Bei diesem Beitrag handelt es sich um einen wissenschaftlich
begutachteten und freigegebenen Fachaufsatz („reviewed paper“).

doi.org/10.37544/1436-4980-2026-03-15
Datum der Einreichung: 25.11.2025
Datum der Annahme: 17.12.2025
Datum der Veröffentlichung: 07. 04.2026

A review of operator neural networks for industrial processes

Neural operators in industrial processes

L. Wang, J. Schilp

ABSTRACT Operator neural networks open up new possibilities for the efficient modeling and simulation of complex industrial processes. This article provides an overview of common model types such as DeepONet, FNO, GNO, and related variants, and demonstrates through various industrial application examples how suitable architectures can be selected.

KEYWORDS
Neural Operator Networks; Industrial Processes

Neural Operator in industriellen Prozessen

ZUSAMMENFASSUNG Operator-Neuronale-Netze eröffnen neue Möglichkeiten für die effiziente Modellierung und Simulation komplexer Industrieprozesse. Dieser Beitrag gibt einen Überblick über die gängigen Modelltypen wie DeepONet, FNO, GNO und weitere Varianten und zeigt anhand verschiedener industrieller Anwendungsbeispiele, wie sich geeignete Netzwerke auswählen lassen.

1 Introduction

In recent years, deep learning has been increasingly applied across various fields. Its strong capabilities and practical value have generated new research interest in multiple areas. In industrial processes, fundamental mathematical models describing physical phenomena—such as fluid dynamics, heat transfer, and material mechanics—often involve solving complex partial differential equations (PDEs). For many complex physical problems, analytical solutions are not even available, making it necessary to resort to numerical methods. This typically leads to high computational costs and time consumption [1–3]. Moreover, in many industrial processes, the governing PDEs of a system are unknown or incomplete. This inherent incompleteness compels modelers to rely on empirical assumptions or inaccurate simplifications, which leads to highly inaccurate results.

As a powerful deep learning model, operator neural networks can achieve high-precision prediction and classification by learning from large datasets, and have been widely applied across various fields [4–6]. Beyond learning from data, they can also directly learn mathematical models [7, 8], demonstrating significant potential for accelerating computations involving complex physical phenomena in industrial applications [9, 10].

In industrial applications, the problems encountered are highly diverse in form. Challenges across different industrial sectors often vary in terms of many factors such as grid structure, complexity of physical fields, and degree of nonlinearity. As a result, a single type of neural operator network can hardly be applied universally to all such cases. There is a need to choose and design network architectures that align with the structural characteristics of specific problems. This paper systematically reviews the main types and characteristics of neural operator networks, summarizes their features and data properties, and explores how to configure suitable neural operator methods for problems across different conditions.

The selected literature primarily originates from academic databases such as Google Scholar, arXiv, IEEE Xplore, and SpringerLink, with search keywords including “neural operator,” “DeepONet,” “Fourier Neural Operator,” “graph neural operator,” and “physics-informed neural operator.” During the literature screening process, priority was given to neural operator methods that have been validated in industrial or practical physical applications. Purely data-driven surrogate models (lacking operator generalization capability) and traditional physics-informed neural networks (which solve only single equations and do not establish operator mappings) were excluded from the core scope of this review. The goal is to assist readers in enhancing the applicability and predictive performance of such models when addressing industrial problems.

2 Neural operator research

Neural networks can learn the mathematical operators of physical systems, unlike traditional numerical methods which need to be completely recomputed whenever initial or boundary conditions change. This capability enables the development of general-purpose PDE solvers. An operator is a mapping from one function space to another. To learn such operators with neural networks, two central steps are involved: encoding and approximation.

Encoding refers to the process of representing a function from a function space in a finite-dimensional vector form. This is achieved by evaluating the function at a set of discrete points or applying transformations such as the Fourier transform. The goal of encoding is to convert an infinite-dimensional function into a

Table 1 Neural operator methods.

Category		Method
Neural Operators	data-driven	DeepONet
		FNO
		GNO
		Deeponet grid uq
		B-DeepONet
		U-FNO
		GINO
	physics-informed	Pi-deeponet
		PINO
		GS-PI-DeepONet
		PINO-PC

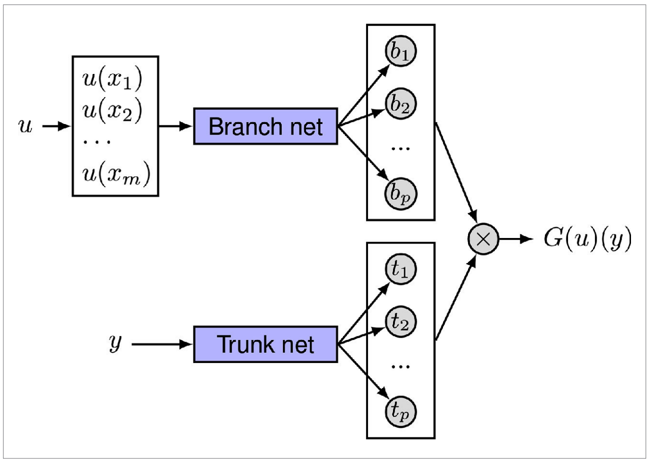

Fig. 1 DeepONet model. *Source: [11]*

finite-dimensional representation, which can then be used as input or output for a neural network.

Approximation refers to the process of using neural networks to learn the mapping between finite-dimensional vectors. This mapping represents an encoded approximation of the target operator. By using the function approximation capability of neural networks, this approach aims to capture the essential characteristics and underlying patterns of the operator.

Through the processes of encoding and approximation, a neural network can be trained to learn arbitrary operators. This enables a single model to solve a problem under varying initial or boundary conditions. Based on the literature review, the relevant methods can be summarized into three core architectures and their extensions, including: DeepONet (Deep Operator Network), Graph Neural Operators (GNO), and Fourier Neural Operators (FNO). **Table 1** summarizes selected neural operator methods. Data-driven and physics-informed approaches will be further detailed in subsequent sections of this chapter.

2.1 Data-driven neural operator methods

Lu et al. [3] proposed the DeepONet for learning operators. It consists of two components: a branch net and a trunk net. The branch net encodes the input function into a finite-dimensional vector, while the trunk net encodes the output function and multiplies it with the vector from the branch net to produce a vector approximating the output function of the operator, as illustrated in **figure 1**.

In response to different application requirements and model performance goals, researchers have developed various variants based on DeepONet architecture. *Moya et al.* [11] introduced a novel data-driven approach to reliably predict the post-fault dynamic trajectories of power grids. The core concept involves first constructing an operator based on DeepONet to simulate the relationship between pre-fault and post-fault trajectories. Subsequently, two methods are developed to quantify the uncertainty in the post-fault trajectory predictions. The first is a Bayesian DeepONet (B-DeepONet), which employs stochastic gradient Hamiltonian Monte Carlo to sample from the posterior distribution of the DeepONet parameters, thereby quantifying the uncertainty. The second is a probabilistic DeepONet (Prob-DeepONet), utilizing a probabilistic training strategy to quantify uncertainty with minimal additional computational cost. The model ultimately achieved high accuracy in simulations based on the New York-New England power grid model.

Also, to address the issue of noisy training data, *Lin et al.* [12] proposed an enhanced B-DeepONet to approximate the solution operator of parametric PDEs with noisy data. Unlike the Bayesian DeepONet by *Moya et al.* [11], which uses a Bayesian optimization-based adaptive sampling strategy, *Lin et al.* [12] employed an accelerated replica-exchange stochastic gradient Langevin dynamics (reSGLD) algorithm. This method trains two distinct DeepONet particles to more effectively handle noisy data and escape local minima.

Li et al. [13] proposed the FNO to learn mappings between function spaces for efficiently solving PDEs. Its core idea involves parameterizing the integral kernel in Fourier space and leveraging the Fast Fourier Transform (FFT) to enable efficient convolution operations. The FNO can be applied to solve a class of parametric PDE problems, including hyperbolic, elliptic, and parabolic types. The trained model can achieve prediction speeds several orders of magnitude faster than traditional PDE solvers.

Building upon the FNO framework, many researchers have conducted further studies [13, 14]. Among them, *Wen et al.* [15] proposed an enhanced Fourier neural operator-based deep learning model for multiphase flow, termed U-FNO. This model incorporates a mini U-Net path alongside the standard FNO to enhance its capacity for representing high-frequency information. The mini U-Net path is a convolutional neural network (CNN)-based method that performs downsampling and upsampling operations in the spatial domain, thereby extracting and reconstructing high-frequency details. Compared to both the standard FNO and pure CNN models, U-FNO achieves higher accuracy.

The Graph Neural Operator (GNO), also referred to as the Graph Kernel Network (GKN), was introduced by *Li et al.* [16] as a framework in operator learning, specifically designed for handling complex geometric problems. This approach achieves discretization-invariant approximation of nonlinear operators, meaning the trained model can generalize unseen resolutions and topological structures. By discretizing the solution domain of a PDE into a graph, GNO employs a message-passing mechanism over the graph to approximate continuous integral kernels. This allows it to naturally handle unstructured meshes and complex industrial geometries, addressing the limitation of FNOs in dealing with intricate domains. Experimental results demonstrate that for problems such as parametric Darcy flow, GNO outperforms traditional finite element methods (FEM) and deep learning methods reliant on fixed grids, achieving high accuracy and significant improvements in inference speed.

In summary, the advantage of DeepONet lies in its ability to handle multiple input and output functions simultaneously without retraining the network. However, it requires substantial training data to ensure generalization capability and often necessitates deeper or wider network architectures for highly nonlinear or complex PDEs. In contrast, FNO leverages the properties of the Fourier transform to reduce computational complexity and memory consumption, making it effective for handling high-dimensional or periodic PDEs. Its limitations include the need for sampling and interpolating both input and output functions, and it typically requires more frequency components to maintain accuracy when dealing with non-periodic PDEs. GNO has good performance at adaptively processing irregular or dynamically changing mesh structures, but they require pre-defined mesh partitioning and topology construction for inputs and outputs. Consequently, DeepONet is a suitable choice for problems with abundant data, FNO demonstrates significant advantages for high-dimensional or periodic PDEs, and GNO is more appropriate for geometric structure dynamically evolving problems.

2.2 Physics-informed neural operator methods

The data-driven neural operator approaches often face challenges related to numerical stability and accuracy. Due to the black-box nature of neural networks, they may fail to accurately capture the mathematical properties of PDE solutions and can suffer from overfitting when trained with limited data. Consequently, these methods typically require large volumes of labeled data to prevent a decline in predictive performance on unseen data. However, obtaining sufficient high-quality training data for complex PDE problems is often exceedingly difficult. Besides, these methods necessitate data sampling across different parameters and boundary conditions, which further compounds the challenges of data acquisition.

In response, some researchers have introduced physical constraints by formulating physics-informed residuals for a class of PDEs sharing similar mathematical characteristics. This enables operator learning with minimal or even non-labelled data. *Koric et al.* [17] employed the finite element method to discretize the spatial domain and treated the neural network's output as coefficients of the finite element basis functions. By incorporating physics constraints—namely, the heat conduction equation and boundary conditions—to guide the neural network's training, they enhanced the operator's generalization capability and physical interpretability, achieving high-precision solutions for the heat conduction equation under parametric heat sources. This method demonstrates high data efficiency, requiring only a small amount of training data to achieve favorable results.

Wang et al. [2] proposed a physics-informed deep neural operator for solving parametrized evolution equations with stochastic initial conditions. Addressing the limitation of existing DeepONet models in achieving long-term stable predictions, they segmented the entire time domain using an iterative algorithm. The prediction from each previous time step served as the initial condition for the next, thereby enabling global predictions over extended time intervals.

Li et al. [18] introduced the Physics-Informed Neural Operator (PINO), a framework to systematically integrate neural operator architectures like FNO with physics-informed learning hrough PDE-based regularization. This hybrid approach achieves high data efficiency, enabling few-shot and even zero-shot operator learning—while maintaining physical consistency in predictions. The method has demonstrated good performance in solving complex industrial-scale PDEs, achieving accelerated inference speeds without compromising accuracy compared to conventional numerical solvers.

In summary, physics-informed neural operator methods integrate the powerful mapping capabilities of operators with strict physical laws, effectively balancing data requirements and stability concerns. By enhancing both data efficiency and physical plausibility, these approaches create new possibilities for developing high-fidelity, real-time predictive models in industrial processes.

3 Industrial applications of neural operators

Industrial applications present a diverse spectrum of challenges with multi-faceted requirements. The decision criteria discussed in this section are organized along methodological dimensions rather than application domains. This is because industrial problems rarely fall into a single category; instead, different industrial sectors typically exhibit distinct dominant combinations of these criteria. Consequently, the following decision logic applies across various industrial domains and is illustrated using representative examples from different industries. To assist readers in efficiently identifying the most suitable neural operator architecture for their specific scenarios, this chapter systematically organizes the model selection pathway along three critical deci-

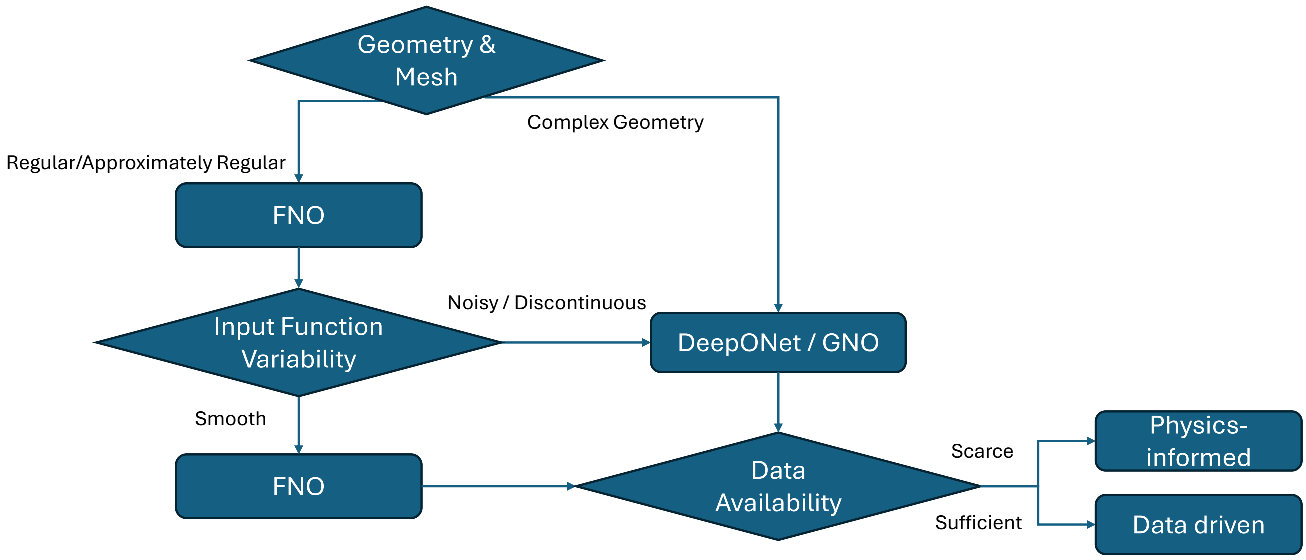

Fig. 2 Model selection pathway for neural operators. *Source: Universität Augsburg*

sion dimensions, supported by concrete research findings and representative application cases, see **figure 2**.

3.1 Geometry and mesh

When selecting neural operator models, the geometric domain and mesh characteristics serve as the primary decision criterion. This is largely determined by the limited adaptability of FNO to complex geometric regions. In contrast, for problems defined on regular geometric domains—such as rectangular or cubic areas with uniform grids—FNO can fully leverage the computational advantages of the Fast Fourier Transform (FFT), outperforming both DeepONet and Graph Neural Operators (GNO) in terms of computational speed and memory efficiency. Therefore, FNO is generally the model of choice under such conditions [13, 19].

Such application scenarios are commonly found in industrial contexts including fundamental fluid dynamics and heat transfer analysis, structural mechanics with regular geometries, thermal management of electronic devices, and basic electromagnetic field computations. Typical examples include fully developed flow in pipes, stress distribution in standardized structural components, steady-state temperature fields in chip packaging, and electromagnetic field analysis in waveguides. These problems share well-defined geometric boundaries and relatively smooth physical field variations. It is worth noting that through appropriate geometric simplification and boundary condition idealization, many complex industrial problems can be transformed into such regular configurations, thereby leveraging FNO's computational advantages for rapid preliminary analysis or parametric studies.

Conversely, when the problem involves complex geometries (e.g., engine blades, porous media) or requires computation on unstructured grids, the standard FNO is no longer suitable [19]. In such cases, more geometrically adaptable models such as DeepONet or GNO should be considered. Although variants like Geo-FNO [19] have been developed to handle certain degrees of geometric variation, models based on graph structures or general operator frameworks tend to offer more natural representation and greater flexibility when dealing with significant topological changes or highly irregular domains.

3.2 Input function variability

After confirming the geometric compatibility of the model, the properties of the input function become the second crucial criterion in model selection.

If the input function exhibits high randomness, discontinuity, or significant noise—such as permeability fields in random porous media or rapidly varying source terms—DeepONet is generally the preferred solution. Its decoupled architecture based on branch and trunk networks provides stronger robustness for processing discontinuous or noisy data [19]. Multiple comparative studies confirm that although FNO achieves higher accuracy on ideally smooth data, its reliance on global Fourier transforms tends to amplify high-frequency noise and discontinuities in the frequency domain, leading to significant degradation in solution accuracy [3, 19]. In contrast, DeepONet demonstrates lower sensitivity to input perturbations and can stably learn operator mappings from irregular data [19].

The GNO occupies a unique intermediate position in this decision hierarchy. By employing local message passing on graph structures to approximate integral kernels [16], GNO's localized operations avoid the excessive amplification of noise inherent in global transformations. Consequently, when handling input functions defined on unstructured or irregularly sampled points, GNO typically exhibits superior robustness compared to standard FNO. This makes it a strong contender in scenarios with complex geometries and input functions containing moderate noise or irregularity, effectively balancing geometric versatility and data robustness.

Conversely, if variations in the input function primarily manifest as smooth evolution of initial conditions, and the physical field to be solved possesses good regularity, FNO can leverage its global convolution operator advantages. When addressing smooth problems, FNO delivers exceptional performance in both accuracy and speed [13]. However, it is crucial to note that FNO exhibits high sensitivity to noise and is therefore only suitable for smooth problems; otherwise, the solution error increases substantially.

Kushwaha et al. [19, 20] employed GRU-enhanced and ResU-Net-based DeepONets to successfully learn thermo-mechanical operators in continuous casting and directed energy deposition processes. These models accurately reconstructed temperature and stress fields across diverse thermal histories, displacement boundary conditions, and topology-optimized geometries, maintaining prediction errors within a few percent while achieving speedups of 10^4–10^5 compared to conventional finite element simulations. The results demonstrate DeepONet's remarkable capability in handling highly variable input functions.

A representative industrial application of GNO is the real-time prediction of aerodynamic fields for 3D vehicle geometries. *Li et al.* [21] proposed the GINO framework, a hybrid Graph-Informed Neural Operator designed to address the challenges of high-Reynolds-number aerodynamic analysis in automotive and aerospace design. Conventional CFD (Computational Fluid Dynamics)-based evaluation of surface pressure distributions and drag coefficients for complex 3D shapes is computationally prohibitive, especially when geometries—represented as point clouds—exhibit significant variability. By combining GNOs to encode irregular geometric inputs with FNOs operating on an implicit regular grid, GINO effectively learns the shape-to-flow field operator and handles substantial input function variability in both geometry and inlet conditions. Trained on only 500 CFD simulations, the method accurately predicts surface pressure and drag, achieving a 26,000× speedup over optimized GPU-CFD solvers and reducing error on unseen shape-boundary-condition combinations by roughly 25 % compared to conventional deep learning approaches.

3.3 Data scarcity

After the model's geometric adaptability and input function characteristics are determined, data availability and the capacity to incorporate physical laws form the final decision-making basis for model selection. The core of this decision layer lies in determining whether physical constraints need to be introduced to compensate for insufficient data, based on the amount of high-fidelity simulation data available.

In data-sufficient scenarios, where a large amount of high-precision numerical simulation or experimental data is available as labels, purely data-driven models such as standard FNO or DeepONet can generally adequately learn the underlying operator mapping and achieve excellent predictive accuracy without explicitly incorporating physical laws [19].

However, in many practical industrial applications, obtaining high-fidelity data is extremely costly, often leading to data-scarce challenges. If the governing partial differential equations are known in such cases, introducing physical constraints becomes essential. The integration of the Physics-Informed Neural Networks (PINNs) framework with operator learning has given rise to models such as the Physics-Informed Neural Operator (PINO) [18] and Physics-Informed DeepONet (PI-DeepONet) [2]. By incorporating the PDE residual as a regularization term into the loss function, these models simultaneously fit sparse data and adhere to physical laws during training, thereby significantly enhancing generalization capability and prediction reliability under limited data conditions.

In industrial structural analysis and design optimization, the high cost of high-fidelity finite element analysis (FEA) and the complexity of geometric structures often result in severely limited training data. To address this challenge, *Zhang et al.* [22] developed GS-PI-DeepONet, which integrates graph neural networks (GNNs) into the DeepONet architecture to handle unstructured finite element meshes and incorporates physical residuals as soft constraints. This design enables the model to maintain physical consistency even under sparse data conditions. Experimental results demonstrate that the method achieves accuracy comparable to traditional FEM in displacement field prediction ($R^2 \approx 0.9999$), while providing 7–8× acceleration in practical engineering applications. Owing to its mesh resolution invariance and high data efficiency, the framework is particularly suitable for rapid parameter sweeping and structural design optimization.

In energy-intensive industries such as aerospace and maritime transportation, active drag reduction control in high-Reynolds-number turbulent flows represents a critical technology for energy conservation. However, traditional CFD-based model predictive control requires repeated solutions of the Navier-Stokes equations, leading to prohibitively long optimization iterations that cannot meet millisecond-level real-time control requirements. To overcome this limitation, *Zhao et al.* [23] developed PINO-Predictive Control (PINO-PC), which replaces the CFD solver in the MPC (Model Predictive Control) loop with a PINO as an ultra-fast and physics-consistent predictive model, reducing prediction time from minutes to milliseconds.

Leveraging PINO's explicit learning capability for Navier-Stokes operators, PINO-PC possesses extrapolation generalization ability absent in traditional model-free reinforcement learning (RL), maintaining stable control performance even under unseen high-Reynolds-number conditions. Experimental results demonstrate that in turbulent channel flow tests at Re = 15,000, PINO-PC achieved a 39 % drag reduction rate, exceeding existing RL and conventional control methods by over 32 %. This breakthrough enables real-time optimal active drag control in high-Re turbulent environments for the first time, establishing a transformative technical pathway for energy-efficient design in aviation and maritime applications.

3.4 Overview of neural operator methods

Table 2 systematically summarizes the key characteristics and applicable scenarios of the mentioned representative neural operator methods from an industrial application perspective.

Spectral methods, such as FNO, exhibit extremely high inference efficiency on regular grids; however, their ability to handle complex geometries and non-periodic boundary conditions remains limited. In contrast, integral kernel-based and graph-based operators offer greater geometric adaptability at the cost of higher training complexity.

Physics-informed enhanced variants, by incorporating physical constraints, significantly reduce dependence on training data volume and demonstrate stronger robustness under data-scarce conditions. This characteristic makes them particularly suitable for industrial application scenarios where high-fidelity numerical simulations or physical experiments are costly.

Table 2 Comparative Overview of Neural Operator Methods.

Method	Geometry/Mesh Adaptation	Data Requirement	Inference Cost	Robustness to Noise/ Discontinuity	Typical Industrial Applications
DeepONet	Mesh-free, Parametric geometry	Medium–High	Low	Medium	Parametric structures, Material models
FNO	Structured grid	Medium	Very low (FFT)	Low–Medium	Fluid flow, Heat transfer
GNO	Unstructured mesh	High	Medium	Medium	PDEs on complex geometries
GINO	Unstructured + Geometric encoding	High	Medium–High	Medium	Large-scale 3D PDEs
PINO	Structured grid	Low–Medium	Medium	High	Multiphysics, Data-scarce scenarios
PI-DeepONet	Parametric geometry	Low–Medium	Low	High	Industrial design space exploration

4 Summary

This paper systematically reviews the development of neural operator methodologies and their applicability in industrial processes. Three key decision dimensions have been identified: geometric and mesh compatibility, input function characteristics, and data scarcity, providing clear guidance for model selection.

Among data-driven approaches, DeepONet, FNO, and GNO each possess distinct advantages: DeepONet demonstrates strong robustness against irregular inputs; FNO achieves exceptional efficiency for smooth problems on regular domains; while GNO specialize in handling complex geometries and unstructured meshes. However, their dependence on high-fidelity data constrains industrial deployment.

To overcome data limitations, physics-informed neural operators such as PINO and PI-DeepONet incorporate governing equation residuals, maintaining physical consistency and generalization capability even with scarce data. Industrial cases demonstrate that these hybrid methods achieve orders-of-magnitude acceleration while preserving high accuracy, establishing new paradigms for real-time simulation and optimal control.

Through this paper, readers can clearly follow the selection pathway provided by the authors when choosing an operator neural network for a specific industrial process problem, allowing them to make informed choices that balance efficiency, applicability, and accuracy.

Future research should focus on establishing more refined quantitative criteria to support the analysis of specific industrial processes. By clearly evaluating the strengths and weaknesses of different neural operator models—such as their performance and efficiency—practitioners can make frictionless and well-informed decisions when selecting network architectures. Furthermore, constructing dedicated databases to train generative AI models that assist practitioners in deploying, configuring, and training problem-oriented, customized operator neural networks will be highly valuable. Such developments would enable an automated, end-to-end workflow from problem formulation to practical application, representing a promising direction for the field.

LITERATUR

[1] Ferziger, J. H.; Perić, M.: Computational Methods for Fluid Dynamics. Heidelberg: Springer 2002

[2] Wang, S.; Wang, H.; Perdikaris, P.: Learning the solution operator of parametric partial differential equations with physics-informed DeepONets. Science Advances 7 (2021) 40, eabi8605

[3] Lu, L.; Jin, P.; Pang, G. et al.: Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. Nature Machine Intelligence 3 (2021), pp. 218–229

[4] Yan, B.; Harp, D. R.; Chen, B. et al.: A physics-constrained deep learning model for simulating multiphase flow in 3D heterogeneous porous media. Fuel 313 (2022), #122693

[5] Cai, S.; Wang, Z.; Lu, L. et al.: DeepM&Mnet: Inferring the electroconvection multiphysics fields based on operator approximation by neural networks. Journal of Computational Physics 436 (2021), #110296

[6] Nasrin, S.; Shylendra, A.; Darabi, N. et al.: ENOS: Energy-Aware Network Operator Search in Deep Neural Networks. IEEE Access 10 (2022), pp. 81447–81457

[7] Stephany, R.; Earls, C.: PDE-READ: Human-readable partial differential equation discovery using deep learning. Neural Networks 154 (2022), pp. 360–382

[8] Gao, H.; Zahr, M. J.; Wang, J.-X.: Physics-informed graph neural Galerkin networks: A unified framework for solving PDE-governed forward and inverse problems. Computer Methods in Applied Mechanics and Engineering 390 (2022), #114502

[9] Sirignano, J.; Spiliopoulos, K.: DGM: A deep learning algorithm for solving partial differential equations. Journal of Computational Physics 375 (2018), pp. 1339–1364

[10] Kochkov, D.; Smith, J. A.; Alieva, A. et al.: Machine learning-accelerated computational fluid dynamics. Proceedings of the National Academy of Science 118 (2021), e2101784118

[11] Moya, C.; Zhang, S.; Lin, G. et al.: DeepONet-grid-UQ: A trustworthy deep operator framework for predicting the power grid's post-fault trajectories. Neurocomputing 535 (2023), pp. 166–182

[12] Lin, G.; Moya, C.; Zhang, Z.: B-DeepONet: An enhanced Bayesian DeepONet for solving noisy parametric PDEs using accelerated replica exchange SGLD. Journal of Computational Physics 473 (2023), #111713

[13] Li, Z.; Kovachki, N.; Azizzadenesheli, K. et al. (Hrsg.): Fourier Neural Operator for Parametric Partial Differential Equations. ICLR 2021, arXiv 2021, https://doi.org/10.48550/arXiv.2010.08895

[14] Deng, Z.; Liu, H.; Shi, B. et al.: Temporal predictions of periodic flows using a mesh transformation and deep learning-based strategy. Aerospace Science and Technology 134 (2023), #108081

[15] Wen, G.; Li, Z.; Azizzadenesheli, K. et al.: U-FNO—An enhanced Fourier neural operator-based deep-learning model for multiphase flow. Advances in Water Resources 163 (2022), #104180

[16] Li, Z.; Kovachki, N.; Azizzadenesheli, K. et al.: Neural Operator: Graph Kernel Network for Partial Differential Equations. arXiv 2020, https://doi.org/10.48550/arXiv.2003.03485

[17] Koric, S.; Abueidda, D. W.: Data-driven and physics-informed deep learning operators for solution of heat conduction equation with parametric heat source. International Journal of Heat and Mass Transfer 203 (2023), #123809

[18] Li, Z.; Zheng, H.; Kovachki, N. et al.: Physics-Informed Neural Operator for Learning Partial Differential Equations. ACM/IMS Journal of Data Science,1, (2024) 3, https://doi.org/10.1145/3648506

[19] Lu, L.; Meng, X.; Cai, S. et al.: A comprehensive and fair comparison of two neural operators (with practical extensions) based on FAIR data. Computer Methods in Applied Mechanics and Engineering 393 (2022), #114778

[20] Kushwaha, S.; Park, J.; Koric, S. et al.: Advanced deep operator networks to predict multiphysics solution fields in materials processing and additive manufacturing. Additive Manufacturing 88 (2024), #104266

[21] Li, Z.; Kovachki, N.; Choy, C. et al.: Geometry-Informed Neural Operator for Large-Scale 3D PDEs, 2023, pp. 35836–35854

[22] Zhang, G.; Xu, T.; Xu, J. et al.: A Graph-Structured, Physics-Informed DeepONet Neural Network for Complex Structural Analysis. Machine Learning and Knowledge Extraction 7 (2025) 4, #137

[23] Zhao, Z.; Li, Z.; Hassibi, K. et al.: Physics-informed Neural-operator Predictive Control for Drag Reduction in Turbulent Flows. arXiv 2025, https://doi.org/10.48550/arXiv.2510.03360

Lichu Wang, M.Sc.
lichu.wang@uni-a.de

Prof. Dr.-Ing. Johannes Schilp
johannes.schilp@informatik.uni-augsburg.de

Universität Augsburg
Lehrstuhl für Ingenieurinformatik – Produktionsinformatik
Universitätsstr. 2, 86159 Augsburg
www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/pi/