Variants of Operator Learning

SEBASTIAN HECKERS, ETH Zürich, Switzerland

Operator learning is an emerging field in computational science and engineering. Traditional neural networks, typically mapping finite-dimensional Euclidean spaces, have been extended to infinite-dimensional spaces to learn solution operators of partial differential equations (PDEs). Various approaches are examined, including encoder-decoder networks, neural operators, and transformers. This research highlights the significant advancements made by these methods, the applications in different fields, and the practical benefits of these advanced models.

CCS Concepts: • **Computing methodologies → Machine learning approaches**; **Neural networks**.

Additional Key Words and Phrases: Neural Operator, Fourier neural operator, Convolutional neural operator, Graph neural operator, Encoder-Decoder Network, Transformer

1 INTRODUCTION TO OPERATOR LEARNING

Many complex phenomena are solved with similar complex and expensive numerical solvers. This standard solvers find numerical approximations to diverse partial differential equations (PDEs). Using data-driven machine learning (ML) techniques gives faster and similar accurate solving methods for even unknown PDEs. Operator learning tries to learn solution operators of PDEs to generalize neural networks for mappings between infinite-dimensional spaces. In recent years various methods have been developed to approach operator learning which are able to be discretization and resolution invariant.

2 ENCODER-DECODER NETWORKS

One approach for operator learning is encoder-decoder-nets. They use an encoder and decoder to map the input functions of the operator. The architecture has 3 steps: encoding the input function as a finite-dimensional vector, mapping the encodings with a neural network, and decoding it to the output space. The 3-step approach is similar to the numerical methods, like finite differences or finite elements, which also use encoding, finite-dimensional mapping, and decoding, but instead of a handcrafted algorithm a neural network is used for encoder-decoder-nets.

Encoder-decoder structures occur in almost all methods of operator learning. Some special instances are DeepONet and PCA-Net. Deep operator networks (DeepONets or DON) are encoder-decoder-nets to learn operators accurately from small datasets. [17] introduced as DONs, made of encoders as the branch-trunk network. The branch net encodes the input functions as sensor values in specific places and the trunk net encodes the location of the output function. The mapped vectors are then merged and decoded to the output space. The MIONet from [7] is an advanced model related to DON. [2] introduces the PCA-Net and combines neural networks and deep learning with model reduction. It has the architecture of encoder-decoder-nets and uses principal component analysis (PCA) as encoder and decoder. So this network can learn mappings between Hilbert spaces. In numerical experiments from [2] the PCA-Net showed good numerical performance and mesh-independence.

3 TRANSFORMERS

Operator Transformers are Transformers for operator learning. They are attention-based frameworks for data-driven learning. Transformers and the attention mechanism is a concept from natural language processing and introduced by [21]. It creates three vectors out of the input sequence,

Author's address: Sebastian Heckers, sheckers@ethz.com, ETH Zürich, Zürich Stadt, Zürich, Switzerland.

namely the query, key and value. The self-attention mechanism calculates the weighted sum of this values to get the similarity between the encodings. This method was then translated to operator learning to exploit the patterns within the input. In [4] they proposed two variants of an Operator Transformer. The architecture is similar to encoder-decoder-nets, but it uses a stack of identical simple attention-based layers. The first variant is termed the Fourier Transformer and is the vanilla attention with the standard softmax attention. The second variant has the same architecture but without softmax. Instead, it uses a new layer normalization modeled after the Petrov-Galerkin projection and is therefore called the Galerkin Transformer (GT). The latter approach showed great accuracy and less training cost compared to FNO.

The Transformer as introduced in [4] has some major problems, because it cannot handle adequately irregular mesh, multiple inputs, and multi-scale problems. New models try to address these problems. In [14] the transformer OFormer is introduced. It uses a fully point-based attention mechanism which has better flexibility for input and output discretizations and generalizes to diverse instances of PDEs. The General Neural Operator Transformer (GNOT) is a scalable and effective transformer-based framework, introduced by [6]. It introduced the Heterogeneous Normalized Attention Block. This block uses different MLPs to compute the keys and values for the cross-attention between the query points and conditional embeddings. The model has the architecture of encoder-decoder-nets with several Heterogeneous Normalized Attention blocks and normalized self-attention layers. It achieved great performance compared to other state-of-the-art methods.

4 NEURAL OPERATORS

Neural Operators try to learn the mapping between the infinite spaces with special operator layers. They are a composition of operator layers and nonlinear activation functions. Different variants of neural operators are the different design choices for the linear operator layers. The paper [8] introduced the neural operators and 4 different classes of operator layers. This class should guarantee both discretization-invariance and universal approximation.

4.1 Graph Neural Operator (GNO)

The graph neural operator is a graph kernel network for partial differential equations. The operator layer is a parameterized integral operator. The integration is approximated by a Monte Carlo sum via a message passing graph network and uses Nyström Approximation for better scaling. [11] demonstrated that this method can achieve competitive performance with other mesh-free, numerical analytic approaches and beat state-of-the-art mesh-dependent neural networks on large grids. However, the disadvantages are the performance and memory usage during runtime, which are significant worse than other NOs.

4.2 Fourier Neural Operator (FNO)

The Fourier neural operator is formulated by a parameterized integral directly in Fourier space. The operator layer is the Fourier integral operator layer where a convolution in Fourier space replaces the integration kernel. Each Fourier layer applies the Fast Fourier Transform (FFT), a linear transform and the inverse Fourier transform. This is combined with a skip connection and MLPs. The FNO [12] achieves a significant performance improvement compared to traditional PDE solvers and has superior accuracy compared to other neural networks with fixed spatial resolution. This allows zero-shot super-solution for the mesh-invariant NO. [9] classifies FNO as a special case from the class of Nonlocal Neural Operators. This outlines the nonlocality and nonlinearity of FNOs.

The initial version of the FNO has been extended and improved to overcome some shortcomings or adapt it to special applications. A problem of the FNO was that it only operates on specific domains, see section 4.2.1. [3] introduced the Spherical FNO (SFNO), which learns with spherical geometries and can better generalize operators on a sphere. It uses the Spherical Harmonic Transform instead of the FFT. Also, successful approaches from other fields of machine learning are applied to FNOs, see section 4.3. Another is the Adaptive Fourier Neural Operator (AFNO) proposed by [5]. It is used for problems with discontinuities and high-resolution inputs. The AFNO implements an effective token mixer in the Fourier domain by using a sparse, block-diagonal structure in the Fourier layer.

4.2.1 Irregular Domains. The Fast Fourier Transform, used in FNOs, is limited to rectangular domains with uniform, equispaced grids. To use the FNO on irregular, deformed domains [10] and [16] present new methods. In [10] the new framework of geometry-aware FNOs is presented. The Geo-FNO uses a deformation neural network to map the input domain into a latent space with a uniform grid. Then the regular FNO is applied to the latent space. This solution has the advantages of having the computational efficiency of the FFT and the flexibility to handle any geometry. Compared to numerical solver it is significantly faster and more accurate than other neural operators, which use interpolation. To overcome the problem of equispaced grids in FFT [16] introduced a modified FNO on non-equispaced distributed points. A matrix multiplication is used to efficiently construct Vandermonde-structured matrices to compute the forward pass and inverse transform for these distributions. This method is termed the Structured Matrix Method (SMM) also known as Vandermonde Neural Operator. The benchmarks in [16] showed that SMM outperforms the baselines of other comparable neural operators by accuracy and training speed.

4.2.2 Physics-Informed Neural Operator for Learning Partial Differential Equations (PINO). The PINO combines operator learning with the physics constraints of Physical-informed neural networks (PINN). This network includes training data to facilitate the optimization and therefore the data loss and equation loss are both used. In [15] the neural operator is based on FNO and uses instance-wise learning. This approximates the operator of many PDEs nearly perfectly and improves the generalization and physical correctness, but it has the same disadvantages as PINNs or FNOs like slow convergence or constraints from the FFT.

4.3 Convolutional Neural Operator (CNO)

Convolutional neural networks are very successful in machine learning tasks but have been largely ignored for operator learning. [19] adapted convolutional neural networks to demonstrate their use in neural operators. The operator layers consist of carefully designed elementary mappings. It has a convolution operator, an activation operator, and an up- or downsampling operator. They are applied on bandlimited-functions to avoid aliasing and to properly use convolution and activation functions. The convolution operator is parameterized by a discrete kernel directly in physical space. The CNO outperformed traditional numerical methods and even FNO in representative PDE benchmarks. It also performed by out-of-distribution testing and generalization on unseen data, like zero-shot resolution manner. However, the CNO operates on a two-dimensional Cartesian domain and is computationally demanding in three-space dimensions or needs transformation for non-Cartesian domains.

5 APPLICATION

Since partial differential equations (PDEs) occur in every field of science and engineering, therefore operator learning can be used everywhere. The neural operator is trained often on many examples from a wide range of fields. Especially problems from fluid dynamics are tested. A good, very well-known application is the FourCastNet from [18]. It stands for Fourier Forecasting Neural Network and is a global data-driven weather forecasting model. The architecture uses the AFNO with Vision Transformer backbone which yields a state-of-the-art high-resolution and shows fine-grained features. The experiments in [18] showed that the FourCastNet achieved accurate high-resolution with computation time an order of magnitude faster than a state-of-the-art numerical weather prediction model. The FourCastNet is further developed and tested with the spherical geometries as in the SFNO. The recent breakthroughs in ML by neural operators have inspired applications in all fields of science, for example, material deformation. In [20] the FNO is used to simulate advanced composite materials with complex hierarchical designs. In [13] a NO is used for industry-standard aerodynamics on 3D vehicle geometries. More applications are listed in [1] and [22].

6 COMPARISON OF RESULTS

Different models for operator learning are compared in [6] with datasets from [12] and [10]. In [19] exhaustive benchmarking with various models on a new, representative set of standard PDEs in ML was done. They used the best model of different methods by fine-tuning the hyperparameters and testing them inside the distribution. The results are summarized in table 1.

Dataset	FFNN	UNet	ResNet	DON/MIONet	FNO	Geo-FNO	CNO	GT	OFormer	GNOT
Darcy2d	–	–	–	5.45%	1.09%	1.09%	–	**0.84%**	1.24%	1.05%
NS2d	–	–	–	–	8.20%	8.20%	–	7.92%	6.46%	**4.43%**
Elasticity	–	–	–	9.65%	5.08%	2.20%	–	2.01%	1.83%	**0.86%**
NS2d-c	–	–	–	2.74%	6.56%	1.41%	–	1.52%	2.33%	**0.67%**
NACA	–	–	–	13.2%	4.21%	1.38%	–	1.61%	1.83%	**0.75%**
Inductor2d	–	–	–	3.10%	–	–	–	25.6%	2.23%	**1.21%**
Poisson Eq.*	5.74%	0.71%	0.43%	12.9%	4.98%	–	**0.21%**	2.77%	–	–
Wave Eq.*	2.51%	1.51%	0.79%	2.26%	1.02%	–	**0.63%**	1.44%	–	–
Smooth Eq.*	7.09%	0.49%	0.39%	1.14%	0.28%	–	**0.24%**	0.98%	–	–
Discontinuous Eq.*	13.0%	1.31%	1.01%	5.78%	1.15%	–	**1.01%**	1.55%	–	–
Allen-Cahn Eq.*	18.3%	0.82%	1.40%	13.6%	**0.28%**	–	0.54%	0.77%	–	–
Navier-Stokes Eq.*	8.05%	3.54%	3.69%	11.6%	3.57%	–	**2.76%**	4.14%	–	–
Darcy Flow*	2.14%	0.54%	0.42%	1.13%	0.80%	–	**0.38%**	0.86%	–	–
Compressible Euler*	0.78%	0.38%	1.70%	1.93%	0.44%	–	**0.35%**	2.09%	–	–

Table 1. The results of operator learning on several datasets from multiple areas. ”-“ means that no result of this test case is available. The best results are **bolded**. The relative mean L^2 test errors is used, except by datasets with an asterix ”*“, they used relative median L^1 error.

7 DISCUSSION

The results from table 1 use different datasets, but other sources show the same outcomes. The standard FFNN often has a higher error and cannot generalize well. Encoder-decoder networks, like DONs, achieve a smaller error than FFNN, especially by out-of-distribution testing, but other methods can beat that. Models of Transformer and Neural Operators are significantly better than DONs and can generalize for out-of-distribution quite well. They achieve similar errors, but the FNO can often outdo the standard transformers. Advanced classical methods, like UNet or ResNet, get similar errors to the methods for operator learning and can generalize surprisingly well. The best methods, surpassing all other models, are the advanced models GNOT and CNO. These are some of the newest models, which combine the advantages of the previous models, but these results were presented by papers introducing the models and have not been tested against each other yet.

7.1 Conclusion

Many models for operator learning exist and the best method might not be found yet. But some models are outstanding, for example, the FNO, CNO, or GNOT. They have achieved great results and surpassed other models. This development would continue for the remaining problems. These are the complex multi-scale PDEs and high training costs. Further advancements could be better architecture designs and extensions for these problems inspired by other ML or numerical techniques.

REFERENCES

[1] Kamyar Azizzadenesheli, Nikola Kovachki, Zongyi Li, Miguel Liu-Schiaffini, Jean Kossaifi, and Anima Anandkumar. 2024. Neural Operators for Accelerating Scientific Simulations and Design. arXiv:2309.15325 [cs.LG]

[2] Kaushik Bhattacharya, Bamdad Hosseini, Nikola B. Kovachki, and Andrew M. Stuart. 2021. Model Reduction and Neural Networks for Parametric PDEs. arXiv:2005.03180 [math.NA]

[3] Boris Bonev, Thorsten Kurth, Christian Hundt, Jaideep Pathak, Maximilian Baust, Karthik Kashinath, and Anima Anandkumar. 2023. Spherical Fourier Neural Operators: Learning Stable Dynamics on the Sphere. arXiv:2306.03838 [cs.LG]

[4] Shuhao Cao. 2021. Choose a Transformer: Fourier or Galerkin. arXiv:2105.14995 [cs.LG]

[5] John Guibas, Morteza Mardani, Zongyi Li, Andrew Tao, Anima Anandkumar, and Bryan Catanzaro. 2022. Adaptive Fourier Neural Operators: Efficient Token Mixers for Transformers. arXiv:2111.13587 [cs.CV]

[6] Zhongkai Hao, Zhengyi Wang, Hang Su, Chengyang Ying, Yinpeng Dong, Songming Liu, Ze Cheng, Jian Song, and Jun Zhu. 2023. GNOT: A General Neural Operator Transformer for Operator Learning. arXiv:2302.14376 [cs.LG]

[7] Pengzhan Jin, Shuai Meng, and Lu Lu. 2022. MIONet: Learning multiple-input operators via tensor product. arXiv:2202.06137 [cs.LG]

[8] Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. 2023. Neural Operator: Learning Maps Between Function Spaces. arXiv:2108.08481 [cs.LG]

[9] Samuel Lanthaler, Zongyi Li, and Andrew M. Stuart. 2023. The Nonlocal Neural Operator: Universal Approximation. arXiv:2304.13221 [math.NA]

[10] Zongyi Li, Daniel Zhengyu Huang, Burigede Liu, and Anima Anandkumar. 2022. Fourier Neural Operator with Learned Deformations for PDEs on General Geometries. arXiv:2207.05209 [cs.LG]

[11] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. 2020. Neural Operator: Graph Kernel Network for Partial Differential Equations. arXiv:2003.03485 [cs.LG]

[12] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. 2021. Fourier Neural Operator for Parametric Partial Differential Equations. arXiv:2010.08895 [cs.LG]

[13] Zongyi Li, Nikola Borislavov Kovachki, Chris Choy, Boyi Li, Jean Kossaifi, Shourya Prakash Otta, Mohammad Amin Nabian, Maximilian Stadler, Christian Hundt, Kamyar Azizzadenesheli, and Anima Anandkumar. 2023. Geometry-Informed Neural Operator for Large-Scale 3D PDEs. arXiv:2309.00583 [cs.LG]

[14] Zijie Li, Kazem Meidani, and Amir Barati Farimani. 2023. Transformer for Partial Differential Equations' Operator Learning. arXiv:2205.13671 [cs.LG]

[15] Zongyi Li, Hongkai Zheng, Nikola Kovachki, David Jin, Haoxuan Chen, Burigede Liu, Kamyar Azizzadenesheli, and Anima Anandkumar. 2023. Physics-Informed Neural Operator for Learning Partial Differential Equations. arXiv:2111.03794 [cs.LG]

[16] Levi Lingsch, Mike Michelis, Emmanuel de Bezenac, Sirani M. Perera, Robert K. Katzschmann, and Siddhartha Mishra. 2023. A Structured Matrix Method for Nonequispaced Neural Operators. arXiv:2305.19663 [cs.LG]

[17] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. 2021. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence* 3, 3 (March 2021), 218–229. https://doi.org/10.1038/s42256-021-00302-5

[18] Jaideep Pathak, Shashank Subramanian, Peter Harrington, Sanjeev Raja, Ashesh Chattopadhyay, Morteza Mardani, Thorsten Kurth, David Hall, Zongyi Li, Kamyar Azizzadenesheli, Pedram Hassanzadeh, Karthik Kashinath, and Animashree Anandkumar. 2022. FourCastNet: A Global Data-driven High-resolution Weather Model using Adaptive Fourier Neural Operators. arXiv:2202.11214 [physics.ao-ph]

[19] Bogdan Raonić, Roberto Molinaro, Tim De Ryck, Tobias Rohner, Francesca Bartolucci, Rima Alaifari, Siddhartha Mishra, and Emmanuel de Bézenac. 2023. Convolutional Neural Operators for robust and accurate learning of PDEs. arXiv:2302.01178 [cs.LG]

[20] Meer Mehran Rashid, Tanu Pittie, Souvik Chakraborty, and N. M. Anoop Krishnan. 2022. Learning the Stress-Strain Fields in Digital Composites using Fourier Neural Operator. arXiv:2207.03239 [cond-mat.mtrl-sci]

[21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[22] Hrishikesh Viswanath, Md Ashiqur Rahman, Abhijeet Vyas, Andrey Shor, Beatriz Medeiros, Stephanie Hernandez, Suhas Eswarappa Prameela, and Aniket Bera. 2023. Neural Operator: Is data all you need to model the world? An insight into the impact of Physics Informed Machine Learning. arXiv:2301.13331 [cs.AI]