Accelerating Computational Fluid Dynamics Simulations for Small Modular Reactor Digital Twin using Surrogate Model

Minseo Lee [a*], Chaehyeon Song [b], Bumjin Jo [b], Shilaj Baral [a], Sangam Khanal [c], Seongmin Oh [d], Minseop Song [b], Joongoo Jeon [a*]
[a]*Division of Advanced Nuclear Engineering, Pohang University of Science and Technology, Pohang-si, Gyeongsangbuk-do, Korea*
[b]*Department of Nuclear Engineering, Hanyang University, Seoul, Korea*
[c]*Graduate School of Integrated Energy-AI, Jeonbuk National University, Jeonju-si, Jeollabuk-do, Korea*
[d]*Department of Quantum System Engineering, Jeonbuk National University, Jeonju-si, Jeollabuk-do, Korea*
[*]*Corresponding author:jgjeon41@postech.ac.kr*

Keywords : small modular reactor, computational fluid dynamics, AI, deeponet, digital twin, deep learning

1. Introduction

Recent progress in artificial intelligence, together with the rapid expansion of data-center infrastructure, is causing electricity consumption to rise markedly. The widespread deployment of large language models (LLMs) and cloud services further underscores the need for power sources that are both dependable and continuous. Addressing this growing demand calls for energy systems that are stable and sustainable. In this context, nuclear power is often viewed as a strong candidate. Among available nuclear options, small modular reactors (SMRs) have emerged as a leading next-generation concept, driven by improved safety characteristics and the practicality of modular deployment.

Digital twins (DT) aimed at safe and efficient SMR operation maintain a close connection with physical assets. By synchronizing on-site physical behavior digitally, these systems enable the precise monitoring of internal reactor states and the optimization of control systems. More broadly, a SMR-oriented DT provides a comprehensive environment for state estimation, prediction, and decision support across a wide range of operating conditions. The U.S. Nuclear Regulatory Commission (NRC) and Department of Energy (DOE) recognize Digital Twin (DT) technology as a crucial enabler for comprehensively enhancing lifecycle performance, safety, and proactive decision-making while mitigating operational errors [1]. Reflecting this direction, Korea has promoted the virtual-SMR(VSMR) platform project within the Global Top Strategic Research Working Group. The VSMR platform aims to advance next-generation reactor technologies through high-fidelity simulation, with a central target of achieving real-time prediction of thermal–hydraulic behavior at computational fluid dynamics (CFD) level accuracy.

CFD typically solves flow fields by numerically solving the Navier–Stokes equations using discretization schemes such as the finite difference method (FDM), finite volume method (FVM), and finite element method (FEM). FVM is particularly prevalent in CFD because of its natural treatment of conservation laws. Nevertheless, practical CFD analyses require geometry preparation and mesh generation, and computational cost escalates rapidly as configurations become larger and more complex. In addition, when initial or boundary conditions are modified, conventional workflows often demand that simulations be re-executed from the start.

To reduce these costs, a growing body of work has investigated machine-learning-based surrogates for accelerating CFD. Neural networks (NNs) employed with non-linear activations, most commonly multilayer perceptron (MLP), convolutional neural network (CNN), and long short-term memory network (LSTM), have shown promise for forecasting unsteady flows. For example, Srinivasan et al. reported data-driven prediction of turbulent shear flows using MLP and LSTM, demonstrating that LSTM can reproduce key turbulence statistics and dynamics [2]. Sangam et al. conducted controlled benchmarks of CNN architectures on a natural-convection CFD dataset to evaluate unsteady prediction performance in small-data settings [3]. Despite these advances, many purely data-driven NN models remain sensitive to the boundary conditions seen during training and can degrade noticeably under out-of-distribution scenarios.

Deep operator network (DeepONet) has been proposed to alleviate this limitation and have achieved strong results for a broad range of non-linear partial differential equations, including the Navier–Stokes equations [4]. Rather than approximating a pointwise input–output function as in standard NNs, DeepONet learn an operator that maps input functions to output functions. This perspective enables initial and boundary conditions to be incorporated in functional form during training, improving predictive robustness across varying conditions. Consequently, once trained, DeepONets can drive fast inference of flow fields without retraining, supporting real-time prediction.

In this work, we build a CFD dataset for an SMR helical coil steam generator (HCSG) to develop a surrogate model depending on mesh type. To further evaluate practicality for complex three-dimensional configurations, we construct datasets on unstructured mesh and structured mesh and compare training behavior

and prediction performance across the surrogate model. And following this, we train a suitable model and assessed its performance as a surrogate model for 3D geometry in SMR.

2. Methods

In this section, the latent DeepONet framework and its reduce-order model are described, including MLP and CNN-based autoencoders for dimensionality reduction and a multi-scale trunk network designed to capture high-frequency periodic flow dynamics.

2.1 Latent DeepONet model for surrogate model

DeepONet is an operator learning framework designed to deliver accurate predictions under varying initial conditions. Unlike conventional neural networks that learn pointwise input and output mappings, DeepONet learns a function to function mapping by leveraging the universal approximation theorem for nonlinear operators [4]. In this setting, an operator maps an input function to an output function, enabling generalization across various initial conditions rather than a single fixed scenario.

Building on this idea, the DeepONet architecture comprises two components: a branch network and a trunk network. The branch network encodes the input function u into a finite-dimensional representation. Because the input is continuous, it is first sampled at a set of sensor locations, and the sampled values are provided to the branch network to learn coefficient features b_k. The trunk network takes spatiotemporal coordinates y as input and learns the basis functions t_k. The final DeepONet prediction $G(u)(y)$ is obtained by combining the branch and trunk outputs over p features, as summarized in Eq. (1).

$$G(u)(y) \approx \sum_{k=1}^{p} b_k t_k \tag{1}$$

This mechanism is conceptually related to the approximation principle used in the finite element method (FEM), where a solution $U(x)$ is expressed as a linear combination of coefficients c_i and basis functions $\phi_i(x)$ over N terms, as shown in Eq. (2).

$$U(x) = \sum_{i=1}^{N} c_i \phi_i(x) \tag{2}$$

From this perspective, DeepONet can be viewed as learning the coefficients via the branch network and the basis functions via the trunk network, forming an efficient representation for PDE solution operators. Across various physical systems, DeepONet has been reported to achieve lower generalization error than standard neural networks while maintaining high accuracy and rapid convergence [4].

Within the VSMR project, a key objective is CFD fidelity flow prediction. However, CFD outputs for realistic reactor geometries are inherently high-dimensional due to fine spatial resolution and multiple physical variables, even when the number of training samples is limited. As a result, training a vanilla DeepONet directly on such data is often restricted by GPU memory and computational cost. To address this challenge, this study adopts latent DeepONet (L-DeepONet), which integrates a reduced-order model with DeepONet. L-DeepONet first compresses high-dimensional flow fields into a low-dimensional space with reduce-order model to train DeepONet in the latent space. And then reconstructs the predicted latent outputs back to the original physical space. Prior work suggests that learning in a compact latent representation can improve both training efficiency and predictive accuracy compared to standard DeepONet [5]. The L-DeepONet architecture is illustrated below Fig. 1.

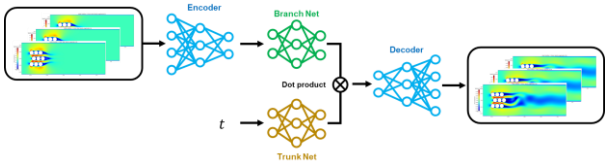

Fig. 1. L-DeepONet architecture. Taking initial conditions as input, the model predicts time-dependent variable fields as output.

For the reduced-order model, classical linear techniques such as proper orthogonal decomposition and principal component analysis are widely used in engineering. Here, we instead employ an autoencoder (AE) to better capture nonlinear structure in flow fields via nonlinear activations [6]. An AE consists of an encoder that compresses the input into a latent code and a decoder that reconstructs the original field; it is trained by minimizing reconstruction error. In this work, the AE architecture is tailored to the mesh type: an MLP-based AE is used for unstructured-mesh CFD data, while a CNN-based AE (CAE) is adopted for structured-grid data. This design allows systematic comparison of predictive performance and model complexity across dimensionality-reduction choices.

Training L-DeepONet requires a pretrained AE. We first train the AE on the CFD dataset, then use the encoder to compress the initial conditions into latent vectors provided to the branch network during training and inference. DeepONet predicts in the latent space, and the pretrained decoder subsequently maps these latent predictions back to the original flow fields, yielding outputs in the original physical space.

2.2 Multi-scale DeepONet for Spectral bias

Transient flows dominated by Kármán vortex shedding contain strong high-frequency content in both time and space, which can expose the spectral bias of neural-network surrogate models, including DeepONet, that typically fit low-frequency components earlier in training. From a Fourier-analysis viewpoint, Rahaman et al. showed that deep networks learn low-frequency functions faster, yielding a frequency-dependent learning-rate bias that can limit accuracy for oscillatory

dynamics [7]. To alleviate this issue, we adopt a multi-scale DeepONet strategy by embedding a multi-scale representation in the trunk network. Wang et al. reported that spectral bias in DeepONet degrades operator learning when mapping between highly oscillatory functions, and proposed multi-scale DeepONet in which multiple trunk sub-networks can be interpreted as learning distinct basis functions, thereby covering different frequency bands [8]. They demonstrated substantial improvements over the vanilla DeepONet with a comparable parameter budget on high-frequency wave-scattering benchmarks, and further showed that applying the multi-scale design to the trunk network is particularly effective for learning operators with high-frequency outputs [8]. In this work, because the target transient flow is governed by time-periodic vortex shedding, we implement the multi-scale trunk by feeding parallel sub-trunk networks with differently scaled time coordinates. By processing scaled time coordinates in parallel, the trunk network learns a richer set of basis functions across a broader frequency range. This approach is expected to enhance reconstruction accuracy and predictive robustness for periodic structures in transient forecasting. The multi-scale L-DeepONet architecture is illustrated in Fig. 2.

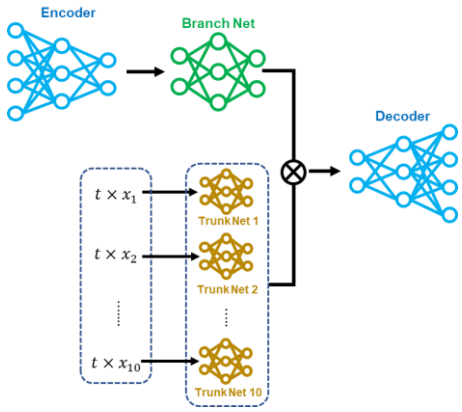

Fig. 2. Multi-scale L-DeepONet architecture. The time coordinate t is scaled and fed into sub-trunk networks, enabling the trunk to learn multi-scale basis representations over a wide frequency range for periodic transient flow forecasting.

2.3 Unstructured HCSG dataset

To train the surrogate models, we generated a two-dimensional transient CFD dataset of primary-side cross-flow in an SMR HCSG, with design parameters derived from KAERI's SMART design. The geometry and mesh were created in ANSYS SpaceClaim 24.2.0, and unsteady simulations were performed in ANSYS Fluent 24.2.0. The tube bundle was modeled as a staggered rod array with offset rows to capture wake deflection and vortex interaction effects. The computational domain was defined relative to a reference cylinder located in the first row and first column, and the downstream length was set to three times the upstream length measured from the last-column cylinder. The overall geometry is illustrated in Fig. 3, with key dimensions summarized in Table I: a radial length of 252 mm, axial length of 94 mm, tube diameter of 12 mm, and pitches of 13.5 mm and 18.29 mm.

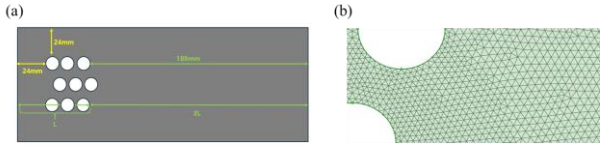

Fig. 3. (a) Domain of geometry. (b) Unstructured mesh.

For all cases, the outlet boundary condition was set to a gauge pressure of 0 Pa and no-slip conditions were applied on all solid walls. Water was used as the working fluid, and the SST k-ω turbulence model was employed to resolve turbulent boundary-layer behavior. The dataset captures unsteady wake dynamics, including persistent Kármán vortex shedding downstream of the tube bundle across the simulated inlet-velocity range; a summary of the dataset and simulation conditions is provided in Table II below. To span operating conditions, 50 inlet velocities from 0.10 m/s to 0.60 m/s were simulated excluding 0.40 m/s, of which 45 cases were used for training and five cases (0.10, 0.20, 0.30, 0.50, and 0.60 m/s) were used for validation, while additional test cases at 0.40 m/s and 0.70 m/s were used to assess generalization.

2.3 Structured HCSG dataset

In addition to the unstructured-mesh dataset, we prepared a two-dimensional structured dataset to support training of a CAE, motivated by future extension to three-dimensional SMR geometries where CNN-based models are often preferred for parameter scalability and computational efficiency. To maintain physical consistency, the structured data were produced using the same solver settings and inlet conditions as the unstructured simulations. Because ANSYS plane-generation and ASCII export utilities are available only for 3D meshes, the original 2D domain was extruded by 0.1 mm in the z-direction to form a thin 3D mesh, from which flow variables were sampled on a regularly spaced plane and exported to construct structured fields consistent with the unstructured mesh physics.

Table I: Parameters for HCSG Geometry

Parameter	**Geometry2**
Radial Length	252 mm
Axial Length	94 mm
Tube Diameter	12 mm
Pitch	13.5 mm / 18.29 mm
Upstream Length	24 mm
Downstream Length	189 mm

Table II: Fluent Solver Settings

Meshing Tool	**Ansys SpaceClaim**

CFD Tool	Ansys FLUENT
Turbulence model	SST k- ω
Tube wall condition	No-slip condition
Heat transfer	No consider
Solver	Pressure-based
Fluid Material	Water
Density	998.2(Const.) [kg/m^3]
Viscosity	0.001 (Const.) [$Pa \cdot s$]
# of Time Steps	10,000
Time step size	0.01s
Export timesteps	0.01s

2.4 Entrance dataset in SMART

In this study, to evaluate the performance of the surrogate model validated on 2D data, a high-resolution steady-state CFD datasets were constructed for the SMART entrance geometry, and a preprocessing procedure was performed to convert it into a format optimized for AI model training. This entrance geometry represents the inlet region where the primary coolant flows into the HCSG cassettes, and a 1/4 symmetric domain was selected for analysis considering computational resources.

In the data generation phase, a total of 41 CFD datasets were acquired by varying the inlet mass flow rate in 2% increments based on the reactor's normal operating range of 20% to 100%. The 100% power condition meant high-speed operation of four reactor coolant pumps, with the inlet mass flow rate set to 396.25 kg/s. To verify the surrogate model, an evaluation dataset outside the training range was established, specifically including the 105% power condition to evaluate the extrapolation performance of the model.

The CFD datasets were processed through the interpolation process from an unstructured mesh to structured mesh for application to a CAE model. This step is essential for maximizing computational efficiency when training on 3D CFD data.

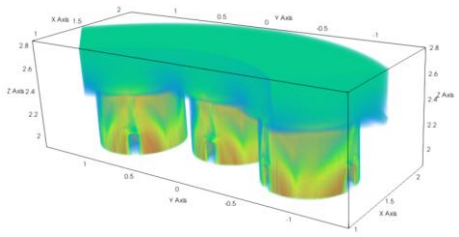

Fig. 4. Interpolated entrance geometry in 100% power condition.

A rectangular bounding box completely encompassing the 3D entrance geometry was defined like Fig. 4, followed by interpolation using the inverse distance weighting method. During this process, a zero-padding method was applied to treat nodes without flow information as 0, and a masking technique based on the distance between nodes was introduced to minimize distortion of the original entrance geometry during interpolation. Through this preprocessing, the existing unstructured mesh data consisting of 5,176,557 nodes was restructured into high-resolution structured mesh data with 12,828,672 nodes, establishing a stable foundation for AI model training.

3. Results

3.1 Results in unstructured CFD dataset

Pressure drop was evaluated by extracting nodal data from the inlet and outlet regions. Figures 5 and 6 illustrate the performance of L-DeepONet trained on the unstructured CFD dataset, where the model accurately reconstructed the pressure field magnitude and the periodic oscillations in the wake for both interpolation and extrapolation scenarios. The time-series analysis in Fig. 6 further highlights these periodic variations, which are effectively captured by the multi-scale trunk network. The model achieved high predictive fidelity, with mean relative errors of 0.91% in Fig. 7 (a) and 2.17% in Fig. 7 (b).

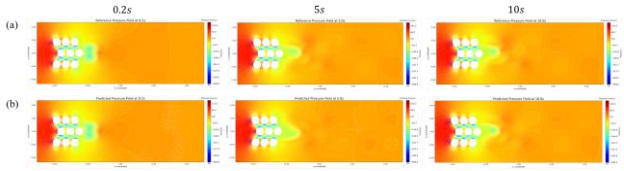

Fig. 5. Unstructured L-DeepONet predicts the pressure field in interpolation.

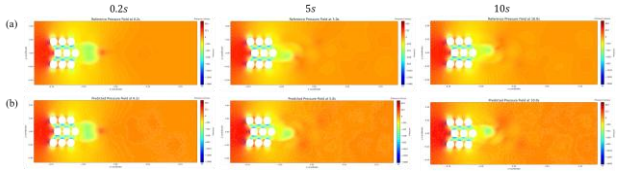

Fig. 6. Unstructured L-DeepONet predicts the pressure field in extrapolation.

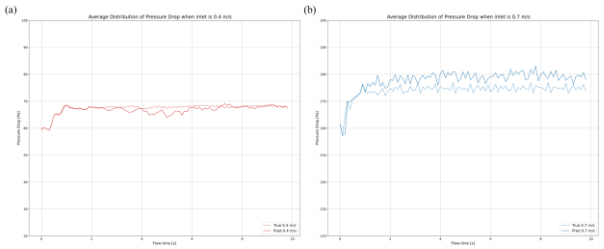

Fig. 7. Comparison with the pressure drop predicted by unstructured L-DeepONet

3.2 Results in structured CFD dataset

Figures 8 and 9 present the L-DeepONet results for the structured-grid dataset, showing consistent accuracy in pressure-field reconstruction. Figure 10 illustrates the pressure drop time-series, where the model demonstrates high fidelity, particularly under interpolation conditions, with mean relative errors of 0.33% and 2.27% for Figures 10 (a) and (b). These findings confirm that both unstructured and structured L-DeepONet variants are

robust surrogate models for SMR applications. Furthermore, the CAE-based approach offers a substantial efficiency advantage; while the MLP-based AE requires 1,649,957,225 parameters (6.15 GB), the CAE uses only 54,183,813 parameters (206.69 MB). This 30-fold reduction in model size highlights the CAE's superior scalability for future transitions to 3D geometries.

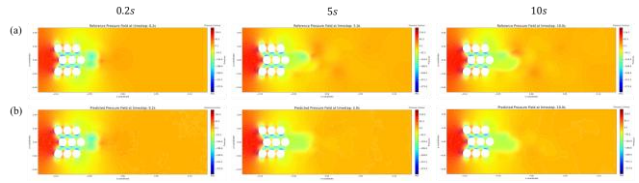

Fig. 8. Structured L-DeepONet predicts the pressure field in interpolation.

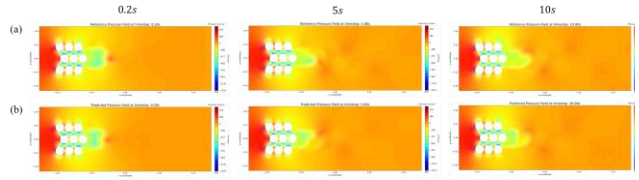

Fig. 9. Structured L-DeepONet predicts the pressure field in extrapolation.

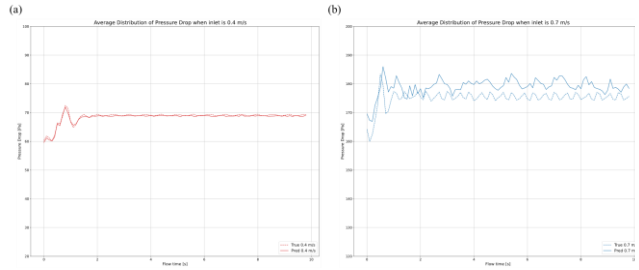

Fig. 10. Comparison with the pressure drop predicted by structured L-DeepONet.

3.3 Results in entrance geometry in SMART

To validate the model's performance for 3D DT applications, we applied the CNN-based L-DeepONet to the 3D SMART entrance geometry.

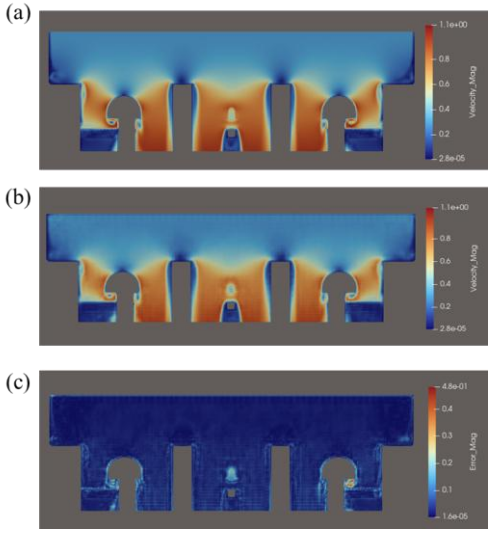

Fig. 11. Extrapolation performance in 105% power condition with CNN-based L-DeepONet. (a) reference. (b) prediction. (c) error between reference and prediction.

In Fig. 11, the surrogate model predicted the velocity magnitude of the steady-state flow field with reasonable accuracy, achieving a relative L_2 error of approximately 0.102759. Comparison between the CFD reference and the prediction shows that the model effectively captured the primary flow characteristics and spatial distribution within the entrance geometry, demonstrating its reliability in approximating complex 3D simulations.

4. Conclusions

This research developed DeepONet-based surrogate models for the digital twin realization of SMR thermal-hydraulic flows in helical coil steam generators. Two types of CFD datasets, unstructured and structured, were created to support MLP-based and CNN-based AEs. The L-DeepONet framework was implemented using a multi-scale representation in the trunk network to effectively resolve spectral bias and periodic flow features like Kármán vortex shedding. The models showed strong predictive performance for pressure fields and fluctuations during interpolation and extrapolation tests at 0.4 m/s and 0.7 m/s. While conventional CFD in 2D required 30 minutes per case, the L-DeepONet model achieved a 450-fold speedup by providing results in 4 seconds. Furthermore, the model effectively predicted complex 3D entrance geometty, capturing the distribution of the flow fields. While conventional CFD simulations in 3D required approximately 4 hours per case, the developed surrogate model provided predictions in just 4.3 seconds. Consequently, a 3,300-fold speedup in computational performance was achieved, highlighting the model's feasibility for real-time 3D digital twin implementations.

ACKNOWLEDGMENT

This work was supported by the National Research Council of Science & Technology (NST) grant by the Korea government (MIST) (No. GTL24031-000) and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MIST) (RS-2025-02634798).

REFERENCES

[1] Yadav, Vaibhav, et al. Regulatory Considerations for Nuclear Energy Applications of Digital Twin Technologies. No. INL/RPT--22-67630; TLR-RES/DE/REB-2022-06. Idaho National Laboratory (INL), Idaho Falls, ID (United States), 2022.

[2] Srinivasan, Prem A., et al. "Predictions of turbulent shear flows using deep neural networks." Physical Review Fluids 4.5 (2019): 054603.

[3] Khanal, Sangam, Shilaj Baral, and Joongoo Jeon. "Comparison of CNN-based deep learning architectures for unsteady CFD acceleration on small datasets." Nuclear Engineering and Technology 57.10 (2025): 103703.

[4] 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.
[5] 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.
[6] Bank, Dor, Noam Koenigstein, and Raja Giryes. "Autoencoders." Machine learning for data science handbook: data mining and knowledge discovery handbook (2023): 353-374.
[7] Rahaman, Nasim, et al. "On the spectral bias of neural networks." International conference on machine learning. PMLR, 2019.
[8] Wang, Bo, Lizuo Liu, and Wei Cai. "Multi-scale deeponet (mscale-deeponet) for mitigating spectral bias in learning high frequency operators of oscillatory functions." arXiv preprint arXiv:2504.10932 (2025).