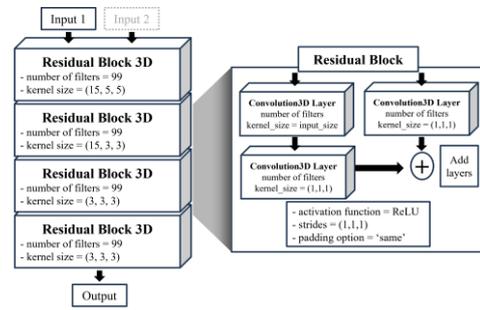


Fig. 4. Shared backbone network architecture of Models 3, 4 and 5.

### 3. Results

All models were regression models designed to predict specific values based on the input data. A total of 37,532 data points (loading patterns) were used. Of these, 32,089 were used for training, 3,754 for testing, and 1,689 for validation during the training process. All models were trained with a learning rate of 1e-4 and Mean Squared Error (MSE) as the loss function. No batch size was assigned, and training was conducted for a total of 150 epochs.

Since the test dataset (3,754 loading patterns) was processed in one batch, only the average prediction speed could be measured. Models 1 and 2 achieved a prediction speed of 1 ms for the 3,754 loading patterns, whereas Models 3, 4, and 5 required 1 second on the computational server detailed in Table III. As prediction and training speeds are typically correlated with model size (parameter count and complexity), the slower prediction speed of Models 3, 4, and 5, relative to Models 1 and 2, is consistent with their larger size. The errors, standard deviations, and average calculation speeds for the predictions of each model are summarized in Table IV.

To evaluate model performance, scatter plots were created showing the ideal line (representing the actual data) and the predicted values (see Fig. 5). The density of points around the ideal line provides a rough

indication of the performance. Figures 6 to 10 demonstrate the error distributions (actual value - predicted value) for Models 1-5, including the mean, max, min, median, mean  $\pm$  stdev, and stdev.

Table III: Computational Server Specifications

Component	Specification
CPU	Intel Xeon Silver 4410Y (18M Cache, 2.10 GHz - 3.30 GHz, 12C/24T, 120W) * 2
GPU	NVIDIA RTX A6000 GDDR6 48GB
RAM	16GB 2RX4 DDR5 RDIMM 4800MHz * 2
OS	Ubuntu 22.04.05 LTS

Table IV: Summary of Prediction Performance for Models 1-5

	Model 1	Model 2	Model 3	Model 4	Model 5
Mean Abs Error	1.16e-02	4.48e-03	5.73e-03	2.98e-03	<b>2.50e-03</b>
Max Abs Error	1.09e-01	2.24e-02	6.07e-02	3.15e-02	<b>1.26e-02</b>
Stdev	1.50e-02	5.41e-03	7.96e-03	3.34e-03	<b>2.52e-03</b>
# of params	251K	252K	1.90M	1.96M	1.97M
Avg. Prediction Time	<b>0.266<math>\mu</math>s</b>	<b>0.266<math>\mu</math>s</b>	0.266ms	0.266ms	0.266ms

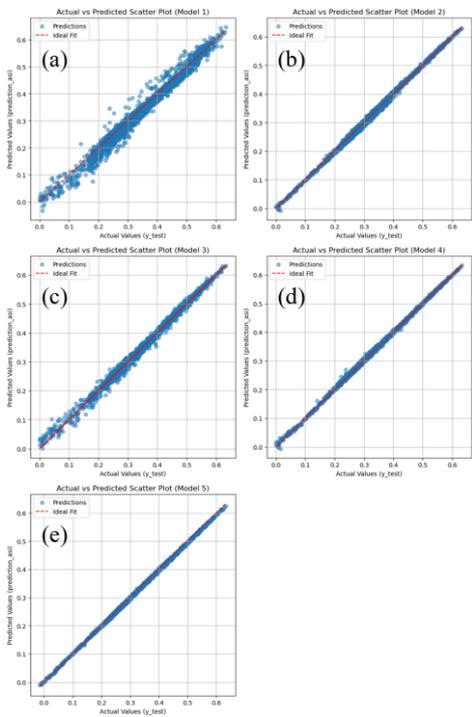


Fig. 5. Scatter plot of predicted values from Models 1-5, with the ideal line.

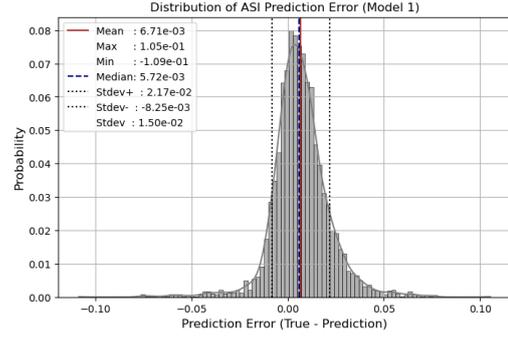


Fig. 6. Distribution of Prediction Errors for Model 1.

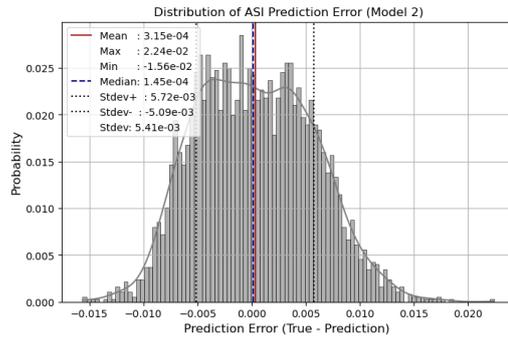


Fig. 7. Distribution of Prediction Errors for Model 2.

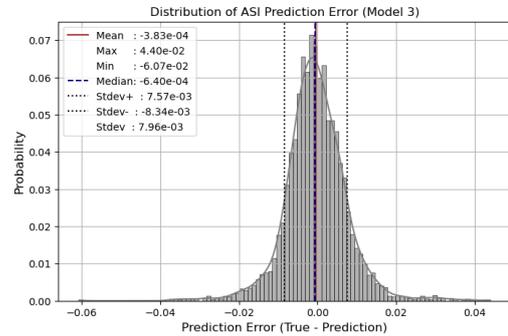


Fig. 8. Distribution of Prediction Errors for Model 3.

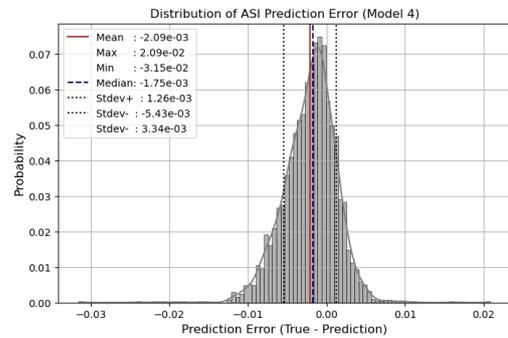


Fig. 9. Distribution of Prediction Errors for Model 4.

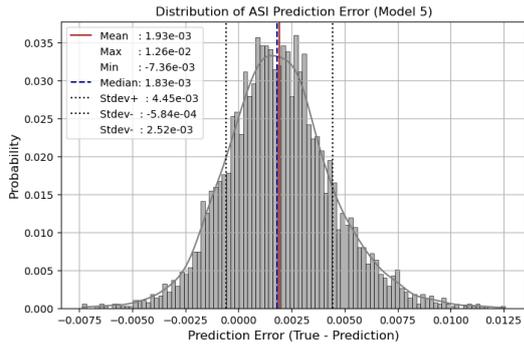


Fig. 10. Distribution of Prediction Errors for Model 5.

#### 4. Conclusions

We used a core analysis surrogate model to compare the prediction performance of the Axial Shape Index (ASI) of a clean core. Model 1, a CNN-based model utilizing a 2D loading pattern developed in a previous study, served as the baseline model. We then compared the prediction performance of four additional models by varying the input data formats (2D and 3D) and neural network structures. The results showed that Model 5, which used both 3D loading pattern and control rod information as input while employing fine-tuning techniques, achieved the best prediction performance (MAE:  $2.50 \times 10^{-3}$ , Stdev:  $2.52 \times 10^{-3}$ ). In comparison to Model 1, the MAE and standard deviation were lowered by 78.4% and 83.2%, respectively.

##### 4.1 Comparison of the Impact of Control Rod Information on Model Performance ('Model 1 vs. Model 2' and 'Model 3 vs. Model 4')

The impact of control rod information on model performance was investigated by comparing 'Model 1 vs. Model 2' and 'Model 3 vs. Model 4', as seen in Fig.11.

In the case of 2D loading pattern models, Model 1 (without control rod information) had a mean absolute error (MAE) of  $1.16 \times 10^{-2}$ , while Model 2 (with control rod information) had an MAE of  $4.48 \times 10^{-3}$ , resulting in a 61.2% reduction in prediction error. Similarly for 3D loading pattern models, Model 3 (without control rod information) had an MAE of  $5.73 \times 10^{-3}$ , whereas Model 4 (with control rod information) achieved an MAE of  $2.98 \times 10^{-3}$ , resulting in a 48.0% reduction in error.

These findings show that the accuracy of ASI prediction in both 2D and 3D loading pattern models is greatly increased by including control rod information.

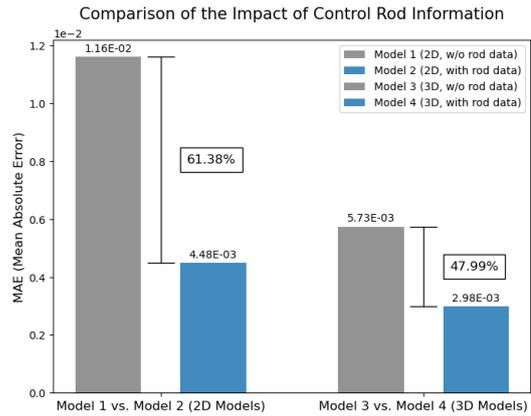


Fig. 11. Impact of control rod information on model performance

##### 4.2 Effect of Fine-Tuning in 3D Models (Model 4 vs. Model 5)

The effect of fine-tuning on model performance was analyzed by comparing Models 4 and 5 as shown in Fig. 12. Both models utilized the same 3D loading pattern and control rod data for input. Model 5 is a fine-tuned variation of Model 4. As a result, the mean absolute error (MAE) decreased from  $2.98 \times 10^{-3}$  in Model 4 to  $2.50 \times 10^{-3}$  in Model 5, indicating a 16.1% improvement in prediction accuracy.

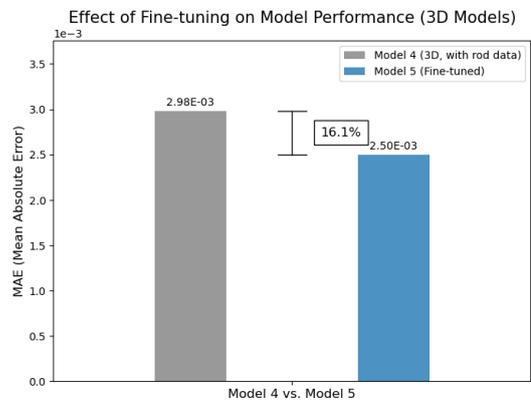


Fig. 12 Effect of fine-tuning on Model Performance (MAE)

##### 4.3 Comparison Between 2D and 3D Loading Patterns Models ('Model 1 vs. Model 3' and 'Model 2 vs. Model 4')

The impact of loading pattern dimensionality on ASI prediction performance was evaluated by comparing 'Model 1 vs. Model 3' and 'Model 2 vs. Model 4', as shown in Fig. 13.

When control rod information was not included, the 3D loading model (Model 3) had a mean absolute error (MAE) of  $5.73 \times 10^{-3}$ , which was 50.6% lower the 2D loading model (Model 1) at  $1.16 \times 10^{-2}$ . Similarly, when control rod information was included, the 3D model

(Model 4) achieved an MAE of  $2.98\text{e-}03$ , a 33.5% reduction over the 2D model (Model 2), which had an MAE of  $4.48\text{e-}03$ .

These results indicate that using a 3D loading pattern significantly improves ASI prediction accuracy, regardless of whether control rod information is given.

The improved accuracy can be attributed to several factors. First, the 3D models utilize the in-core 3D distribution, which represents the uncompressed values used to derive ASI. This single scalar value, in contrast, inherently provides less information for understanding the correlation between input and target data compared to the 3D distribution.

Furthermore, the 3D loading pattern includes additional features, such as the cutback region at the top layer, providing more information in the input data. Specifically, the input and output dimensions for the 3D model are (24, 5, 5, 10) and (24, 5, 5), respectively, offering much richer information compared to the 2D case, where the input and output dimensions are (5, 5, 10) and (1). It is believed that this detailed spatial and dimensional information contributed to the model's enhancement.

[4] J. Kwon, T. Park, S.K. Zee, Development of CNN-Based surrogate models to predict power distributions in i-SMR, Annals of Nuclear Energy, Vol.213, 2025.

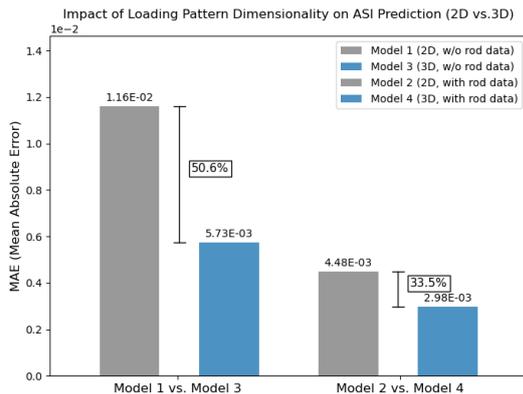


Fig. 13. Impact of the dimensionality of loading pattern on prediction performance (MAE)

## ACKNOWLEDGEMENT

This research was supported by the National Research Council of Science & Technology(NST) grant by the Korea government (MSIT) (No. GTL24031-000).

## REFERENCES

- [1] W. Li et.al, Artificial neural network reconstructs core power distribution, Nuclear Engineering and Technology, Vol.54, p. 617, 2022.
- [2] J. Kwon, T. Park, S.K. Zee, AI-Based Prediction Module of Key Neutronic Characteristics to Optimize Loading Pattern for i-SMR with Flexible Operation, Korean Journal of Chemical Engineering, Vol.41, p.2741–2759 2024.
- [3] Y. Lecun, L. Bottou, Y. Bengio, P. Hafner, Gradient-based learning applied to document recognition, IEEE, Vol.86, p. 2278, 2024.