Transactions of the Korean Nuclear Society Virtual Autumn Meeting
October 21-22, 2021

Physics Informed Neural Network for a Nuclear Power Plant

Young Ho Chae[a], Hyeonmin Kim[b], 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*
[*]*Corresponding author: phseong1@kaist.ac.kr*

1. Introduction

Securing the safety of a nuclear power plant (NPP) under the accident situation is extremely important. The most accurate results can be obtained by actually conducting an experiment to understand the behavior of NPP under the accident situation. However, experiment in real world is almost impossible due to time, cost, and risk. Therefore, by modeling the NPP using mathematical and physical knowledge and performing simulations with the model, the accident situation is figured and proper strategies are established. Monte Carlo N-Particle Extended (MCNPX) is the representative simulation method to model transport phenomena of neutron which is developed by Los Alamos National Laboratory (LANL). Multi-dimensional Analysis of Reactor Safety (MARS) is the representative simulation method to model thermal hydraulic behavior which is developed by Korea Atomic Energy Research Institute (KAERI).

Simulation codes are based on the method of numerically analyzing partial differential equations (PDEs) to obtain solutions. Fundamentally, simulation codes operates by dividing the analysis target into unit element (mesh) and numerically solving the equations of the unit element. Due to these fundamental characteristic, simulation codes inherently has a trade-off between accuracy and calculation time. To speed up the calculation, mesh size should be increased. As a result of increased mesh size, the accuracy of solution will be decreased and vice versa.

Therefore, in this study, we suggest a code acceleration method based on a physics informed neural network (PINN). PINN is firstly suggested by M. Raissi et. al. [1]. The PINN is a neural network model that can reflect the form of the equation. The advantage of reflecting the equation in training neural network is that the equation provides a training restriction which can be called prior knowledge. By learning with two different restrictions, the convergence speed is fast and the model shows a fairly robust for extrapolation which is almost impossible in a naive neural network model. Also, unlike a neural network that has to provide the answer of an actual equation during the training process, the PINN is free from data generation because PINN utilizes only the initial conditions and boundary conditions.

In this study, to figure out the applicability of the PINN to NPP analysis code acceleration, two differential equations were solved by using the PINN. First equation is steady state diffusion equation which is closely related with reactor analysis, and the second equation is heat conduction equation which is closely related with thermal hydraulic analysis.

2. Physics Informed Neural Network

The physics informed neural network (PINN) is an algorithm that provides equation which can be called prior knowledge to the loss of neural network. The algorithm firstly proposed by M. Raissi et. al. [1]. The biggest difference between PINN and existing naive neural networks is the type of losses. There are two losses in PINN. The first loss is the difference between the target value and the estimated value (latent vector). The second loss is the loss that occurs when calculating the differential equation based on the calculated latent vector. The schematic diagrams of PINN and Neural network are described in figure 1. As shown in figure, in PINN the equation itself act as a restriction to support training.

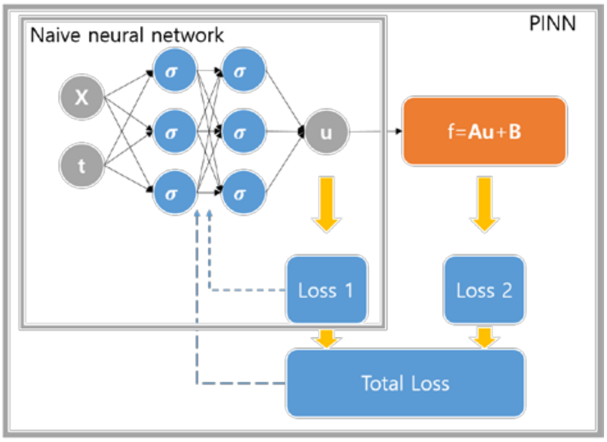

Figure 1 PINN and Naive NN, A is differential equation as form of matrix, and B is non-linear term

3. Application to Differential Equations

To figure out the applicability of PINN to NPP field, two different differential equations are solved by using PINN algorithm. The first is the steady state diffusion equation in one dimension. The second equation is the heat conduction equation in t and x dimension.

3.1 Diffusion equation

Diffusion equation is a simplified version of the transport equation, which describes the flux of neutron. The steady state, 1D diffusion equation with infinite planar source can be written as Eq. 1.

$$\frac{d^2\phi(x)}{dx^2} - \frac{1}{L^2}\phi(x) = 0 \qquad \textbf{Eq. 1}$$

$$\lim_{x\to 0} J(x) = 0 \ \text{(Initial condition)}$$

Where, ϕ is flux and L is diffusion length

The analytic solution of Eq.1 is described in Eq. 2.

$$\phi(x) = \frac{s}{2D} e^{-\frac{abs(x)}{L}} \qquad \textbf{Eq. 2}$$

where, D is diffusion coefficient and s is source

The calculation results from analytic solution and PINN are described in Figure 2.

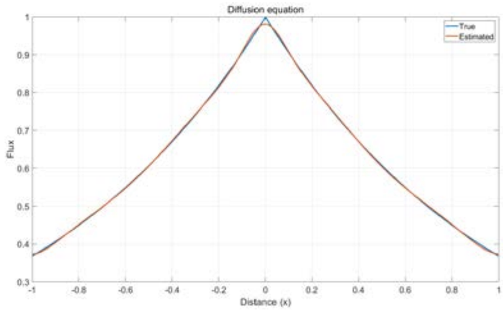

Figure 2 PINN analysis result (Diffusion equation)

The PINN model has small error at x=0, however, in most cases, the PINN shows good results.

3.2 Heat conduction equation

NPPs are operated by transferring heat generated from nuclear fuel to water to boil water. Therefore, conduction and convection is critical phenomenon in NPP. Time dependent 1D conduction equation can be written as Eq.2. Assume that the length of material is L ([-1 1]).

$$\frac{\partial^2 u}{\partial x^2} = \frac{\partial u}{\partial t} \qquad \textbf{Eq. 3}$$

$$u(-1,t) = u(1,t) = 0 \ (Boundary\ condition)$$

$$u(x,0) = 1 \ \text{(Initial condition)}$$

As the input to train the neural network, boundary condition and initial condition vectors were provided. Any information related with the solution does not provided.

The analytic solution of the equation is described in Eq.5.

$$u(x,t) = \frac{1}{\pi}\sum \frac{1-\cos(n\pi)}{n} e^{-\frac{kn^2\pi^2 t}{L^2}} \sin\left(\frac{n\pi x}{L}\right) \qquad \textbf{Eq. 5}$$

The analysis result using PINN is described in Figure 3.

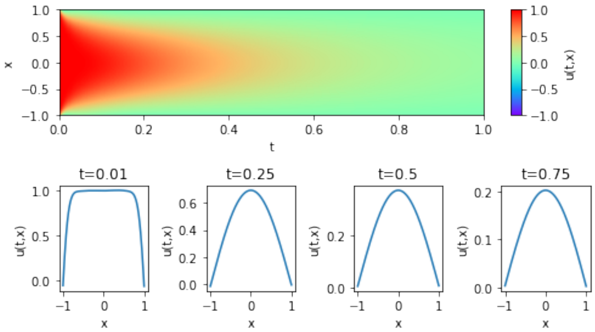

Figure 3 PINN Analysis result (Time dependent heat conduction equation)

Overall mean squared error (MSE) was 10e-9 order.

4. Conclusion

Various analysis codes are used in NPPs to estimate plant status. However, since the differential equation is calculated using a numerical method, the calculation time is considerably long. Therefore, in this study, in order to figure out the applicability of the PINN to accelerate NPP analysis code in the future, several differential equations were solved based on PINN algorithm. As a result, the PINN showed reasonable results. Therefore, it is expected that pre-trained PINN can be applied to accelerate the NPP analysis code.

5. Future works

Accelerating the nuclear power plant analysis code using PINN is expected to ultimately contribute to performing dynamic probabilistic safety assessment (DPSA), which was previously impossible due to calculation time. By performing DPSA, not only can calculate the risk of nuclear power plants due to dynamic effects, but also the structural limitations of conventional static PSA, such as grouping order or the need to set success criteria, can be avoided.

Various methods have been developed to perform DPSA. Representative examples are Monte Carlo simulation, dynamic event tree, dynamic flow graph, and Markov modeling. However, Monte Carlo simulation and Markov model are impossible to calculate complex system because the branch increases explosively. The dynamic flow graph can only identify trends of parameters and the dynamic event tree has bottleneck in calculation speed of the physical process model. In case of NPP, the calculation speed of MARS is not sufficient for the dynamic event tree method.

If it is possible to accelerate the MARS code based on PINN, it is expected to be able to perform dynamic event tree based DPSA analysis by accelerating the calculation of plant status.

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)

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