Solving Neutron Diffusion Equations Using Physics-Informed Neural Networks: Performance Analysis Against Traditional FEM

Applied Artificial Intelligence Section, KAERI

Yohan Lee, Byoungil Jeon*, Yonggyun Yu

*corresponding author:
bijeon@kaeri.re.kr

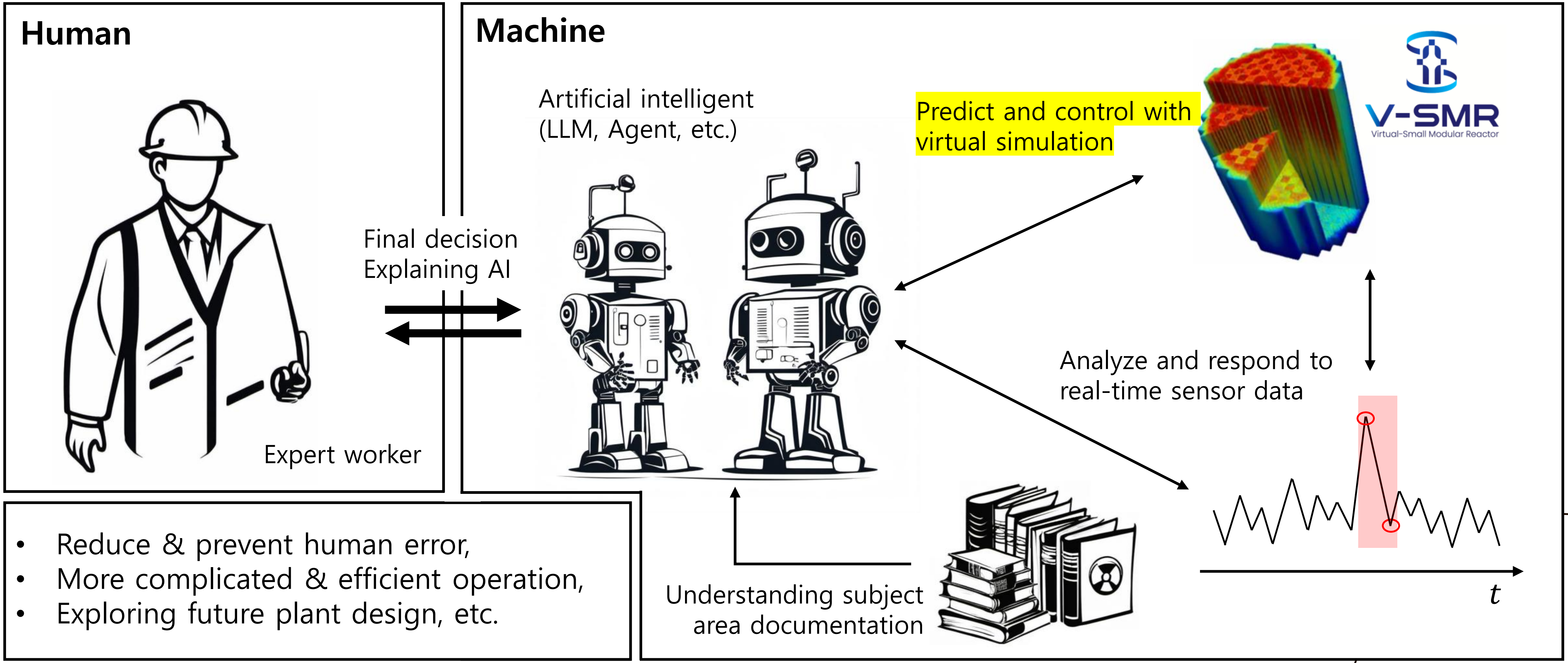
Introduction

Exploring Physics-Informed Neural Networks as an Innovative Approach for Reactor Core Analysis.

Moving Beyond Traditional Numerical Methods to Address Computational Challenges in Nuclear Engineering Design and Safety Assessment.

Research Motivation

Computational Challenges in Nuclear Engineering

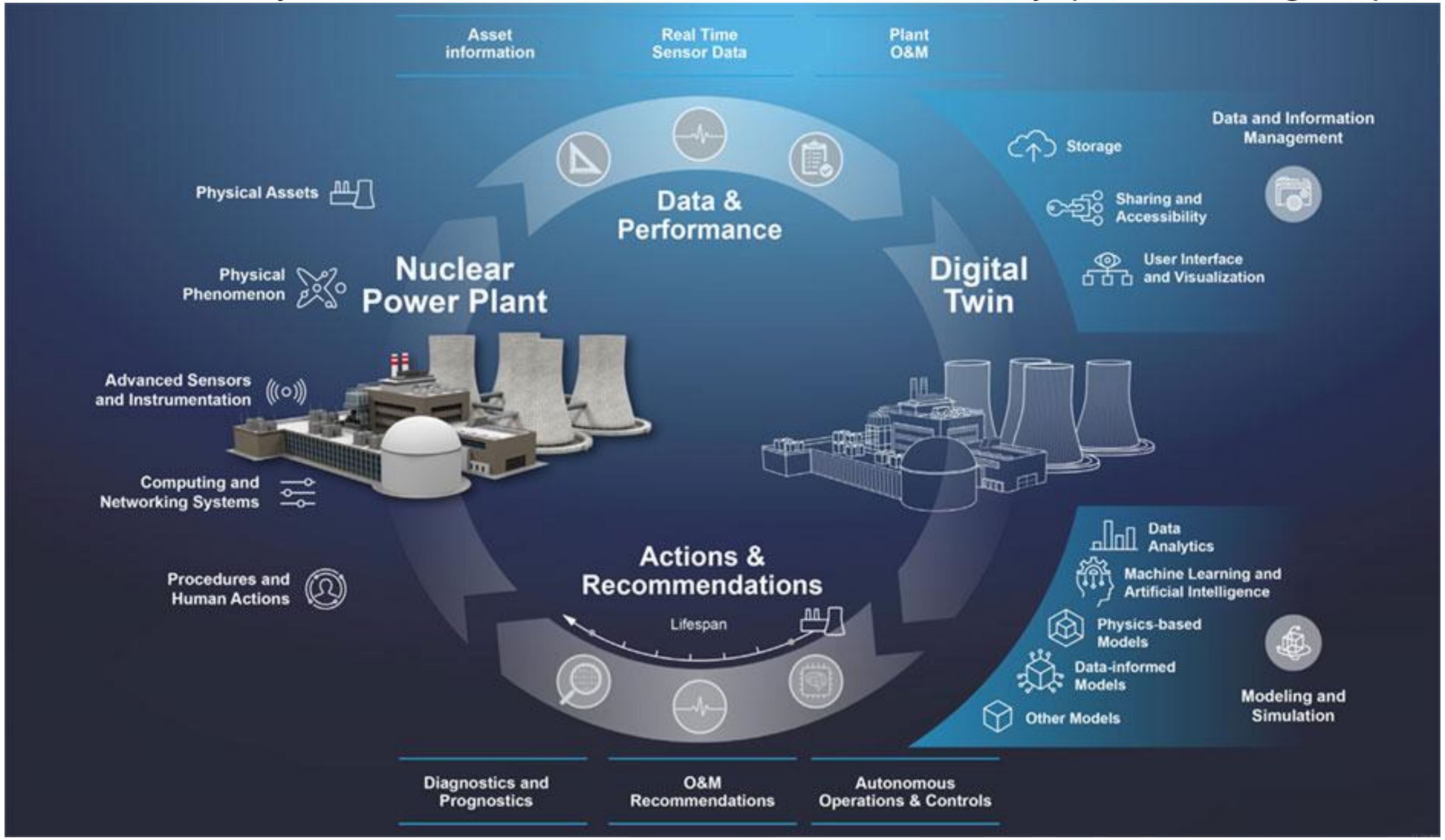


- Reduce & prevent human error,
- More complicated & efficient operation,
- Exploring future plant design, etc.

Research Motivation

Computational Challenges in Nuclear Engineering

High-speed nuclear reactor analysis is necessary for virtual reactor technology.
However, conventional analysis methods are limited in their ability perform high-speed simulations.



Research Motivation

Computational Challenges in Nuclear Engineering

High-speed nuclear reactor analysis is necessary for virtual reactor technology.
However, conventional analysis methods are limited in their ability perform high-speed simulations.

Objectives and Approach

Combining Physics with Deep Learning

Probabilistic solution (Monte-Carlo, etc.)

Simulating individual neutron behavior through stochastic sampling

- Based on tracking neutron interactions through random number sampling
- Minimal approximations in physics; considered "exact" reference solution
- Handles complex geometries and detailed physics with high fidelity

Deterministic solution (FEM, FDM, etc.)

Solving discretized forms of governing equations with mathematical rigor

- Based on spatial and energy discretization of transport/diffusion equations
- Provides continuous flux distributions with guaranteed convergence
- Widely established in industry and regulatory frameworks

Objectives and Approach

Combining Physics with Deep Learning

Probabilistic solution (Monte-Carlo, etc.)

Simulating individual neutron behavior through stochastic sampling

- Based on tracking neutron interactions through random number sampling
- Minimal approximations in physics; considered "exact" reference solution
- Handles complex geometries and detailed physics with high fidelity

Machine learning solution (PINN, etc.)

Embedding physical laws into neural network architectures

- Since PINN embeds physics, it is more reliable than pure data-driven methods
- Offers continuous, differentiable solutions with adaptive resolution capability
- Emerging approach with potential for computational efficiency at scale

Deterministic solution (FEM, FDM, etc.)

Solving discretized forms of governing equations with mathematical rigor

- Based on spatial and energy discretization of transport/diffusion equations
- Provides continuous flux distributions with guaranteed convergence
- Widely established in industry and regulatory frameworks

PINN combine **the mathematical rigor of physical laws** with **the flexibility and computational advantages** of neural networks, offering a promising middle ground between traditional deterministic methods and Monte Carlo simulations.

The key value proposition lies in providing continuous, parameterized solutions that facilitate rapid design exploration, enable seamless data fusion, and potentially reduce computational burden for specific reactor analysis applications without sacrificing essential physics fidelity.

Methodology

Developing a Hybrid PINN Framework that Integrates Physical Constraints with Neural Network Flexibility.

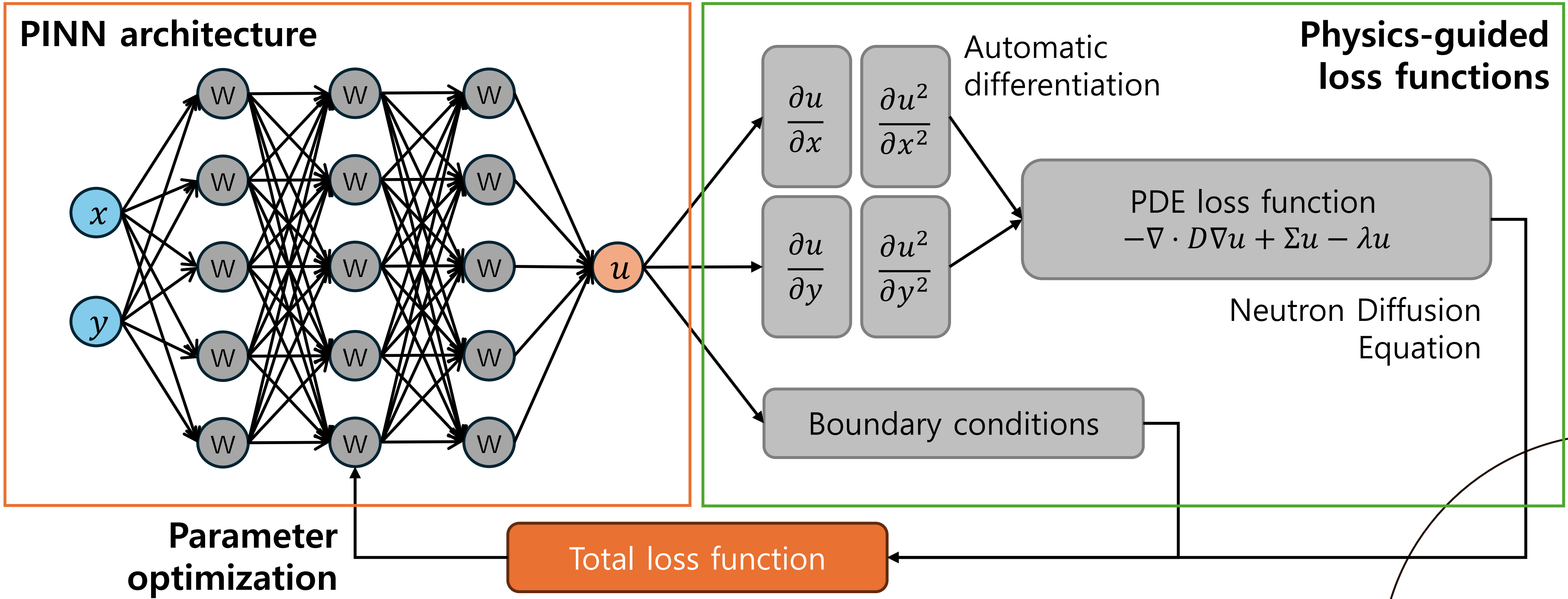
Incorporating Prior FDM Solutions and Rayleigh Quotient Formulation to Enhance Training Stability and Solution Accuracy.

Tailoring Network Architecture for the Complexity of Neutron Diffusion Eigenvalue Problems.

Fundamentals of PINN

Basic principle of Physics-Informed Neural Network

PINN fundamentally operate by **incorporating differential equations directly into the neural network's loss function**, enabling the model to learn solutions that simultaneously fit available data and satisfy governing physical laws.



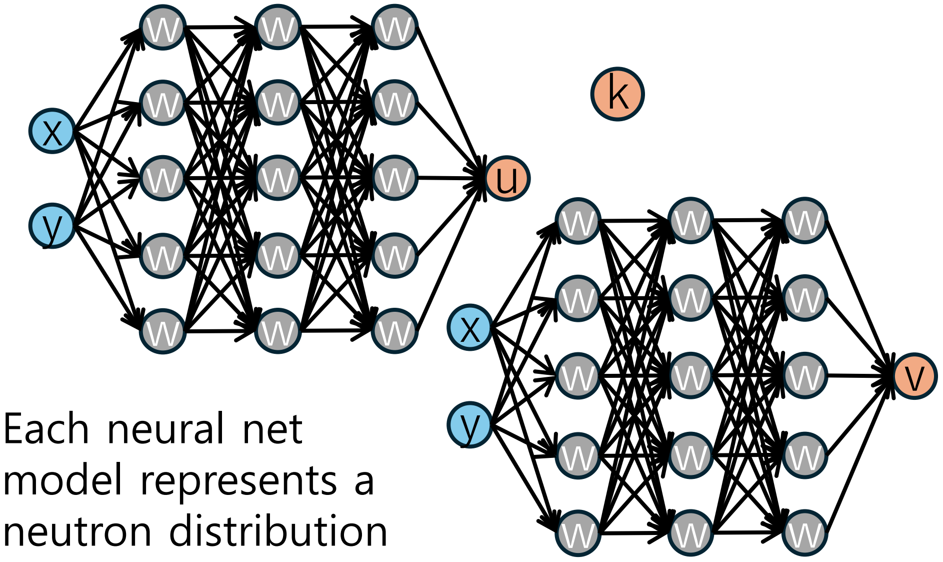
PINN for solving diffusion equation

Embedding Governing Equations

- For solving the **2D 2-group neutron diffusion equation**, the model structure was defined as follows:
  - The MLP model receives two-dimensional coordinate values as input.
  - Each MLP model outputs the neutron distribution of a single group. That is, two MLP models are used here to solve the two-group neutron diffusion equation.
  - In order to accurately and stably calculate the neutron distribution and effective multiplication factor k_{eff}, we utilize a combination of pre-calculation using FDM and the Rayleigh quotients method.

When the 2D neutron distribution is as follows,

$$u = \phi_1(x, y)$$
$$v = \phi_2(x, y)$$



Each neural net model represents a neutron distribution

- Fast neutrons

$$-\nabla \cdot D_1(\vec{r})\nabla u + \left(\Sigma_{a,1}(\vec{r}) + \Sigma_{s,1\to 2}(\vec{r})\right) u = \frac{1}{k}\left[\nu\Sigma_{f,1}(\vec{r})u + \nu\Sigma_{f,2}(\vec{r})v\right]$$

- Thermal neutrons

$$-\nabla \cdot D_2(\vec{r})\nabla v + \Sigma_{a,2}(\vec{r})v = \Sigma_{s,1\to 2}(\vec{r})u$$

- Boundary conditions

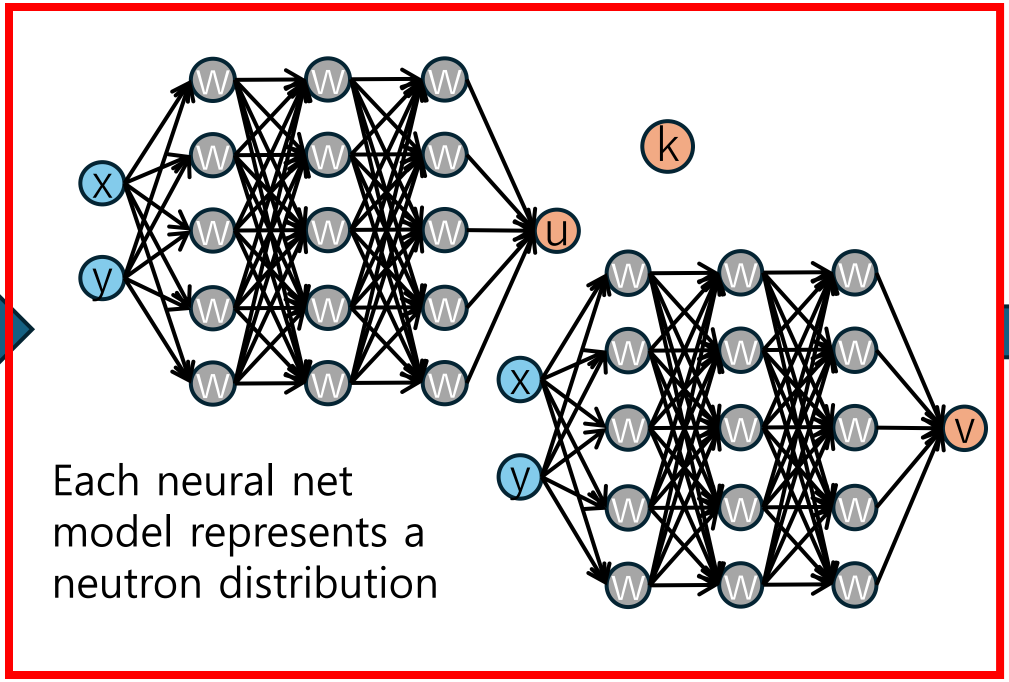
PINN for solving diffusion equation

Embedding Governing Equations

- For solving the **2D 2-group neutron diffusion equation**, the model structure was defined as follows:
  - The MLP model receives two-dimensional coordinate values as input.
  - Each MLP model outputs the neutron distribution of a single group. That is, two MLP models are used here to solve the two-group neutron diffusion equation.
  - In order to accurately and stably calculate the neutron distribution and effective multiplication factor k_{eff}, we utilize a combination of pre-calculation using FDM and the Rayleigh quotients method.

When the 2D neutron distribution is as follows,

$$u = \phi_1(x, y)$$
$$v = \phi_2(x, y)$$



- Fast neutrons

$$-\nabla \cdot D_1(\vec{r})\nabla u + \left(\Sigma_{a,1}(\vec{r}) + \Sigma_{s,1\to 2}(\vec{r})\right)u = \frac{1}{k}\left[\nu\Sigma_{f,1}(\vec{r})u + \nu\Sigma_{f,2}(\vec{r})v\right]$$

- Thermal neutrons

$$-\nabla \cdot D_2(\vec{r})\nabla v + \Sigma_{a,2}(\vec{r})v = \Sigma_{s,1\to 2}(\vec{r})u$$

- Boundary conditions

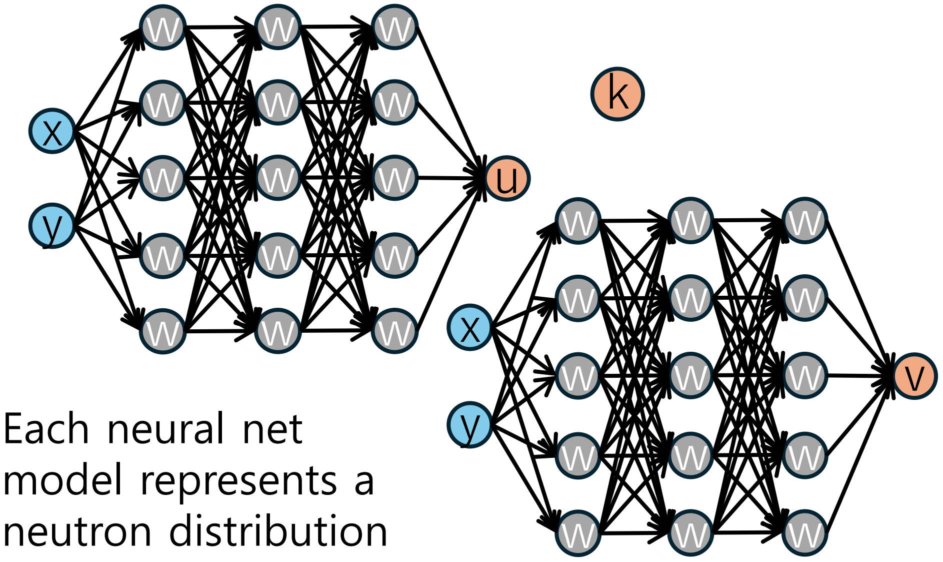
PINN for solving diffusion equation

Embedding Governing Equations

- For solving the **2D 2-group neutron diffusion equation**, the model structure was defined as follows:
  - The MLP model receives two-dimensional coordinate values as input.
  - Each MLP model outputs the neutron distribution of a single group. That is, two MLP models are used here to solve the two-group neutron diffusion equation.
  - In order to accurately and stably calculate the neutron distribution and effective multiplication factor k_{eff}, we utilize a combination of pre-calculation using FDM and the Rayleigh quotients method.

When the 2D neutron distribution is as follows,

$$u = \phi_1(x, y)$$
$$v = \phi_2(x, y)$$



Each neural net model represents a neutron distribution

- Fast neutrons

$$-\nabla \cdot D_1(\vec{r})\nabla u + \left(\Sigma_{a,1}(\vec{r}) + \Sigma_{s,1\rightarrow 2}(\vec{r})\right) u = \frac{1}{k}\left[\nu\Sigma_{f,1}(\vec{r})u + \nu\Sigma_{f,2}(\vec{r})v\right]$$

- Thermal neutrons

$$-\nabla \cdot D_2(\vec{r})\nabla v + \Sigma_{a,2}(\vec{r})v = \Sigma_{s,1\rightarrow 2}(\vec{r})u$$

- Boundary conditions

PINN for solving diffusion equation

Integrating Physical Constraints using FDM pre-calculation

- For solving the **2D 2-group neutron diffusion equation**, the model structure was defined as follows:
  - The MLP model receives two-dimensional coordinate values as input.
  - Each MLP model outputs the neutron distribution of a single group. That is, two MLP models are used here to solve the two-group neutron diffusion equation.
  - In order to accurately and stably calculate the neutron distribution and effective multiplication factor k_{eff}, we utilize a combination of pre-calculation using FDM and the Rayleigh quotients method.

Loss function

$$\mathcal{J}(\theta) = \mathcal{J}_{PDE}(\theta) + \mathcal{J}_{BC}(\theta) + \mathcal{J}_{prior}(\theta)$$

$$\mathcal{J}_{PDE}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left[\left|F_{res,fast}\right|^2 + \left|F_{res,thermal}\right|^2\right]$$

$$\mathcal{J}_{prior}(\theta) = \alpha\frac{1}{M}\sum_{j=1}^{M}\left[\left|\phi_1(\vec{r}_j) - \phi_{1,FDM}(\vec{r}_j)\right|^2 + \left|\phi_2(\vec{r}_j) - \phi_{2,FDM}(\vec{r}_j)\right|^2\right]$$

$$\mathcal{J}_{prior}(\theta) = \frac{1}{K}\sum_{i=1}^{K}\left\{\left[\phi_1(x_{i,prior}, y_{i,prior}) - \phi_{1,FDM}(x_{i,prior}, y_{i,prior})\right]^2 + \left[\phi_2(x_{i,prior}, y_{i,prior}) - \phi_{2,FDM}(x_{i,prior}, y_{i,prior})\right]^2\right\}$$

This loss function indicates that the task is to find a DNN model that outputs values satisfying the neutron diffusion equation (PDE), the desired geometric conditions (BC), and the pre-calculated FDM results (Prior) for all points in the two-dimensional domain.

Mathematical Formulation of Hybrid PINN

Calculate k-effective with Rayleigh quotients method

- For solving the **2D 2-group neutron diffusion equation**, the model structure was defined as follows:
  - The MLP model receives two-dimensional coordinate values as input.
  - Each MLP model outputs the neutron distribution of a single group. That is, two MLP models are used here to solve the two-group neutron diffusion equation.
  - In order to accurately and stably calculate the neutron distribution and effective multiplication factor k_{eff}, we utilize a combination of pre-calculation using FDM and the Rayleigh quotients method.

To ensure the reliability of the inference of the core multiplier k_{eff} value and to provide a solid mathematical and background for the results, we propose a hybrid method that calculates k_{eff} using **the Rayleigh Quotients** method applied to the inferred neutron distribution.

$$\boldsymbol{M}\vec{\phi} = \frac{1}{k_{eff}}\boldsymbol{F}\vec{\phi} \longrightarrow \begin{pmatrix} -\nabla\cdot D_1\nabla + (\Sigma_{a1}+\Sigma_{s12}) & 0 \\ -\Sigma_{s12} & -\nabla\cdot D_2\nabla + \Sigma_{a2} \end{pmatrix}\begin{pmatrix}\phi_1\\ \phi_2\end{pmatrix} = \frac{1}{k_{eff}}\begin{pmatrix} \nu\Sigma_{f1} & \nu\Sigma_{f2} \\ 0 & 0 \end{pmatrix}\begin{pmatrix}\phi_1\\ \phi_2\end{pmatrix}$$

$$k_{eff} = \frac{\langle\vec{\phi},\boldsymbol{F}\vec{\phi}\rangle}{\langle\vec{\phi},\boldsymbol{M}\vec{\phi}\rangle} = \frac{\sum_i\left(\phi_1^i\cdot\nu\Sigma_{f1}^i\cdot\phi_1^i + \phi_2^i\cdot\nu\Sigma_{f2}^i\cdot\phi_2^i\right)}{\sum_i\left(\phi_1^i\cdot M_1^i + \phi_2^i\cdot M_2^i\right)}$$

$$^{*}M_1 = \Sigma_{a1}\phi_1 - \nabla\cdot D_1\nabla\phi_1 + \Sigma_{s12}\phi_1$$
$$M_2 = \Sigma_{a2}\phi_2 - \nabla\cdot D_2\nabla\phi_2 - \Sigma_{s12}\phi_1$$

Results & Conclusion

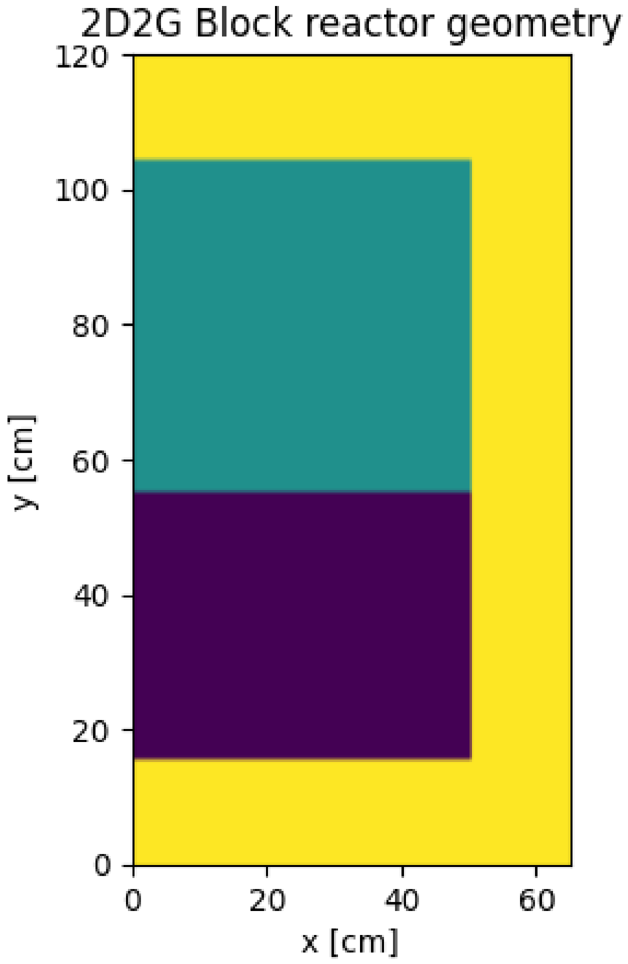
Comparative Analysis of Flux Distributions and K-effective Calculations Across FEM, Standard PINN, and Hybrid PINN Methodologies.

Quantifying Performance Improvements at Material Interfaces and Convergence Characteristics.

Evaluating Computational Efficiency and Accuracy Trade-offs for Practical Reactor Physics Applications.

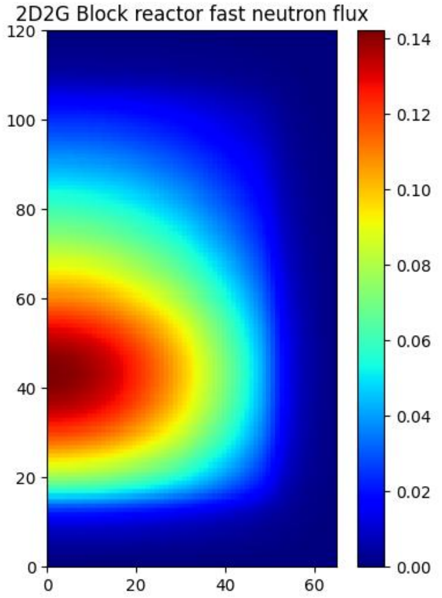
Benchmark Problem

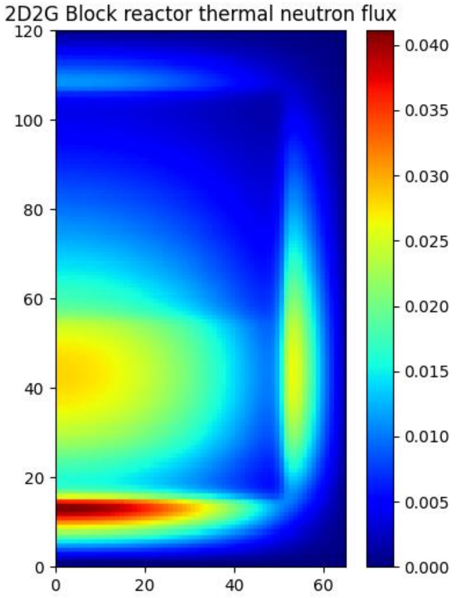
2D Reactor Configuration with Multiple Fuel Regions



Region	Material	D_1 (cm)	D_2 (cm)	$\Sigma_{a,1}$ (cm^{-1})	$\Sigma_{a,2}$ (cm^{-1})	$\Sigma_{a,1\to 2}$ (cm^{-1})	$\nu\Sigma_{f,1}$ (cm^{-1})	$\nu\Sigma_{f,2}$ (cm^{-1})
	Fuel 1	1.2670	0.3540	0.0121	0.1210	0.0241	0.0085	0.1851
	Fuel 2	1.2800	0.4000	0.0100	0.1000	0.0160	0.0060	0.1500
	Reflector	1.1300	0.1660	0.0004	0.0200	0.0493	0.0000	0.0000

$$k_{eff} = 1.14233$$

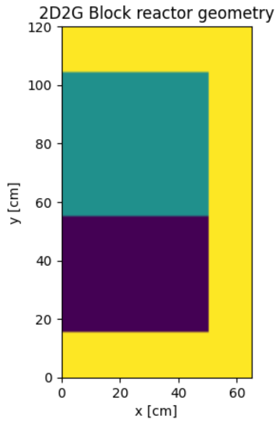




Benchmark Problem

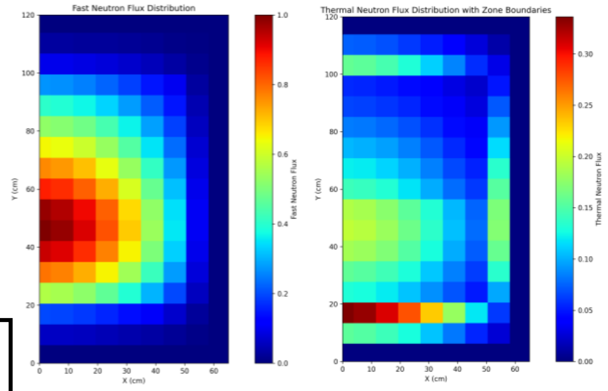
Example flow diagram solving benchmark problem

FDM



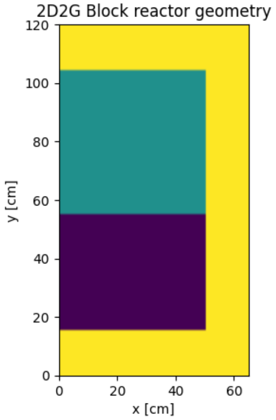
Benchmark Problem

Example flow diagram solving benchmark problem



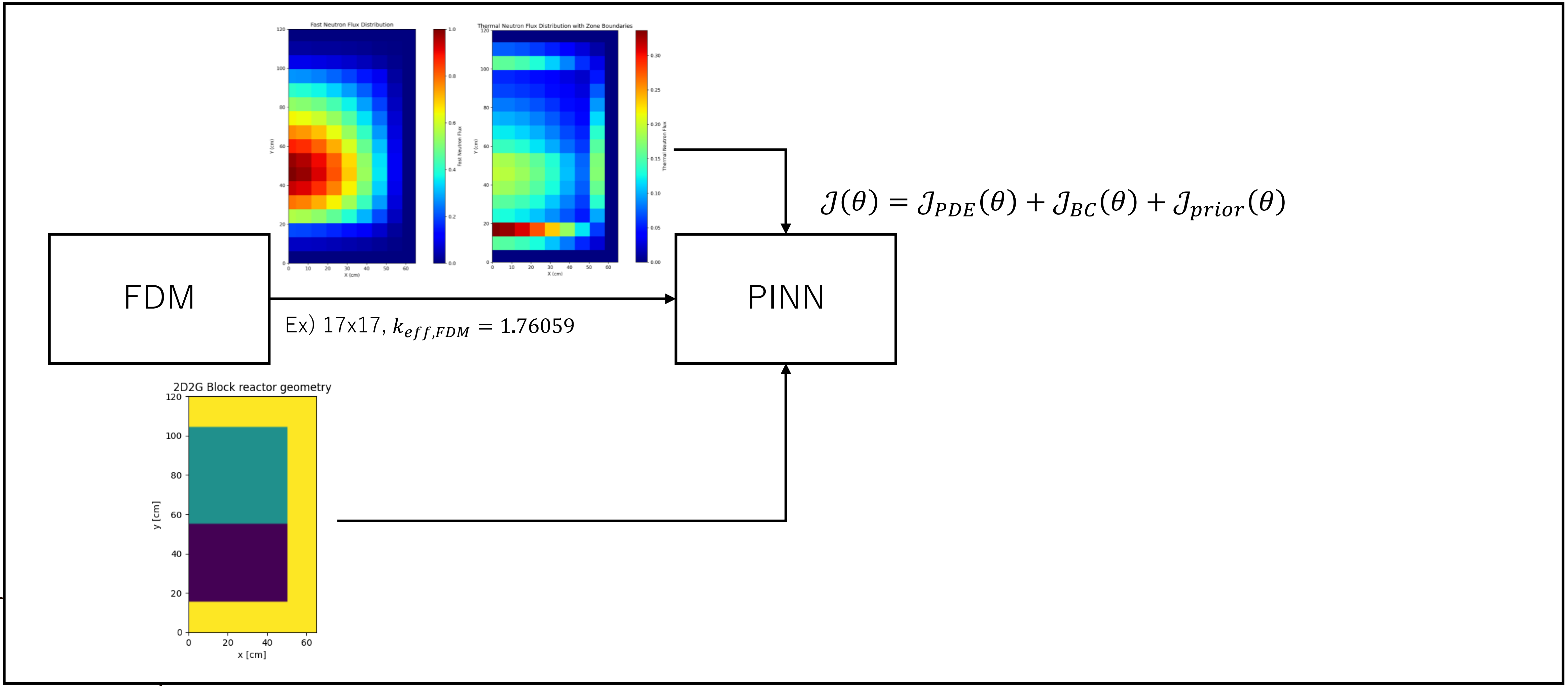
FDM

Ex) 17x17, $k_{eff,FDM} = 1.76059$



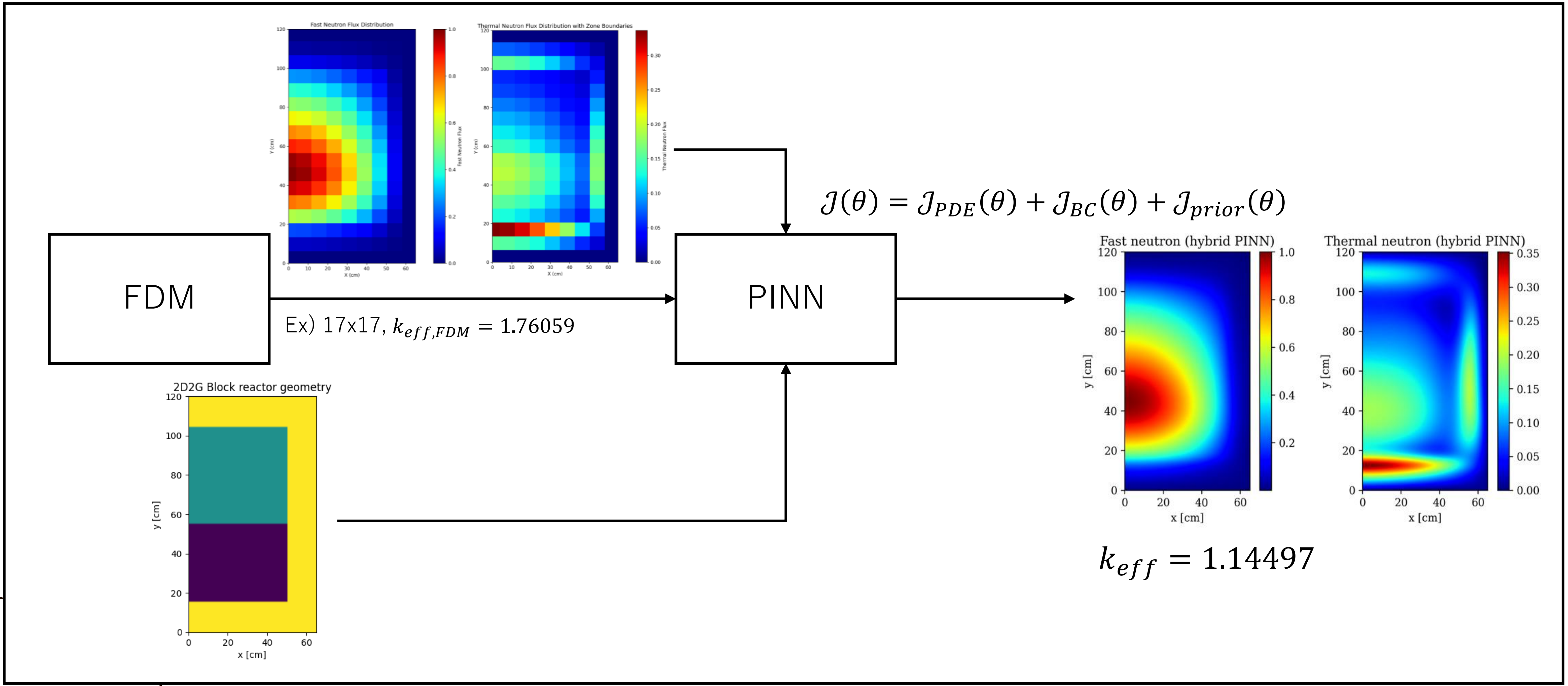
Benchmark Problem

Example flow diagram solving benchmark problem



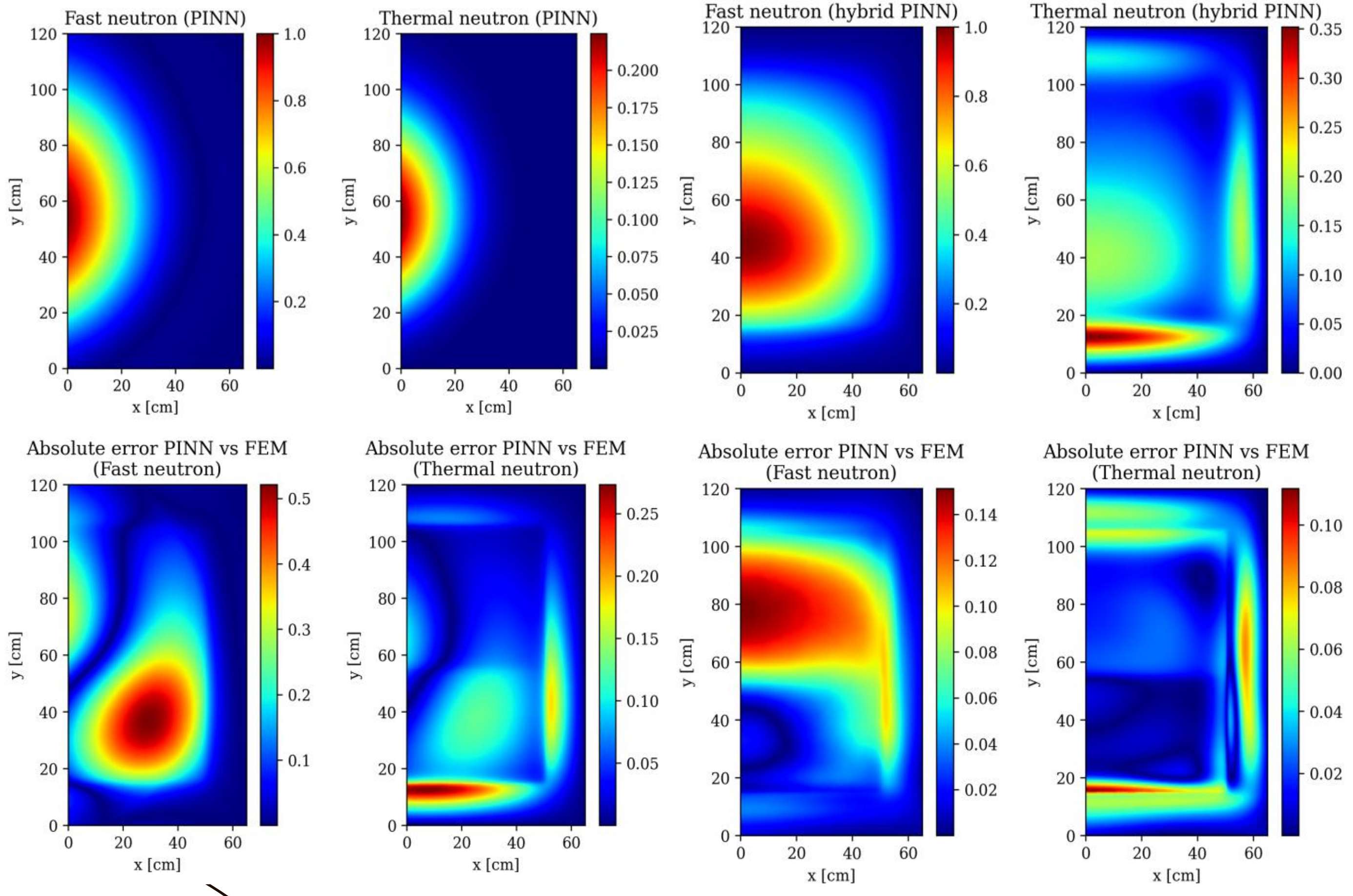
Benchmark Problem

Example flow diagram solving benchmark problem



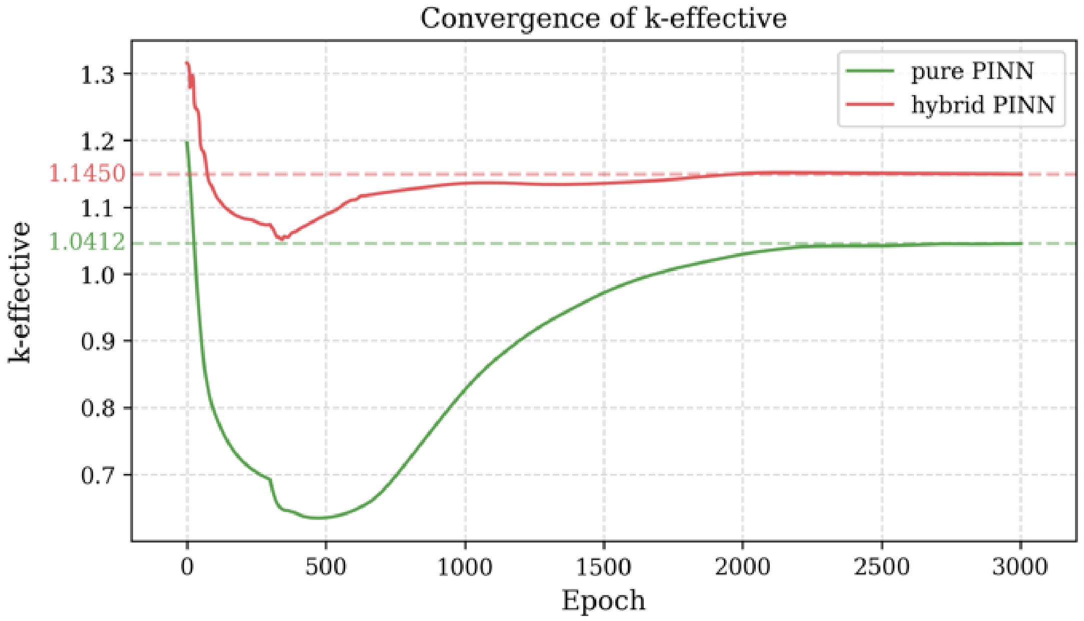
Hybrid PINN Training Results

Neutron Flux comparison



K-effective Analysis

	FEM	PINN	Hybrid PINN
k_{eff}	1.14233	1.04128	1.14497
Error (|pcm|)	-	10105	264
Time	< 30 sec	< 10 min	< 10 min



Conclusions

Summary

- We are attempting a deep learning PINN-based approach for high-precision, high-speed core analysis methods for next-generation nuclear reactor research and development.
- A hybrid strategy of training PINN using prior information obtained from rough FDM calculations to improve the learning stability of deep learning models and derive physically and mathematically meaningful results.

Limitations and Future works

- In core analysis involving a wide domain and complex shapes and physics, obtaining prior information using FDM can act as a bottleneck.
- Conduct future research by refining the methodology to make it more versatile and conducting verification using diverse and practical benchmark problems.

Thank you