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

Preliminary CFD Benchmarking of L-DeepONet for Nuclear Reactor Digital Twin

Minseo Lee [a], Sangam Khanal [b], Shilaj Baral [b], Minseop Song [c], Chaehyeon Song [c], Joongoo Jeon [a,b,d*]
[a] *Department of Applied Plasma and Quantum Beam Engineering, Jeonbuk National University*
[b] *Graduate School of Integrated Energy-AI, Jeonbuk National University,*
567, Baekje-daero, Deokjin-gu, Jeonju-si, Republic of Korea
[c] *Department of Nuclear Engineering, Hanyang University,*
Wangsimni-ro, Seongdong-gu, Seoul, Republic of Korea
[d] *Department of Quantum System Engineering, Jeonbuk National University,*
567, Baekje-daero, Deokjin-gu, Jeonju-si, Republic of Korea
[*] *Corresponding author: *jgjeon41@jbnu.ac.kr*

***Keywords :** computational fluid dynamics, deep learning, small modular reactor, deeponet, helical coil steam generator

1. Introduction

Small Modular Reactors (SMRs) are garnering global attention as a transformative solution for sustainable and clean energy production. Small Modular Reactors (SMRs) have garnered attention as a technology capable of producing sustainable and environmentally friendly energy. Compared to traditional large-scale reactors, SMRs offer advantages such as enhanced safety, modular design, and flexible power generation. However, they also face practical challenges, including higher generation costs and limited construction and operational experience. To overcome these limitations and maximize SMR efficiency, computational fluid dynamics (CFD) has been widely utilized for optimization studies.

CFD is a powerful tool for analyzing fluid flow behavior in physical systems. However, high-resolution simulations present a significant challenge due to the exponentially increasing computational cost. In particular, the Navier-Stokes equations lack an analytical solution, necessitating the use of numerical methods, which impose high computational demands [1]. Traditional numerical approaches, such as the finite volume method (FVM), require high-resolution meshes, making real-time CFD simulations for complex geometries impractical. Consequently, machine learning (ML)-based CFD acceleration techniques have emerged as a crucial research area [2,3].

Representative recent studies have leveraged AI models such as physics-informed neural networks (PINNs) [4] and deep operator networks (DeepONet) [5] to accelerate CFD simulations, demonstrating superior computational efficiency compared to conventional numerical methods. Especially, Jeon et al. developed a finite volume method network (FVMN) model to predict a CFD time series by introducing FVM principles into the network architecture and the loss function [6]. However, existing research still faces several challenges, including: (1) limited applicability to complex geometries, (2) accuracy and stability issues in long-term simulations, and (3) a lack of validation for specialized systems such as nuclear reactors.

This study applies the Latent Deep Neural Operator (L-DeepONet) to develop an AI-based surrogate model for CFD acceleration and evaluate its performance[7]. L-DeepONet extends DeepONet by incorporating an autoencoder structure, compressing high-dimensional CFD data into a lower-dimensional latent space to enable more efficient learning and prediction [8]. This approach allows for accurate approximation of complex flow fields with significantly lower computational costs compared to conventional numerical methods.

2. Methods

In this section, the model architecture, training methodology, and data preprocessing techniques used in this study are described in detail.

2.1 Latent Space based DeepONet

DeepONet is an operator learning model based on the universal approximation theorem [9], designed to map input functions to output functions. It consists of two sub-networks: the branch net, which trains input function characteristics, and the trunk net, which maps spatial-temporal coordinates to output function. The final prediction is obtained by combining outputs from both networks. DeepONet has demonstrated high accuracy, fast convergence, and reduced generalization error compared to conventional neural networks.

As a data-driven model, DeepONet requires CFD data for training. CFD simulations require significant computational costs, especially in thermal-hydraulic analysis of nuclear reactors, where complex geometries demand extensive calculation resources. To address this challenge, L-DeepONet is introduced, which trains DeepONet in a latent space using a dimension-reduction neural network. This approach leverages a pretrained autoencoder’s encoder to compress high-dimensional CFD data into a lower-dimensional space for efficient training. After training, a decoder reconstructs the original high-dimensional data.

By learning high-dimensional CFD data in a reduced space, L-DeepONet significantly reduces calculation time and GPU resource requirements, making it a promising surrogate model for fluid simulation in SMRs. The architecture of L-DeepONet is illustrated in Figure 1.

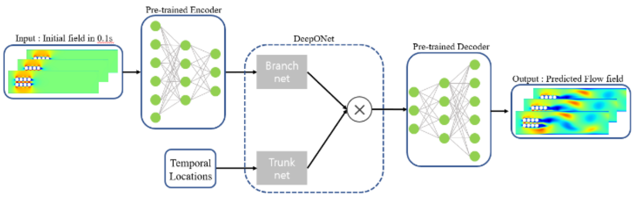

Figure 1. L-DeepONet architecture for CFD applications

2.2 Datasets

A two-dimensional (2D) transient CFD dataset was contructed to analyze the primary-side cross-flow in a helical coil steam generator (HCSG) within a SMRs. This study is based on the SMART reactor design parameters from the Korea Atomic Energy Research Institute (KAERI). ANSYS SpaceClaim 24.2.0 was used for geometry generation, while ANSYS Fluent 24.2.0 was employed for flow simulations. Simulations were conducted under inlet velocity conditions of 0.05 m/s, 0.1 m/s, 0.15 m/s, 0.2 m/s, and 0.3 m/s, with an outlet gauge pressure of 0 Pa and no-slip boundary conditions on all solid walls. Water was used as the working fluid, and the Shear Stress Transport (SST) k-ω model was applied for turbulence modeling. The dataset primarily focuses on analyzing the formation of the Karman vortex street in the wake region behind the tube bundle.

In Geometry 1, Karman vortex shedding occurs at lower inlet velocities (0.05–0.2 m/s), but at 0.3 m/s, the flow stabilizes. This phenomenon is attributed to its straight-row tube arrangement, longer radial length (306 mm), and shorter axial length (77 mm), which promote streamlined flow at high Reynolds numbers. It is illustrated below Figure 2.

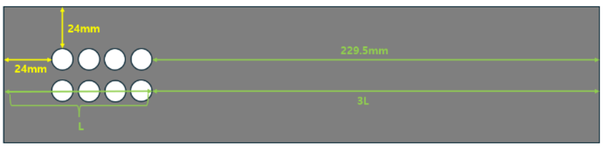

Figure 2. Domain of Geometry 1

In contrast, as shown in Figure 3, Geometry 2 exhibits persistent Karman vortex shedding across all inlet velocities, with strong wake interactions. Its staggered tube arrangement, shorter radial length (252 mm), and longer axial length (94 mm) contribute to increased turbulence and mixing effects.

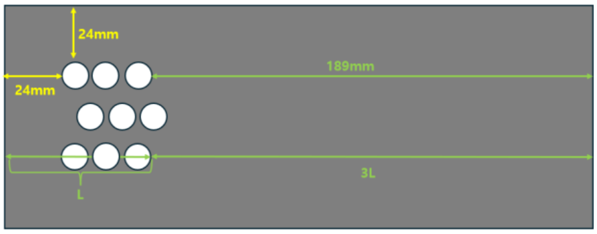

Figure 3. Domain of Geometry 2

In conclusion, Geometry 1 tends to develop stable flow at high velocities, whereas Geometry 2 maintains continuous vortex shedding. This study provides fundamental data for understanding HCSG flow dynamics and optimizing reactor design.

2.3 Training method

To facilitate the training of the AutoEncoder and DeepONet, we performed a preprocessing step on the CFD dataset, which is detailed in Section 2.2. The dataset includes flow fields with inlet velocities of 0.05 m/s, 0.1 m/s, and 0.2 m/s as initial conditions, along with velocity magnitude data across all time steps. Specifically, we utilized simulation data over 10s with 100 timesteps, each with a size of 0.1. The hyperparameters, such as the number of training epochs, used for model learning are summarized in Table 1 below.

Table I: Hyperparameter used in learning L-DeepONet

	Epoch	Batch size	Learning rate	Loss
AutoEncoder	10000	90	10^{-3}	MSE
DeepONet	90000	3	10^{-4}	MSE

3. Results

We measured and compared the interpolation and extrapolation performance for initial conditions of 0.15 m/s and 0.3 m/s, respectively. First in Geometry 1, the model showed poor performance at an inlet velocity of 0.15 m/s. The model failed to accurately predict the flow field even at the initial stage t = 0.1s, and this issue persisted throughout the simulation. This indicates that the model did not sufficiently learn the flow characteristics at this particular inlet velocity. However, at a higher inlet velocity of 0.3 m/s, the model demonstrated improved predictions. Although the velocity scale and trends were not perfectly accurate, the prediction is similar to the stable flow which is seen in the CFD data. Nevertheless, Prediction is deviate the stable state after 95 time step.

For Geometry 2, at an inlet velocity of 0.15 m/s, the model achieved higher accuracy compared to Geometry 1. It successfully captured the flow structures around the cylinder arrays and effectively reproduced the periodic nature of the Karman vortex street in the wake region.

Likewise, at a higher velocity of 0.3 m/s, the model's prediction can capture the Karman vortex flow patterns.

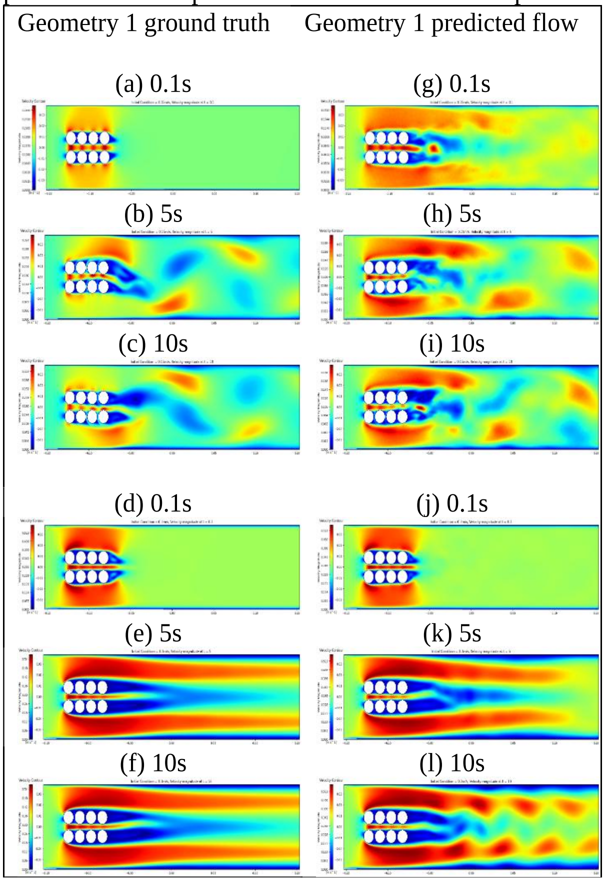

Figure 3. Ground truth and predict flow in Geometry 1. (a), (b), (c), (g), (h), and (i) correspond to an inlet velocity of 0.15 m/s, while (d), (e), (f), (j), (k), and (l) correspond to an inlet velocity of 0.3 m/s.

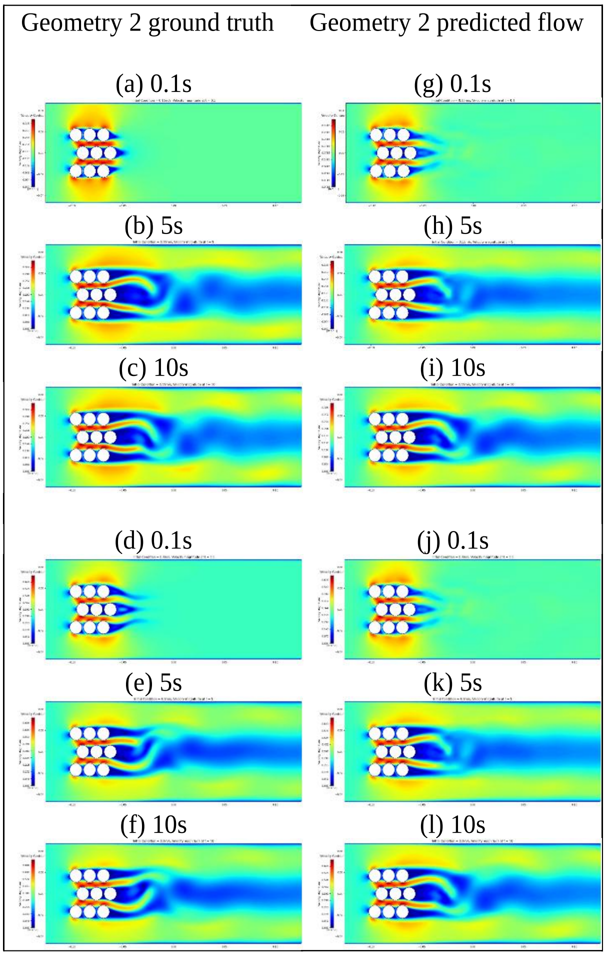

Figure 4. Ground truth and predict flow in Geometry 2. (a), (b), (c), (g), (h), and (i) correspond to an inlet velocity of 0.15 m/s, while (d), (e), (f), (j), (k), and (l) correspond to an inlet velocity of 0.3 m/s.

Additionally, we analyzed velocity variations at specific node locations over time. In both Geometry 1 and Geometry 2, velocity magnitude variations over time were analyzed at three selected node locations. In Geometry 1, node 14104 is positioned 50 mm to the left, node 14318 is located at the geometric center, and node 14282 is located 50 mm to the right of the geometric center. For, Geometry 2, node 14380 is positioned 50 mm to the left, node 20504 is located at the geometric center, and node 14203 is located 50 mm to the right of the geometric center.

In Geometry 1 at 0.15 m/s, we can confirm the graph patterns can't follow the ground truth. On the other hand at 0.3 m/s, velocity magnitude scale is inaccurate but the model showed similarities to the ground truth at certain locations. Geometry 2 showed better overall predictions, capturing general flow trends effectively. Notably, at node 20504, despite inaccuracies in magnitude, the model was capable of reproducing the characteristic periodic velocity fluctuations. These results are illustrated in Figures 5 to 8 below.

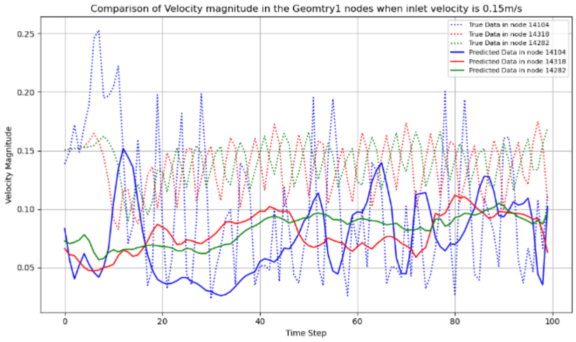

Figure 5. Comparison of Velocity magnitude in the Geometry 1 nodes when inlet velocity is 0.15 m/s.

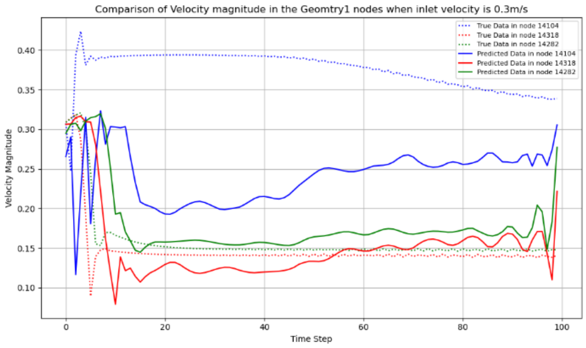

Figure 6. Comparison of Velocity magnitude in the Geometry 1 nodes when inlet velocity is 0.3 m/s.

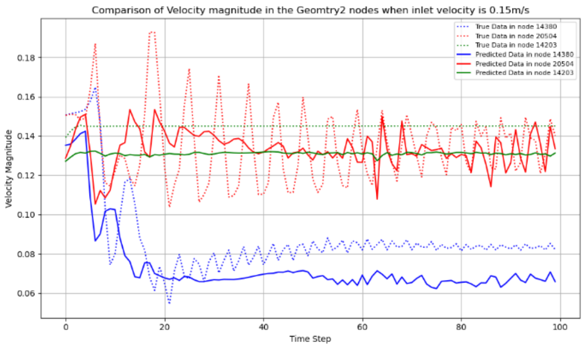

Figure 7. Comparison of Velocity magnitude in the Geometry 2 nodes when inlet velocity is 0.15 m/s.

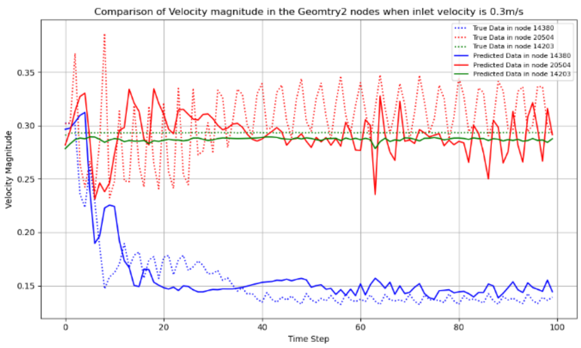

Figure 8. Comparison of Velocity magnitude in the Geometry 2 nodes when inlet velocity is 0.3 m/s.

In summary, our L-DeepONet exhibited superior interpolation and extrapolation performance in more complex geometries compared to simpler ones. Although it did not achieve optimal predictive accuracy, it successfully learned the periodic patterns. In particular, Geometry 2 demonstrated better performance despite its more complex shape, which can be attributed to the stronger periodicity of the Kármán vortex flow in the training data of Geometry 2 compared to Geometry 1. These results suggest that if L-DeepONet sufficiently learns continuous flow characteristics, its performance can be further improved.

4. Conclusions

A preliminary study applying the L-DeepONet model to a partial geometry of the Helical Coil Steam Generator (HCSG) demonstrated that it effectively accelerates CFD simulations even for complex geometries, despite being trained on a limited dataset. These findings suggest the potential applicability of this CFD acceleration methodology to realistic, full-scale SMR geometries. Future studies incorporating diverse initial and boundary conditions are expected to further enhance the generalization capability of the model. In addition, 3D expansion is essential so that it can be applied to actual SMR geometries. In Junyan He et al [10]and Ali Rabeh et al [11], signed distance function were used to show the enough generalization performance of the DeepONet model in three dimensions. However, most studies like these are steady-state analysis. Hence our goal is to expansion these method to transient analysis for simulation the 3D SMR geometries.

ACKNOWLEDGMENT

This work was supported by the National Research Council of Science & Technology (NST) grant by the Korea government (MIST) (No. GTL24031-000) and the Nuclear Safety Research Program through the Korea Foundation of Nuclear Safety (KoFONS) using the financial resource granted by the Nuclear Safety and Security Commission (NSSC) of the Republic of Korea (no. RS-2024-00403364).

REFERENCES

[1] B.C. Author(s), “Title of the Journal Paper,” Journal Name in Italic, Volume in Bold (followed by the issue number in parentheses if known, see Ref. 5 below), pp. 34-89 (2007).
[2] D.E. Author(s), “Article Title,” *Proceedings of Meeting in Italic*, Location, Dates of Meeting, Vol. 31, pp. 134-156 (2012).
[3] F.G. Author, *Book Title in Italic*, Page or Chapter Numbers, Publisher, City and Country (1995).
[4] Raissi, Maziar, Paris Perdikaris, and George E. Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations." *Journal of Computational physics* 378 (2019): 686-707.
[5] Lu, Lu, et al. "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators." *Nature machine intelligence* 3.3 (2021): 218-229.

[6] Jeon, Joongoo, Juhyeong Lee, and Sung Joong Kim. "Finite volume method network for the acceleration of unsteady computational fluid dynamics: Non-reacting and reacting flows." *International Journal of Energy Research* 46.8 (2022): 10770-10795.

[7] Kontolati, Katiana, et al. "Learning nonlinear operators in latent spaces for real-time predictions of complex dynamics in physical systems." *Nature Communications* 15.1 (2024): 5101.

[8] Bank, Dor, Noam Koenigstein, and Raja Giryes. "Autoencoders." *Machine learning for data science handbook: data mining and knowledge discovery handbook* (2023): 353-374.

[9] Cybenko, George. "Approximation by superpositions of a sigmoidal function." *Mathematics of control, signals and systems* 2.4 (1989): 303-314.

[10] Rabeh, Ali, Adarsh Krishnamurthy, and Baskar Ganapathysubramanian. "3D Neural Operator-Based Flow Surrogates around 3D geometries: Signed Distance Functions and Derivative Constraints." arXiv preprint arXiv:2503.17289 (2025).

[11] He, Junyan, et al. "Geom-deeponet: A point-cloud-based deep operator network for field predictions on 3d parameterized geometries." Computer Methods in Applied Mechanics and Engineering 429 (2024): 117130.