Transactions of the Korean Nuclear Society Spring Meeting
Jeju, Korea, May 22-23, 2025

Development of a MAAP5 Metamodel for Use in Dynamic PSA

Hyeonmin Kim [a], Sang Hoon Han [a], Dong-San Kim [a*]
[a] *Risk Research Assessment Division, Korea Atomic Energy Research Institute, 111, Daedeok-daero 989 beon-gil, Daejeon, Korea*
[*] *Corresponding author: reisi-fard.1@osu.edu*

***Keywords :** metamodel, deep learning, dynamic PSA, surrogate model

1. Introduction

Probabilistic Safety Assessment (PSA) has emerged as an essential component in nuclear safety engineering. It provides a structured and quantitative approach for evaluating the frequency and potential consequences of accidents within nuclear power plants. By systematically modeling combinations of equipment malfunctions, operator errors, and various accident scenarios, PSA enables a comprehensive risk assessment.

Conventional PSA techniques rely on static event tree (ET) and fault tree (FT) models to enumerate accident sequence and system failures. While the conventional approach has been effective in identifying dominant risk contributors, it assumes a fixed branching structure of events and does not explicitly account for the dynamics of accident progression. In reality, nuclear accidents evolve over time with complex interactions – equipment can degrade gradually, control systems respond in real time, and operators take actions at various points. Rigid ET/FT method struggle to capture such time-dependent behaviors, leading to potential uncertainties. This limitation has motivated a transition toward dynamic PSA, in which the temporal evolution of scenarios is explicitly simulated for more realistic assessment. A dynamic PSA approach integrates deterministic system simulations (e.g., thermal-hydraulic codes) with stochastic scenario sampling, going beyond pre-defined event sequences. Moving from static to dynamic PSA allows analysts to enhance the accuracy and realism of safety assessments [2].

Despite its promise, dynamic PSA comes with significant challenges. One major issue is the state-space explosion problem: as we incorporate time steps and more detailed system states, the number of possible accident sequence paths grows [3, 4]. This can result in an overwhelming explosion of scenarios or system states to consider, far beyond what analysts can manage with brute-force simulation. Indeed, a dynamic simulation of even a single initiating event can branch into countless sequences when varying the timing of component failures or operator actions. Handling this combinatorial complexity often requires heavy computational resources and intelligent truncation or sampling strategies.

To address these challenges, researchers have begun exploring the use of metamodels (or surrogate models) to make dynamic PSA more tractable. The key idea is to replace or augment the direct physics-based simulations with an approximate model that is much faster to execute. In particular, data-driven deep learning models are promising metamodel candidates for PSA because of their ability to learn complex non-linear mappings from inputs to outputs [5]. For example, a trained neural network can take as input the parameters of an accident scenario (e.g. initiating event, sequence of component statuses, timing of events) and instantly predict the outcome (such as core damage probability or key system parameters), after being trained on a library of simulation results. This approach can alleviate the computational burden dramatically. Instead of running tens of thousands of full simulations for different scenarios, one can run a smaller set of simulations to train the metamodel and then let the metamodel rapidly evaluate millions of what-if scenarios. In essence, leveraging metamodels rooted in artificial intelligence can make dynamic PSA feasible for practical use by cutting down computation time and automating the analysis of myriad scenario variations.

Building on this background, the present study applies deep learning-based metamodeling to improve dynamic PSA for nuclear power plants.

We develop a novel PSA metamodel framework that leverages an inception-inspired neural network with attention mechanisms to predict accident scenario outcomes efficiently and accurately.

This paper extends our previous study entitled “Dynamic PSA-based multi-unit accident scenario modeling approach,” which provided a comprehensive description of the data generation procedures. Hence, in this paper, we only summarize the main aspects of data generation and instead focus on how the deep learning-based metamodeling framework is constructed and validated.

2. Data Generation and preprocessing

In this chapter, we describe data that use to train model and represent accident scenario. To generate dynamic accident scenarios, we used the Modular Accident Analysis Program version 5 (MAAP5), a computer code capable of simulating severe accident sequences in light water reactors.

2.1 Data Generation considering dynamic features

This chapter describes the process of generating data for dynamic accident scenarios in nuclear power plants, focusing on Loss of Offsite Power (LOOP) and Station Blackout (SBO). A LOOP occurs when external power is disrupted—for instance, by a damaged transmission line—prompting the start-up of Emergency Diesel Generators (EDGs). If both EDGs fail, it escalates into an SBO.

We also consider multi-unit operations to reflect how backup resources are shared. In an SBO, multiple reactors may rely on Alternative AC (AAC) or MACST diesel generators, and these shared systems can affect more than one reactor simultaneously. Figure 1 provides an example timeline showing which components remain functional or fail over time, creating several possible accident sequences.

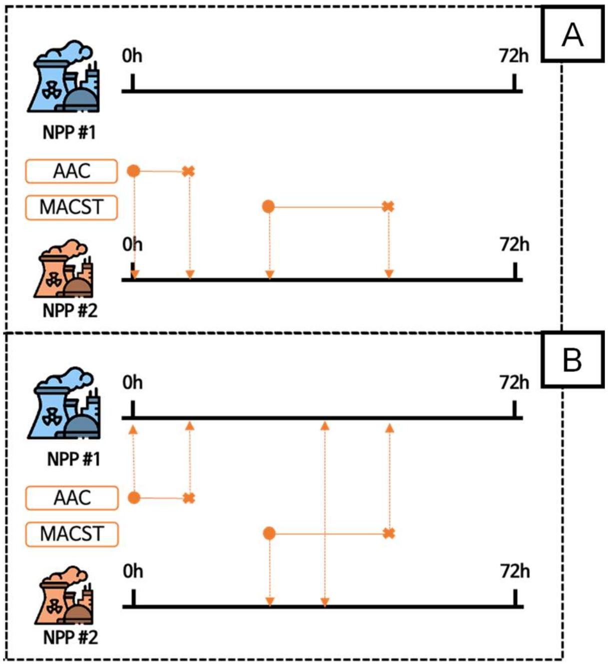

Figure 1. Simplified example for accident sequence modeling

To analyze these scenarios, we developed a computer program that probabilistically determines whether each component is in standby, operating, or failed. Details of this approach are described in our previous research papers[1] .

2.2 Data Preprocessing

First, we extracted key variables (e.g. AC power, high-pressure safety injection (HPSI), auxiliary feedwater system (AFW), portable low-pressure pump (PLPP), and containment spray system (CSS)) from the simulation text using regular expressions, thereby quantifying the operating or non-operating status of each. If a particular variable wad undefined, we assigned a sentinel value (e.g., -1). Additionally, if it was determined that no failures occurred for the entire simulation period, we designated a special value (e.g., 1000) to distinguish that case.

Next, we performed min-max scaling on all variables to ensure consistent representation in subsequent modeling. Usually, this follows the formula:

$$x_{\text{scaled}} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

where x_{min} and x_{max} represent the minimum and maximum values for each variable. Although the typical mapping is $[0,1]$, alternate ranges such as $[-1,1]$ may also be used. By keeping relative distances intact, smaller-range variables (such as level or temperature) and larger-range variable (pressures in the order of billions of pascals) can be effectively learned by the same model. Otherwise, even small variations in a high-scale variable could be misconstrued as a disproportionately significant change.

In complex accident scenarios, the actual probability of core damage is generally extremely rare compared to normal cases. Due to practical time constraints in generating training data, we adjusted the ratio of accident occurrences. Empirically, we reasoned that core damage would be physically more complex, influenced by denser intervals, whereas "OK" cases have a broader operational range and can be sampled more sparsely. Consequently, we deliberately increased the probability of these otherwise low-probability core damage events. However, depending on how various probability parameters are adjusted, data imbalances can arise (75% core damage case, 25% OK case). To address these distributions, we employed SMOTE (Synthetic Minority Over-sampling Technique) to strengthen the minority class. Suppose we have minority-class sample A (in case of OK) and we pick one of its k-nearest neighbors B. By introducing a random scaler $\delta \in [0,1]$, a new synthesized sample A_{new} is computed as:

$$A_{\text{new}} = A + \delta(B - A)$$

This approach expands the distribution of the minority class without relying on mere duplication, thereby mitigating overfitting risks and promoting the model's ability to learn essential patterns.

In summary, the sequence of extracting major variables, handling missing or special values, applying Min-Max scaling, and augmenting data via SMOTE forms a more cohesive dataset. Balanced and normalized data consequently reduces model sensitivity to extremes or minority-class shortfalls, which increases predictive reliability in subsequent analyses, such as Dynamic Event Tree methods or machine-learning algorithms for accident scenario evaluation. The clearer interpretability

[1] Sang Hoon Han, Hyeonmin Kim, Dong-San Kim, "Dynamic PSA-based multi-unit accident scenario modeling approach," Korean Nuclear Society Spring Meeting, Korea, May 22-23, 2025.

of results also benefits actual design and safety assessment phases, helping to guide more informed decision-making.

3. Structure of Metamodel

This chapter provides an in-depth overview of the developed model, emphasizing its distinctive capability to predict core states with only limited input data. Central to this predictive capacity is a novel convolutional structure that forms the foundation of Convolutional Neural Networks (CNNs), leveraging convolutional filters for efficient feature extraction. In the context of one-dimensional (1D) convolution, where data are arranged along a single spatial dimension, the network effectively captures local dependencies in time series or sequential inputs, supporting tasks such as sequence classification and language processing. This multi-scale feature extraction approach enables the model to construct intricate data representations while preserving a relatively small input dimension.

To achieve robust performance, the proposed architecture incorporates an inception-based design [6], originally recognized for its high efficiency in image recognition domains. The Inception framework excels at learning complex patterns across multiple scales, optimizing parameter usage, and exhibiting flexible structural designs. Specifically, this design allows the model to concurrently capture fine-grained and large-scale patterns, using multi-branch convolutions of varying kernel sizes, combined with residual and pooling pathways.

The model consists of four specialized inception variants - namely *inception_n*, *inception_s*, *inception_w*, and *inception_a* - each tailored to address different spatial scopes and operational requirements:

- *Inception_n*

  - Intended for capturing patterns within narrower input segments.
  - Employs 1 × 1 and 3 × 3 convolutions alongside average pooling.
  - Targets localized feature representation by gradually expanding the receptive field.

- *inception_s*

  - Functions similarly to inception_n by focusing on narrow regions.
  - Incorporates a scaled channel approach that halves the output channel size relative to inception_n, helping mitigate gradient vanishing issues.
  - Balances representational power and computational efficiency.

- *inception_w*

  - Optimized for capturing features over broader input segments.
  - Deploys larger kernels (e.g., 15 × 15, 17 × 17) to learn longer-range patterns.
  - Targets inputs where extended temporal or spatial dependencies are critical.

- *inception_a*

  - Extends inception_n's structure by integrating a self-attention mechanism.
  - Improves global context modeling, enabling the network to highlight key features beyond local receptive fields.

The self-attention component, inspired by transformer architectures, facilitates parallel attention across multiple feature representations, affording significantly improved performance on various deep learning tasks [7]. Meanwhile, the model's overall non-linearity is enhanced by the Gaussian Error Linear Unit (GELU) activation function, while training stability benefits from batch normalization and residual skip connections. These design choices collectively yield a model that can capture salient features at multiple scales and emphasize those of greatest importance.

Extensive empirical evaluations led to a final model structure comprising three repeated blocks, each followed by a fully connected layer. Although *inception_w* was implemented to capture broad input segments with large kernels, we ultimately did not include it in the final architecture after empirical evaluations. Each block contains a sequence of submodules ($s \rightarrow n \rightarrow n \rightarrow b \rightarrow a$), specifically:

- Inception_s (s)
- Inception_n (n)
- Inception_n (n)
- Batch Normalization (b)
- Inception_a (a)

After the repetition of these three blocks, the model transitions to a fully connected (*FC*) layer comprising 1024 units. This architecture can be succinctly expressed as:

$$3 \times [s\text{-}n\text{-}n\text{-}b\text{-}a] + FC(1024)$$

By blending multi-scale inception paths, attention-based global modeling, residual connections, and advanced activation functions, the proposed model efficiently handles large-scale sequences with minimal input requirements. In practice, this approach yields high predictive accuracy for core status monitoring under complex scenarios, while imposing relatively modest demands on input dimensionality and computational overhead.

As the final step in data preprocessing, the complete dataset was split into three subsets: train, validation, and test. First, 10% of the entire dataset was set aside as the test set to be used exclusively for the final evaluation of model performance. Form the remaining 90%, another 10% was separated as the validation set, while the remaining 80% served as the train set. This split design ensures that the model is initially trained on the train set. During training, the validation set is used to prevent overfitting by tuning hyperparameters. Finally, the test set is utilized to measure prediction performance, thereby verifying how well the model generalizes to new, unseen data.

4. Results

Table 1 compares performance metrics of the developed model classification model without and with SMOTE. When SMOTE was applied, the accuracy slightly increased from 0.9700 to 0.9710. Although the recall decreased from 0.9895 to 9750, the precision increased markedly 0.9716 to 9867. Consequently, the F1 score improved slightly from 0.9805 to 0.9808. These outcomes indicate that applying SMOTE boosts the model's overall precision and maintains a balanced performance across different metrics. Figure 2 further illustrates how the SMOTE-enhanced model performs across various data points.

Table 1. Comparison of performance metrics by SMOTE

Metric	Without SMOTE	With SMOTE
Accuracy	0.9700	0.9710
Precision	0.9716	0.9867
Recall	0.9895	0.9750
F1 Score	0.9805	0.9808

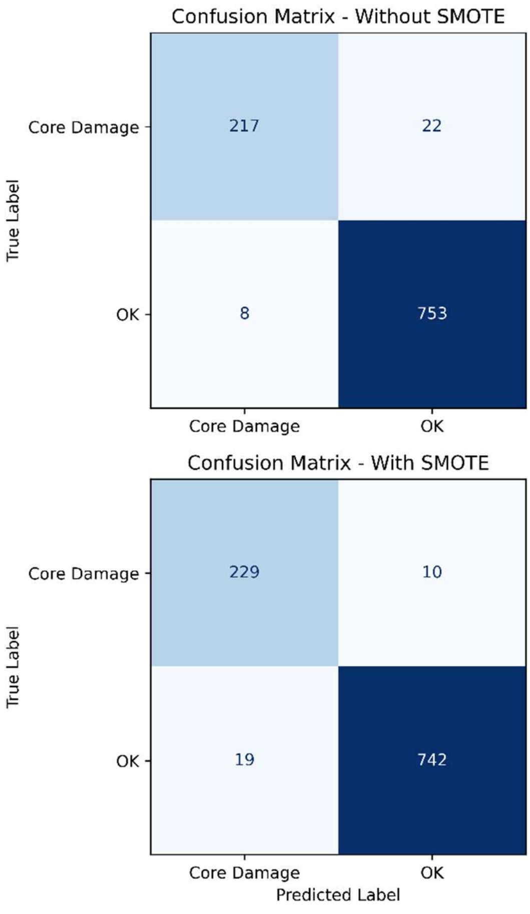

Figure 2. Results of developed model by SMOTE

One of the most critical obstacles in dynamic PSA involves mitigating the so-called "state explosion," wherein the computational burden escalates exponentially in response to increasing accident-scenario complexity. The present study illustrates that, although achieving high predictive accuracy is indispensable, computational efficiency, especially inference speed, merits equal attention. For instance, simulating 10,000 LOOP-SBO accident scenarios using MAAP5 on a 24-core system can require roughly ten days, whereas a trained model can evaluate the same set of scenarios in approximately 7.5 s on an NVIDIA RTX 6000A GPU. This contrast in runtime clearly demonstrates the practical benefits conferred by faster inference methods.

3. Conclusions

A contribution of this work lies in its comparative analysis of models trained with and without SMOTE, revealing a notable trade-off between precision and recall. While both approaches achieve approximately 97% accuracy and an F1 score near 0.98, the SMOTE-based model attains higher precision but exhibits a modest decline in recall, whereas the non-SMOTE model preserves better recall at the expense of slightly lower precision. Consequently, the decision as to which approach to adopt depends heavily on the relative costs of false negatives versus false positives within a given nuclear safety context. In scenarios where missing a core damage event poses disproportionate risks, a model favoring recall (i.e., the non-SMOTE approach) proves more suitable. Conversely, if the burden of false alarms introduces undue operational complications, a model with SMOTE's enhanced precision may be the better choice. In addition, targeted hyperparameter tuning, such as adjusting the SMOTE ratio or optimizing the decision threshold, can refine the balance between recall and precision to suit specific risk tolerances and cost structures.

Finally, this study underscores the importance of presenting model results with explicit measures of uncertainty, for example, confidence intervals for the main performance metrics, to bolster decision-making in safety-critical environments. Future work could include ablation studies or other forms of sensitivity analysis to map more precisely the influence of class-imbalance strategies, threshold settings, and cost-sensitive learning. Ultimately, while model accuracy remains a cornerstone of dynamic PSA, the ability to execute large-scale accident-scenario simulations rapidly may prove just as essential for managing nuclear power plant operations both safely and efficiently.

ACKNOWLEDGEMENT

This work was supported by the National Research Foundation of Korea (NRF), funded by the Ministry of Science and ICT (No.RS-2022-00143695)

REFERENCES

[1] Smith, C., Mandelli, D., & Le Blanc, K. (Eds.). (2023). Risk-informed methods and applications in nuclear and energy engineering: Modeling, experimentation, and validation (1st ed.). Elsevier.
[2] Parhizkar, T., Vinnem, J. E., Utne, I. B., & Mosleh, A. (2021). Supervised dynamic probabilistic risk assessment of complex systems, part 1: General overview. Reliability Engineering & System Safety, 208, 107406.
[3] Heo, G., Baek, S., Kwon, D., Kim, H., & Park, J. (2021). Recent research towards integrated deterministic–probabilistic safety assessment in Korea. Nuclear Engineering and Technology, 53(11), 3465–3473.
[4] Ryu, S., Kim, H., Kim, S. G., Jin, K., Cho, J., & Park, J. (2018). Probabilistic deep learning model as a tool for supporting the fast simulation of a thermal–hydraulic code. Expert Systems with Applications, 200, 116966.
[5] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., … & Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1–9). IEEE..
[6] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), Advances in neural information processing systems (Vol. 30). Curran Associates, Inc.