Physics Informed Neural Network based NPP Simulation methods

Young Ho Chae [a], Hyeonmin Kim [b], Jungjin Bang [c], Poong Hyun Seong [a*]
[a]*Department of Nuclear And Quantum Engineering, Korea Advanced Institute of Science and Technology, 34141, 291, Daehak-ro, Yuseong-gu, Daejeon, Republic of Korea*
[b]*Korea Atomic Energy Research Institute, 34057, 111, Daedeok-daero, 989 Beon-gil, Yuseong-gu, Daejeon, Republic of Korea*
[c]*Future and Challenge Tech. Co. Ltd., Heungdeok1ro 13, Yeongdeok-dong, Giheung-gu, Yongin-si, Gyeonggi-do, Republic of Korea*

[*]*Corresponding author: phseong1@kaist.ac.kr*

1. Introduction

A nuclear power plant (NPP) is a safety-critical system with large size and high complexity. To secure the safety of an NPP, several analysis methods were developed to identify possible accidents and prepare for accidents. The analysis methods can be divided into two categories. The first is the experimental method and the other is the simulation method. the most accurate and realistic analysis result can be obtained by an experiment. However, in reality, experiments on NPPs are very expensive and difficult to perform due to safety issues. Therefore, various simulation-based analysis methods were developed. For the reactor analysis, Open-source Nuclear Codes for Reactor Analysis (ONCORE) was developed. For the thermal-hydraulic analysis, Reactor Excursion and Leak Analysis Program (RELAP), Modular Accident Analysis Program Software (MAAP), Modular, engineering-level computer code (MELCOR), Multi-dimensional Analysis of Reactor Safety (MARS) were developed. For the behavioral analysis of NPP, Compact Nuclear Simulator (CNS) was developed. However, the simulation-based analysis methods have limitations that, if the number of nodes and time units are divided to increase the analysis resolution, the time required for the calculation tends to increase exponentially, and there are very limited ways to update the real data to the simulator.

By solving the simulation accuracy-calculation speed trade-off, the simulation methods can be applied to the fields that require calculation speed. For example, in the current situation, the thermal-hydraulic codes are hard to apply dynamic probabilistic safety assessment due to the calculation speed. And improving the simulator that can reflect data from the real experiment will enable more realistic simulations. Therefore, we propose a novel AI utilized Physics Related Information-based Simulation Method (A-PRISM). A-PRISM consists of a solution generator and an equation generator. Both generators are based on physics informed neural network model. The solution generator calculates the simulation results according to the given initial and boundary condition. The equation generator creates the equation that best describes the data from the real world. The generated equation automatically updates the physics part of the solution generator. As a result, the model creates more realistic simulation results.

2. AI utilized Physics Related Information based Simulation Method (A-PRISM)

The schematic diagram of A-PRISM model is described in Fig. 1. A-PRISM consists of a solution generator and an equation generator (model generator). Both generator are based on physics informed neural network (PINN) model. The solution generator calculates the simulation result for the given condition. And the equation generator creates appropriate equations automatically from real-world data. The created equation is provided to the solution generator. And as a result of the update, a more realistic solution is generated.

In this chapter, description about PINN, solution generator and equation generator of A-PRISM will be provided.

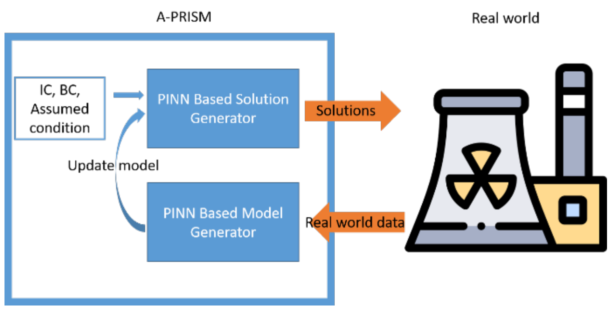

Fig. 1 A-PRISM Model

2.1 Physics Informed Neural Network (PINN)

A-PRISM, the model proposed in this study, is based on the Physics informed neural network (PINN) model. PINN is one of the AI models using inductive bias and operates by providing physics information with inductive bias. The algorithm was firstly proposed by M. Raissi et al. [1]. The main difference between PINN and the naive neural network model is loss. The naive neural network model calculates loss as the difference between the output of the neural network (latent vector) and the target value (target vector). The PINN model utilizes the conventional loss from a naive neural network and it also utilizes the loss from the equation. The schematic diagram of the neural network and PINN are shown in

Fig. 2. By adopting inductive bias from the form of the equation, PINN can increase data efficiency compared to the existing neural network model and has extrapolation robustness. PINN model has several benefits however, due to the nature of the data-driven model, the prediction may fail when the solution has a discontinuity point.

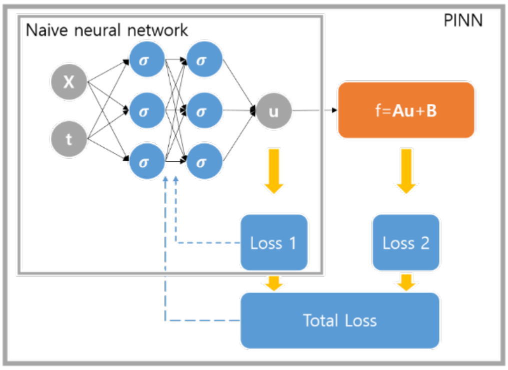

Fig. 2 Neural network and PINN

2.2 Solution Generator

The solution generator calculates the result corresponding to the given condition like the existing simulation method. As an input for the network initial condition, boundary condition assumed condition, and equations are provided. And the output from the network is the simulation result (calculation result) of the given condition. The schematic diagram of solution generator is shown in Fig. 3. The detailed architecture of solution generator is listed in below.

Neural network part
- Number of layer: 4 layers
- Neurons in each layer: 30 neurons
- Activation function: adoptive rectified linear unit
- Loss function: mean squared error (MSE)
- Optimization algorithm: Limited memory-BFGS

Physic network part
- Number of layer: 2 layers
- Neurons in each layer: 3 (partial derivation to t, partial derivation to z, I)
- Activation function: adoptive rectified linear unit
- Loss function: MSE from physics loss

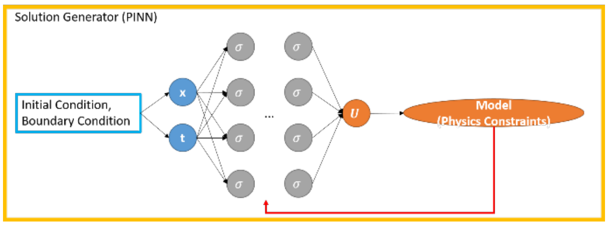

Fig. 3 Solution generator

2.3 Equation Generator (Model Generator)

The numerical modeling procedure follows the sequences below.

1. Specify the problem
2. Set up a metaphor
3. Formulate a mathematical model
4. Solve a mathematical problem
5. Interprete solution
6. Compare with reality
7. Use developed model

Each stage requires not only human intelligence but cost. Therefore, updating the simulation model with experiment results requires a lot of resources. Therefore, A-PRISM model has an equation generator to automatically analyze the experiment result and update the simulator (solution generator). The schematic diagram of equation generator is described in Fig. 4. The detailed architecture of equation generator is listed in below. As an output equation generator creates partial differential equation for the given experiment data.

Neural network part
- Number of layer: 4 layers
- Neurons in each layer: 30 neurons
- Activation function: adoptive rectified linear unit
- Loss function: mean squared error (MSE)
- Optimization algorithm: Limited memory-BFGS

Physic network part (AutoDiff)
- Number of layer: 2 layers
- Neurons in each layer: 3 (partial derivation to t, partial derivation to z, I)
- Activation function: adoptive rectified linear unit
- Loss function: MSE from physics loss

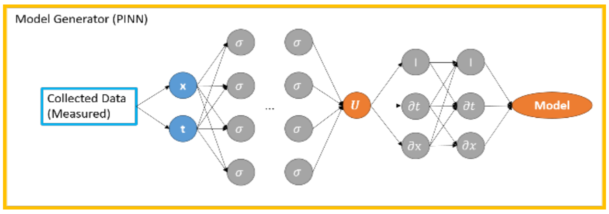

Fig. 4 Equation (model) generator

3. Experiment

3.1 Experiment 01

To verify and validate the A-PRISM model, we solved an infinite planar source with time dependent neutron diffusion equation as a pilot example. As real-world experiment data, the points that satisfy Eq.1 are collected. 10,000 points were collected. To prevent the duplication of data, the Latin hypercube sampling method is also adopted. With collected data, the equation generator generates an equation.

$$\frac{\partial^2 \phi}{\partial x^2} - \phi * 0.5 = \frac{\partial \phi}{\partial t} \quad \text{Eq. 1}$$

After the sufficient training sequences (computation time: 98.831sec), the equation generator creates the equation as Eq.2.

$$0.99997\frac{\partial^2 \phi}{\partial x^2} - \phi * 0.50011 = 0.99999\frac{\partial \phi}{\partial t} \quad Eq. 2$$

The result shows that the equation generator successfully creates the model (successfully imitates Eq.1). After the creation, the output of the equation generator is utilized to update the physics network of the solution generator. The solution generator generates an appropriate solution that satisfies the equation from the equation generator. The calculation speed via each convergence criteria is summarized in Fig. 5.

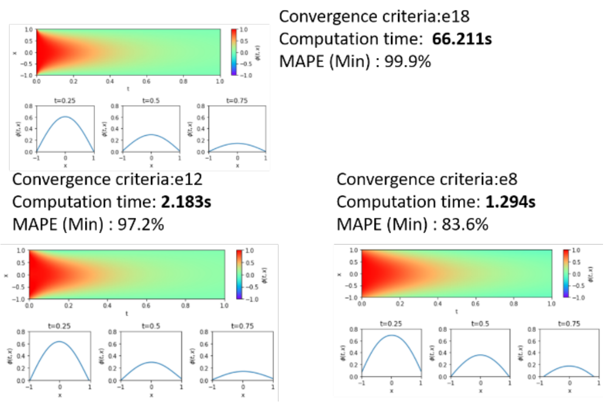

Fig. 5 Analysis results – Experiment 01

Due to the characteristics of the model, the stricter the convergence conditions were set, the longer the calculation time increased. If an appropriate level of convergence condition was set, accurate results could be obtained quickly.

3.2 Experiment 02

To verify the applicability of the suggested method for the actual NPP analysis code, MARS code-based experiment is conducted.

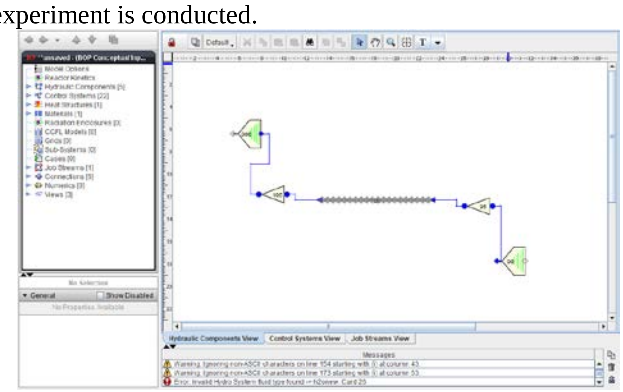

Fig. 6 MARS Geometry

The schematic geometry for experiment 02 is shown in Fig.6. The total pipe length is assumed as 100m, and the area of the pipe is assumed as 0.01 square meters. Initially, both side of the pipe is pressurized with 5bar and 20 seconds later, the output side is depressurized to 1 bar linearly (20s – 1000s). The detailed specifications are summarized as follows.

- Description of data discretization

1. $x \in [5, 100]_{n=20}$
2. $t \in [20, 800]_{n=40}$

The equations are trained with the flow velocity data from 20 seconds to 800 seconds and predicted flow velocity from 800s to 1000s. The calculated results are shown in Fig. 7 and Fig.8. The Fig.7 shows actual MARS code analysis results and the Fig.8 shows trained and predicted results from the suggested model.

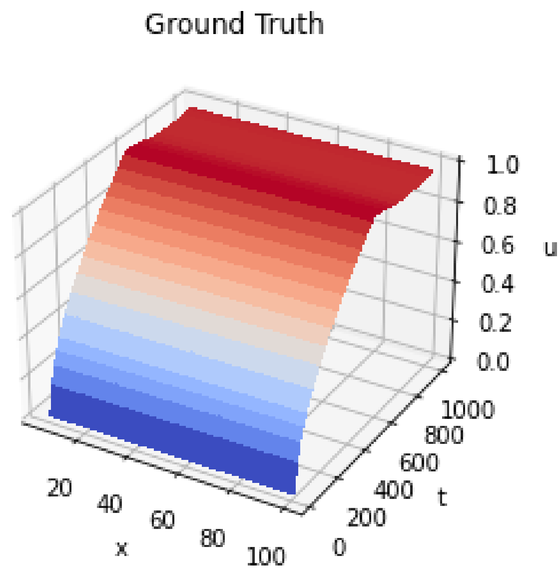

Fig. 7 MARS analysis results

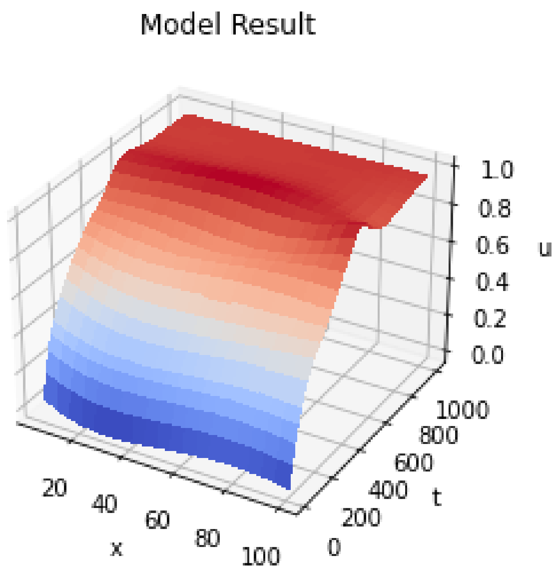

Fig. 8 Analysis results from the model

The model shows small fluctuations however, overall, the error was 4.200229e-02. Therefore, the model successfully imitates and simulates the experiment conditions.

4. Conclusions

Existing simulation methodologies have accurate calculation results, but they take a long time to calculate due to the nature of the numerical method. And it has the disadvantage that it is difficult to update the simulation model based on the experimental results.

Therefore, in this study, an AI utilized Physics Related Information based Simulation method (A-PRISM) was proposed to solve the aforementioned problems. A-PRISM is composed of an Equation generator that generates an appropriate equation to reflect the actual data to the model, and a Solution generator that performs an actual simulation based on the created equation. When the proposed methodology was applied to the time-dependent neutron diffusion equation, it was confirmed that the Equation generator simulated the equation well, and also the solution generator calculates simulation results quickly.

As further research, the suggested method will be applied to the thermal-hydraulic analysis which is composed of the energy equation, momentum equation, and continuity equation.

Using the proposed methodology that can easily reflect fast calculation results and real data, it is expected that the method will contribute to dynamic probabilistic safety assessment and digital twins.

Acknowledgement

This research was supported by the National R&D Program through the National Research Foundation of Korea (NRF) funded by the Korean Government. (MSIP: Ministry of Science, ICT and Future Planning) (No. NRF-2016R1A5A1013919)

NPP image in Fig. 1 is from flaticon.com

REFERENCES

[1] M. Raissi, Paris Perdikaris, and George Em Karniadakis, 'Physics Informed Deep Learning (Part 1): Data-driven Solutions of Nonlinear Partial Differential Equaitons, arXiv:1411.10561v1, 28 Nov., 2017