Reduced-rank Factorized Fourier Neural Operator

Chieh-An Chou d114053001@mail.nchu.edu.tw

Lu-Hung Chen luhung@email.nchu.edu.tw

Department of Applied Mathematics, National Chung Hsing University, Taichung 402, Taiwan

Editors: Hung-yi Lee and Tongliang Liu

Abstract

We present R^2-FFNO, a novel neural operator architecture designed to address the overparameterization common in Factorized Fourier Neural Operators (FFNO) through reduced-rank factorization of spectral components. While neural operators are effective for learning solutions to partial differential equations (PDE), their architectures often contain an excessive number of parameters, which can lead to overfitting and diminished generalization capabilities. Inspired from reduced-rank learning techniques, the R^2-FFNO approach decomposes spectral kernels into lower-rank representations, enabling systematic control over the model's capacity. This low-rank factorization facilitates a balance between the model's expressiveness and capability of generalization. Empirical analysis reveals that performance saturates once an optimal rank is achieved and degrades if the rank is increased beyond this point. This observation highlights an optimal trade-off between model complexity and accuracy, underscoring the importance of principled rank selection in designing neural operators.

To further enhance performance, a targeted data augmentation strategy is utilized. This strategy introduces high-frequency variations during training to address spectral bias, thereby enhancing the model's capacity to resolve fine-scale PDE dynamics. A comprehensive evaluation on benchmark datasets confirms the efficacy of the R^2-FFNO. Compared to the original FFNO architecture, R^2-FFNO demonstrates significant error reductions: 46.5% reduction in the Navier-Stokes problem, 31.6% in the Kolmogorov Flow problem, and 34.7% in the Darcy Flow problem. The proposed method offers a principled framework for managing overparameterization in neural operators, contributing to the development of more efficient and generalizable PDE solvers for wide application in scientific computing. Our code is available at https://github.com/Chieh997/R2FFNO.

Keywords: Factorized Fourier neural operator; low-rank learning; scientific computing.

1. Introduction

Partial differential equations (PDE) represent the foundational mathematical framework for modeling complex physical phenomena in diverse disciplines of science and engineering, from fluid dynamics (Herron and Foster, 2008) and structural mechanics (Öchsner, 2020) to climate science (Stocker, 2011). Despite their ubiquitous importance, the solution of these equations for real-world applications remains computationally intensive. Traditional numerical solvers, while accurate, face significant limitations in computational efficiency when handling multidimensional problems (Julien and Watson, 2009) or when requiring rapid solutions for time-sensitive applications.

The emergence of neural operators as surrogate models for PDE solutions has revolutionized scientific computing by enabling dramatic speedups while maintaining acceptable

© 2025 C.-A. Chou & L.-H. Chen.

accuracy (Azizzadenesheli et al., 2024). Among these approaches, the Fourier Neural Operator (FNO) (Li et al., 2021) has demonstrated remarkable capabilities in learning mappings between infinite-dimensional function spaces. The FNO leverages the efficiency of the fast Fourier transform (FFT) to capture long-range spatial dependencies with $O(n \log n)$ computational complexity, making it particularly suitable for multidimensional problems such as fluid dynamics.

Building upon this foundation, the Factorized Fourier Neural Operator (FFNO) (Tran et al., 2023) represented a significant advance by introducing separable spectral layers through Fourier factorization, which processes each spatial dimension independently in the Fourier domain before combining them in physical space. This innovation, coupled with improved residual connections and improved training strategies, reduced the model complexity by an order of magnitude while simultaneously improving accuracy.

In this article, we contend that the separation of layers in the FFNO architecture may result in overparameterization. To investigate this issue, we conduct an analysis of the eigenvalues associated with the linear mappings in FFNO, thereby providing empirical support for the overparameterization hypothesis. Furthermore, we propose a Reduced-Rank FFNO (R^2-FFNO) in which each learnable linear mapping is decomposed into the product of two lower-dimensional learnable matrices. Additionally, to mitigate high-frequency spectral bias, we employed a data augmentation strategy by introducing high-frequency data variations. Our experimental results demonstrate that the R^2-FFNO achieves a reduction in error ranging from 31% to 46% relative to the original FFNO, and attains state-of-the-art performance in our simulation studies.

2. Related Works

Classical numerical partial differential equation solvers form the backbone of computational science and engineering, providing robust and accurate methods for solving complex mathematical models. These established techniques have been developed over several centuries, with foundational contributions from mathematicians like Euler, Lagrange, and Fourier, and continue to be essential tools for modeling real-world problems. The foundation of classical PDE solvers lies in discretization schemes that transform continuous differential equations into solvable algebraic systems. The finite difference method represents the most straightforward approach which approximates derivatives using differences between function values at discrete grid points. This method excels in simple geometries and is particularly effective for structured problems, though it can struggle with complex boundary conditions. The finite element method (FEM) offers greater flexibility by dividing the computational domain into small, nonoverlapping elements and using piecewise polynomial functions to approximate solutions. The strength of FEM lies in handling complex geometries and varying boundary conditions, making it the preferred choice for structural analysis and engineering applications. Spectral methods discretize the equations by utilizing global basis functions such as trigonometric polynomials or orthogonal polynomials. These methods achieve exponential convergence rates for sufficiently smooth solutions, making them particularly valuable in fluid dynamics and atmospheric modeling.

Although these traditional methods have proven invaluable for scientific computing, they share fundamental limitations that become pronounced in modern applications. Each

method solves individual PDE instances with fixed parameters, requiring complete recomputation for different initial conditions, boundary conditions, or parameter values. This instance-by-instance approach becomes computationally prohibitive when exploring higher-dimensional problems whose parameter spaces increase exponentially, such as atmospheric modeling or when real-time predictions are required. Furthermore, traditional solvers exhibit inherent speed-accuracy trade-offs based on spatial resolution, often requiring extremely fine meshes to achieve desired accuracy, particularly for turbulent flows or multiscale phenomena.

Neural Operators. Neural operators emerged as a revolutionary approach to overcome the limitations of traditional PDE solvers by learning mappings between infinite-dimensional function spaces rather than finite-dimensional vector spaces (Kovachki et al., 2024). This fundamental shift enables neural networks to learn the solution of PDEs from data, contrasting sharply with classical methods that solve one equation instance at a time. DeepONet (Lu et al., 2021) stands as a pioneering architecture in neural operator learning, characterized by its two sub-networks that encode different aspects of the operator mapping. The branch network encodes the input function at a fixed number of sensor locations, while the trunk network encodes the spatial coordinates where the output function is evaluated. This decomposition allows DeepONet to handle functions as inputs—infinite-dimensional objects—and map them to the corresponding output functions in the solution space. Graph neural operators (Li et al., 2020a,b) extend the concept of neural operator by incorporating graph structures to handle irregular geometries and variable discretizations.

Convolutional neural networks and the U-Net architecture, traditionally successful in image domains, have been pivotal in the advancement of neural operators. For example, convolutional neural operators (Raonić et al., 2023) employ convolutional layers to model mappings between high-dimensional function spaces, often achieving robust and accurate performance on PDE benchmarks. The U-shaped neural operator (Rahman et al., 2024) integrates the U-Net architecture into the neural operator framework. By using the skip connections that merge multi-scale feature maps, the U-shaped neural operator has achieved significant performance gains.

FNO and its Variants. The Fourier Neural Operator (FNO) (Li et al., 2021) represents a landmark achievement in neural operator design, introducing a novel architecture that parameterizes integral kernels directly in Fourier space. This approach leverages the computational efficiency of FFT while maintaining the theoretical rigor of neural operator frameworks. The FNO architecture consists of multiple Fourier layers, each performing a linear transformation in the frequency domain, followed by an inverse fast Fourier transform (IFFT) and non-linear activation. The key innovation lies in learning the kernel weights in Fourier space, which enables efficient computation of global convolutions that would be prohibitively expensive in spatial domains. The resolution-invariance property stems from FNO's parameterization in function space rather than discrete space, enabling the same learned parameters to operate across different discretization levels. Experimental results show that FNO achieves up to three orders of magnitude speedup compared to traditional PDE solvers while maintaining competitive accuracy.

The exponential parameter growth with increasing dimensionality has been a persistent challenge for FNO in high-resolution applications. Recent developments have introduced sophisticated parameterization strategies to address these scalability concerns. Adaptive

Fourier Neural Operator (AFNO) (Guibas et al., 2022) introduced the concept of patch embedding, processing input functions in localized patches. Crucially, AFNO also refined the parameterization of the spectral filters by applying weight parameters separately to the real and imaginary parts of the Fourier transform, allowing for more flexible and expressive kernel learning. The Factorized Fourier Neural Operator (FFNO) (Tran et al., 2023), addressed the computational challenges of high-dimensional inputs by factorizing the weight parameters. Instead of operating on a single high-dimensional Fourier transform, FFNO applies independent one-dimensional Fourier transforms across each dimension and learns respective weight parameters for each, significantly reducing computational complexity and memory footprint. The Amortized Fourier Neural Operator (AM-FNO) (Xiao et al., 2024) represents a significant advancement in parameter efficiency, utilizing neural networks to approximate the mapping between frequencies and function values rather than maintaining separate parameters for each frequency mode. This approach enables the handling of arbitrarily many frequency modes using a fixed number of parameters.

Recent spectral analysis has revealed important insights into FNO's learning characteristics, showing that FNO exhibits significant capability in learning low-frequency information but faces challenges with high-frequency components (Qin et al., 2024). Recent research has developed several innovative approaches to address this fundamental limitation. The SpecBoost framework (Qin et al., 2024) represents a breakthrough in ensemble learning for neural operators, employing multiple FNOs to better capture high-frequency information through a residual learning approach. The Conv-FNO architecture (Liu et al., 2025) introduces convolutional neural network preprocessing to extract local spatial features directly from input data, effectively bridging the gap between local and global representations. The U-FNO architecture (Wen et al., 2022) enhances the experssiveness of FNO by appending a mini U-Net path to the Fourier layer. The Wavelet Neural Operator (WNO) (Tripura and Chakraborty, 2023) extends the Fourier approach by incorporating wavelet transformations, which provide superior time-frequency localization compared to pure Fourier methods. WNO harnesses wavelets' ability to capture both spatial and frequency information simultaneously, enabling more effective learning of functional mappings in problems with localized features or multi-scale phenomena. The wavelet basis functions' localization properties make WNO particularly effective for problems exhibiting sharp gradients, discontinuities, or hierarchical structures that pure Fourier methods might struggle to represent accurately.

3. Reduced-Rank Fourier Neural Operator (R^2-FFNO)

The fundamental FNO architecture is designed to learn the complex, non-linear mapping $\mathcal{G}$ from input fields $a(x,t)$ to spatio-temporal solutions $u(x,t)$ across diverse PDE systems. Motivated by the overparameterization hypothesis of the FFNO architecture, we propose a novel approach that addresses this challenge through reduced-rank factorization. To establish the theoretical foundation for our contributions, we first present the mathematical formulation of the standard FNO and its factorized variant (FFNO), which serve as the baseline architectures for our proposed enhancements.

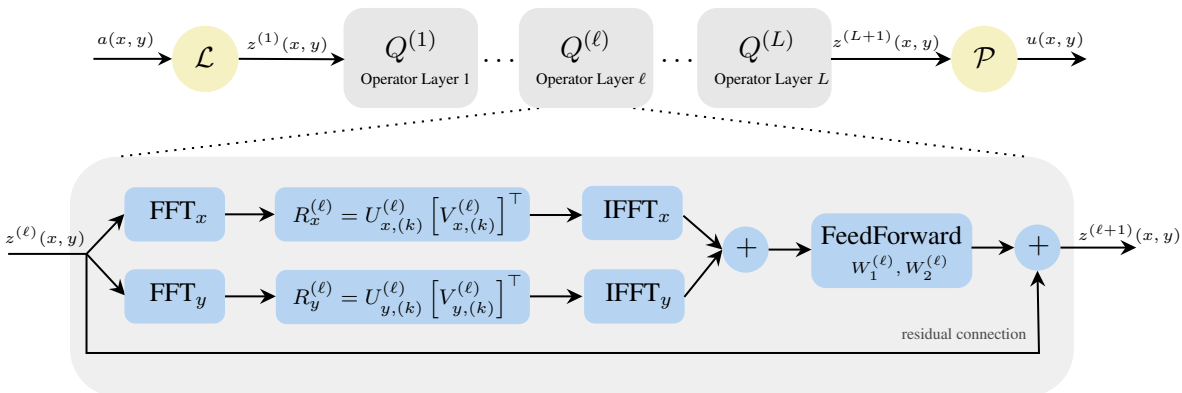


Figure 1: The architecture of our proposed Reduced-Rank Factorized Fourier Neural Operator (R^2-FFNO). This diagram extends the FFNO framework by introducing low-rank factorization for the learnable spectral kernels $R_x^{(\ell)}$ and $R_y^{(\ell)}$ within each operator layer $Q^{(\ell)}$. Specifically, each kernel $R_d^{(\ell)}$ is decomposed with rank k as $R_d^{(\ell)} = U_{d,(k)}^{(\ell)} \left[V_{d,(k)}^{(\ell)}\right]^\top$.

3.1. Fourier Neural Operator (FNO)

The FNO model proposed a mapping $\mathcal{G}$ from an input function a to an output function u, structured as:

$$\mathcal{G} = \mathcal{P} \circ \mathcal{Q}^{(L)} \circ \cdots \circ \mathcal{Q}^{(1)} \circ \mathcal{L}, \tag{1}$$

where $\circ$ is function composition, $\mathcal{L}$ is a lifting operator that maps the input function to a higher-dimensional representation, followed by an iterative application of L learnable Fourier operator layers $\mathcal{Q}^{(\ell)}$, and P is a projection operator that maps the final hidden state back to the output function space.

Each operator layer $\mathcal{Q}^{(\ell)}$ processes the hidden state $z^{(\ell)} \mapsto z^{(\ell+1)}$ by:

$$z^{(\ell+1)} = \mathcal{Q}^{(\ell)}(z^{(\ell)}) = \sigma(W^{(\ell)} z^{(\ell)} + b^{(\ell)} + \mathcal{K}^{(\ell)}(z^{(\ell)})), \tag{2}$$

where σ is a non-linear activation function, $W^{(\ell)}$ is a linear transformation matrix, and $b^{(\ell)}$ is a bias vector. The kernel integral operator $\mathcal{K}^{(\ell)}$ performs global convolution efficiently in the Fourier domain:

$$\mathcal{K}^{(\ell)}(z^{(\ell)}) = \mathcal{F}^{-1}(R^{(\ell)} \cdot \mathcal{F}(z^{(\ell)})), \tag{3}$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ represent the D-dimensional FFT and its inverse transformation (IFFT), respectively, and R is the learnable Fourier domain weighted parameter, often referred to as a spectral convolution kernel.

3.2. Factorized Fourier Neural Operator (FFNO)

The FFNO introduces a modified operator layer $\mathcal{Q}^{(\ell)}$ with a residual connection and a more complex non-linearity to enhance stability and performance:

$$\mathcal{Q}^{(\ell)}(z^{(\ell)}) = z^{(\ell)} + \sigma(W_2^{(\ell)} \sigma(W_1^{(\ell)} \mathcal{K}^{(\ell)}(z^{(\ell)}) + b_1^{(\ell)}) + b_2^{(\ell)}) \tag{4}$$

where $W_1^{(\ell)}$, $W_2^{(\ell)}$ are the weight matrices, and $b_1^{(\ell)}$, $b_2^{(\ell)}$ are bias vectors.

The key innovation in FFNO lies in factorizing the kernel integral $\mathcal{K}^{(\ell)}$. Instead of computing a full-dimensional Fourier transform, FFNO approximates the kernel integral by summing independent one-dimensional Fourier transforms across each dimension $d \in D$:

$$\mathcal{K}^{(\ell)}(z^{(\ell)}) = \sum_{d \in D} \mathcal{F}_d^{-1}(R_d^{(\ell)} \cdot \mathcal{F}_d(z^{(\ell)})), \tag{5}$$

where $\{\mathcal{F}_d\}_{d \in D}$ and $\{\mathcal{F}_d^{-1}\}_{d \in D}$ denote the one-dimensional FFT and IFFT along dimension d respectively, and $R_d^{(\ell)}$ is the learnable spectral convolution kernel specific to that dimension.

3.3. Proposed Method: Reduced-Rank Fourier Neural Operator ($\mathbf{R}^2$-FFNO)

Our proposed Reduced-Rank Factorized Fourier Neural Operator (R^2-FFNO), depicted in Figure 1, constrains model capacity through reduced-rank factorization of the spectral kernels. This design is motivated by an eigenvalue analysis of the FFNO architecture (see Figure 2 and detailed in Section 4.2), which empirically supports the presence of overparameterization in the spectral domain. The underlying principle of reduced-rank factorization is inspired by techniques such as reduced-rank regression (Reinsel et al., 2022) and recent developments in parameter-efficient large-scale models (Gu et al., 2022). Specifically, each one-dimensional spectral kernel R_d is decomposed as:

$$R_d^{(\ell)} = U_{d,(k)}^{(\ell)} \left[V_{d,(k)}^{(\ell)} \right]^\top, \tag{6}$$

where $U_{d,(k)}^{(\ell)}$ and $V_{d,(k)}^{(\ell)}$ are reduced-rank decomposition matrices of rank k along dimension $d \in D$.

Given that $R_d^{(\ell)} \in \mathbb{R}^{H \times H \times M}$, where H is the hidden size for both input and output channels, and M the number of top Fourier modes retained along the spatial dimension, we explored applying low-rank factorization across two different tensor dimensions: between the input and output channels, and between the output channels and spatial dimensions. Our experiments demonstrated that the latter approach yields superior performance.

From a computational complexity perspective, the number of parameters for FFNO's $R_d^{(\ell)}$ scales as $O(LH^2MD)$, which is reduced to $O(LHk(H+M)D)$ after applying our reduced-rank decomposition. This substantial parameter reduction directly addresses the overparameterization problem by constraining the model's effective capacity without sacrificing expressiveness. The low-rank structure enforces a principled bottleneck that captures essential spectral patterns while eliminating redundant parameters, thereby preventing overfitting and enhancing generalization to unseen PDE configurations. By controlling the rank k, we can systematically regulate model complexity and achieve an optimal balance between representational power and generalization capability.

4. Numerical Experiments

We compared the proposed method against a variety of Fourier operator variants—FNO, AFNO, AM-FNO, Geo-FNO(Li et al., 2023a), U-FNO, WNO, and FFNO—across three

distinct fluid dynamics datasets: Darcy flow, the Navier-Stokes equation, and Kolmogorov flow. All experiments employ two-dimensional unit square domains with input and output functions discretized at 128×128 resolution.

Experimental Details. All models were trained for 1000 epochs using $N = 1000$ training samples and 200 test samples, with early stopping implemented to prevent overfitting. Training employed the AdamW optimizer (Loshchilov and Hutter, 2019) with an initial learning rate of 1×10^{-3} and weight decay of 1×10^{-4}, combined with cosine annealing learning rate scheduling (Loshchilov and Hutter, 2017). All models were implemented in PyTorch (Paszke, 2019) and trained on a single RTX 3080 GPU.

Mean Squared Error (MSE) served as the loss function during training, while evaluation metrics included both MSE and relative L2 loss, defined as:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N} \|\hat{\omega}_i - \omega_i\|_2^2\,; \text{ relative L2 loss} = \frac{1}{N}\sum_{i=1}^{N} \frac{\|\hat{\omega}_i - \omega_i\|_2}{\|\omega_i\|_2},$$

where $\|\cdot\|_2$ is the L2-norm, and $\hat{\omega}$ is the model's prediction of the corresponding ground truth ω. Prediction variability was assessed by calculating the standard deviation of the individual sample-wise MSE and relative L2 errors.

Model Configurations. To ensure fair comparison, all models were configured to project input data to a 32-channel latent space. Model-specific configurations were optimized through hyperparameter search: the FNO baseline and Geo-FNO used a mode size of 12, AFNO employed a patch size of 8 (creating 16×16 patches), AM-FNO utilized a width of 32 with 32 modes for MLP-based spectral layers, and FFNO was evaluated with a mode size of 32. For U-FNO and WNO, we adopted the default configurations from their respective source code: U-FNO with a width of 36 and mode size of 10, and WNO with discrete settings and a width of 40. The U-FNO model, originally designed for 3D problems, was adapted for our 2D experiments.

Our proposed R^2-FFNO applies reduced-rank factorization to the FFNO baseline with mode size 32. The optimal rank k is chosen from the set $\{1, 2, 4, 8, 16, 24, 28\}$. We employed a data augmentation strategy that involves the addition of Gaussian noise, aiming to mitigate high-frequency spectral bias. Empirical results suggest that this data augmentation indeed improved the model's performance. This improvement may be attributed to the inherent property of Gaussian noise, which possesses energy across a wide range of frequencies and introduces abrupt, fine-scale fluctuations. Its addition can thus make the model robust to these small, high-frequency perturbations. This strategy appears to effectively counteract the tendency of models to overfit low-frequency components, thereby potentially reducing spectral bias and improving generalization.

Our analysis proceeds in four stages. First, Section 4.2 provides empirical evidence for the low-rank nature of FFNO spectral kernels through normalized eigenvalue analysis. Second, Section 4.3 compares the proposed R^2-FFNO, using its optimal rank, against all baseline models. Section 4.4 presents a comprehensive sensitivity analysis, demonstrating the critical importance of rank selection for achieving optimal performance across different PDE systems. Finally, in Section 4.5, we conduct an ablation study to isolate the individual contributions of reduced-rank factorization and data augmentation to the overall performance.

4.1. Benchmark Datasets

4.1.1. Darcy Flow

The Darcy Flow dataset is generated from the steady-state Darcy equation, a fundamental elliptic PDE that describes fluid flow through porous media. The equation is defined as follows:

$$\begin{aligned} -\nabla \cdot (a(x)\nabla u(x)) &= f(x) \qquad x \in (0,1)^2 \\ u(x) &= 0 \qquad\quad x \in \partial(0,1)^2 \end{aligned}$$

Here, $u(x)$ represents the pressure field, $a(x)$ is the permeability field (coefficient), and $f(x)$ is a source term. The problem is posed on a unit square domain $(0,1)^2$ with homogeneous Dirichlet boundary conditions. The dataset is sourced from the Python package *Neural Operator* (Kovachki et al., 2024; Kossaifi et al., 2024).

4.1.2. Navier-Stokes Equations

The Navier-Stokes Equations dataset involves transient, incompressible fluid flow. These equations are a set of non-linear PDEs crucial for describing the motion of viscous fluids. The specific form used for this dataset is:

$$\begin{aligned} \partial_t \mathbf{u}(x,t) + \mathbf{u}(x,t) \cdot \nabla \mathbf{u}(x,t) &= -\nabla p(x,t) + \nu \Delta \mathbf{u}(x,t) && x \in (0,1)^2,\ t \in (0,T], \\ \nabla \cdot \mathbf{u}(x,t) &= 0 && x \in (0,1)^2,\ t \in (0,T], \\ \mathbf{u}(x,0) &= \mathbf{u}_0(x) && x \in (0,1)^2, \end{aligned}$$

where $\mathbf{u}(x,t)$ is the velocity vector at position x and time t, $p(x,t)$ is the pressure, and $\nu = \mu/\rho$ is the kinematic viscosity. Here, μ is dynamic viscosity, and ρ is the constant fluid density (taken to be unity). Δ denotes the **Laplacian operator**. The first equation represents momentum conservation, the second ensures incompressibility (divergence-free flow), and the third sets the initial velocity field. The dataset is also sourced from the Python package *Neural Operator*.

4.1.3. Kolmogorov Flow

Kolmogorov flow is a special case for the Navier-Stokes equations under a sinusoidal external forcing function in the first spatial direction (Chen, 2024):

$$\begin{aligned} \partial_t \mathbf{u}(x,t) + \mathbf{u}(x,t) \cdot \nabla \mathbf{u}(x,t) &= -\nabla p(x,t) + \nu \Delta \mathbf{u}(x,t) + \nu \mathbf{f}(x), \\ \nabla \cdot \mathbf{u}(x,t) &= 0, \\ \mathbf{u}(x,0) &= \mathbf{u}_0(x), \end{aligned}$$

where $\mathbf{f} = [a\sin(x_2), 0]^\top$ is an external forcing term. We generate the dataset using the python package *KolSol*[1], which implements the pseudospectral solver using a Fourier-Galerkin approach (Canuto et al., 1988).

1. https://github.com/MagriLab/KolSol

4.2. Low-Rank Characteristics of Spectral Kernels

The low-rank nature of spectral kernels is empirically demonstrated by analyzing their normalized eigenvalues. The normalized eigenvalue $\tilde{\lambda}_i$ is defined as:

$$\tilde{\lambda}_i = \frac{\lambda_i}{\sum_{j=1}^{K} \lambda_j},$$

where $\{\lambda_i\}_{i=1}^{K}$ are the eigenvalues in descending order and K is the full rank size.

The normalized eigenvalues of the FFNO spectral kernels (R_x and R_y) are presented by layer ($\ell = 0$ to 3) after 1000 training epochs, reflecting the decomposition across output channels and spatial dimensions. Figure 2 shows the analysis for the Navier-Stokes dataset, and Figure S1 in Supplemental Material A provides the results for Darcy Flow and Kolmogorov Flow. The observed rapid decay of the eigenvalues, particularly within the first few ranks, provides strong empirical evidence for the inherent low-rank characteristics of these spectral kernels, motivating the use of structured capacity constraints in R^2-FFNO.

While each plot exhibits distinct decay patterns, suggesting that optimal rank k might vary for individual spectral kernels, setting unique ranks would introduce excessive hyperparameter complexity. Therefore, for practical implementation and simplified model configuration, we adopt a uniform rank k across all spectral kernels within the model.

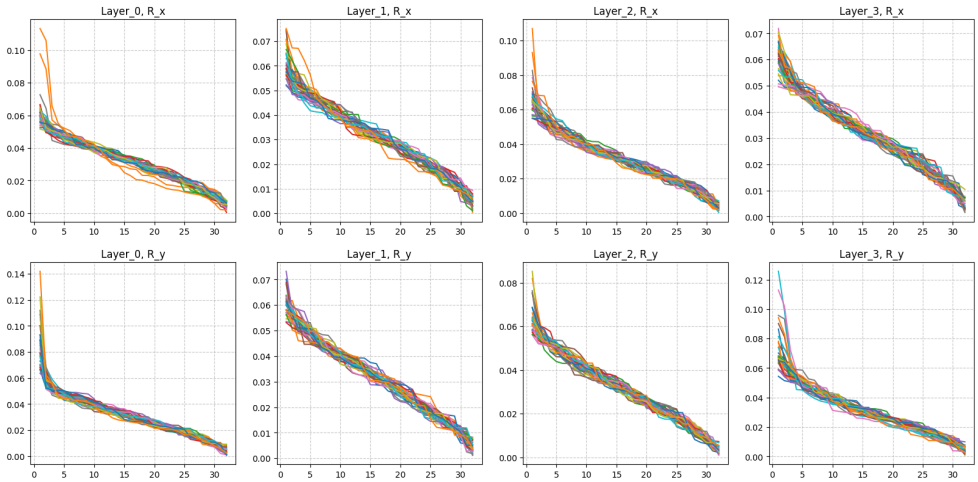


Figure 2: Normalized eigenvalues of the FFNO spectral kernels $R_x^{(\ell)}, R_y^{(\ell)}$ after 1000 training epochs on the Navier-Stokes dataset. The top row presents the eigenvalues for the x-dimension kernels $R_x^{(\ell)}$, while the bottom row for the y-dimension kernels $R_y^{(\ell)}$, across different layers ℓ.

4.3. Main Results

Table 1 presents a comprehensive comparison of MSE and relative L2 error across all datasets, with corresponding visual representations of model predictions shown in Figure 4.

Our proposed R^2-FFNO consistently demonstrates competitive and often superior performance across the evaluated benchmarks. On the Navier-Stokes dataset, R^2-FFNO achieved

optimal results with a substantial reduction of 46.5% on MSE and 27.6% on relative L2 compared to the baseline FFNO. For the Kolmogorov Flow dataset, R^2-FFNO outperformed the current state-of-the-art AM-FNO model with a 23.5% MSE reduction, representing a 31.6% improvement on MSE and an 18.5% reduction on relative L2 over the original FFNO baseline. While R^2-FFNO ranked second on the Darcy dataset, it still demonstrated substantial improvement with a 34.7% MSE reduction and an 18.1% relative L2 reduction over FFNO.

To confirm the significance of these gains, we conducted one-sided paired t-tests comparing the sample-wise MSE of R^2-FFNO to the FFNO baseline. The results showed statistically significant improvements across all three datasets, with p-values of 0.0256 for Darcy Flow, 6.83e-43 for Navier-Stokes, and 1.25e-7 for Kolmogorov Flow. As all these p-values are well below the 0.05 significance level, we can confidently conclude that our proposed method consistently and significantly outperforms the baseline. These consistent improvements across diverse fluid dynamics problems indicate the robustness and generalizability of the reduced-rank approach.

Beyond accuracy improvements, R^2-FFNO demonstrates enhanced stability across different experimental runs. The method exhibits the lowest SD of MSE and L2-loss for most test cases, with AM-FNO on the Darcy dataset being the sole exception. This pattern indicates that our model not only maintains or improves model accuracy but also contributes to greater robustness by minimizing prediction variability across runs.

	Darcy		**Navier-Stokes**		**Kolmogorov**	
	MSE	**relative L2**	**MSE**	**relative L2**	**MSE**	**relative L2**
FNO	19.60 ± 47.63	119.96 ± 77.64	34.16 ± 16.44	181.19 ± 17.97	3.82 ± 5.84	52.36 ± 49.48
AFNO	11.61 ± 40.61	82.40 ± 60.27	6.65 ± 6.37	73.67 ± 20.25	3.03 ± 4.08	48.63 ± 39.98
AM-FNO	**2.90 ± 5.99**	**46.17 ± 29.65**	6.64 ± 5.11	76.62 ± 14.09	1.70 ± 2.56	34.37 ± 33.69
Geo-FNO	17.35 ± 39.11	114.30 ± 68.84	32.68 ± 14.29	178.18 ± 15.32	2.23 ± 3.39	38.09 ± 39.02
U-FNO	21.00 ± 64.26	105.86 ± 87.20	5.17 ± 4.39	66.47 ± 15.77	2.18 ± 3.15	40.68 ± 34.49
WNO	15.77 ± 36.21	110.57 ± 63.91	28.28 ± 15.07	163.23 ± 19.15	2.73 ± 4.10	43.76 ± 41.38
FFNO	4.94 ± 16.64	59.25 ± 39.17	4.30 ± 3.71	60.76 ± 13.17	1.92 ± 2.95	36.35 ± 36.07
R^2FFNO	3.25 ± 9.24	48.49 ± 29.53	**2.31 ± 2.25**	**43.96 ± 10.65**	**1.31 ± 1.99**	**29.56 ± 30.39**

Table 1: Comparison of the mean and standard deviation of MSE and relative L2 error for different models on the Darcy, Navier-Stokes, and Kolmogorov datasets. All values are in units of 10^{-3}

4.4. Rank Sensitivity

To investigate the sensitivity of our proposed R^2-FFNO model to low-rank factorization, we conducted a parameter study on rank size k. With a full mode size of 32, we evaluated model performance across various rank values: $k \in \{1, 2, 4, 8, 16, 24, 28\}$. The results, detailing both MSE and relative L2 loss, are summarized in Table 2 and visualized in Figure 3.

The analysis reveals that the rank size has a significant impact on model performance, with optimal values varying across datasets. For the Darcy Flow and Navier-Stokes Equation datasets, the optimal rank achieving the lowest errors is $k = 16$. In contrast, the Kolmogorov Flow dataset achieves the best performance with a higher rank of $k = 24$. This

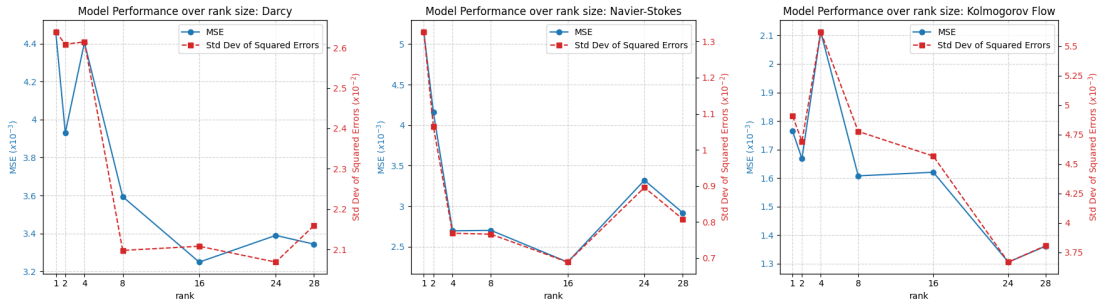
variation indicates that the ideal reduced-rank approximation depends on the underlying complexity and characteristics of each specific PDE problem.

Across all datasets, we observe a consistent pattern: MSE and relative L2 loss initially decrease as rank increases, followed by either saturation or a slight increase, suggesting degraded performance beyond the optimal rank. The SD generally mirrors this trend, indicating that optimal rank selection contributes not only to improved accuracy but also to more consistent and robust predictions. However, the optimal rank for the mean error does not always align with the optimal rank for the minimum variability. For example,in the Darcy Flow analysis, the lowest mean MSE is observed at rank $k = 16$, whereas the lowest SD of MSE is at $k = 8$. This suggests a nuanced trade-off between maximizing predictive accuracy and maximizing prediction consistency for certain PDE systems.

These results highlight the existence of an optimal trade-off point where model complexity, as defined by rank, maximizes predictive performance and robustness. The varied optimal ranks across datasets further confirm that carefully tuning the rank size to the specific characteristics of the problem is crucial to achieving both high performance and robust generalization.

	Darcy		**Navier-Stokes**		**Kolmogorov**	
	MSE	**relative L2**	**MSE**	**relative L2**	**MSE**	**relative L2**
rank=1	4.36 ± 10.92	57.33 ± 34.79	4.99 ± 3.98	66.12 ± 13.12	1.77 ± 2.62	36.29 ± 32.56
rank=2	3.86 ± 12.10	53.89 ± 34.13	4.01 ± 3.29	59.02 ± 12.19	1.68 ± 2.51	35.06 ± 32.72
rank=4	4.35 ± 10.90	55.22 ± 34.68	2.61 ± 2.30	47.12 ± 10.56	2.11 ± 3.23	37.70 ± 38.90
rank=8	3.56 ± 8.91	50.36 ± 33.57	2.62 ± 2.49	46.88 ± 11.27	1.60 ± 2.48	32.64 ± 33.51
rank=16	**3.21 ± 9.24**	**47.98 ± 29.66**	**2.25 ± 2.21**	**43.30 ± 10.64**	1.61 ± 2.63	32.09 ± 35.08
rank=24	3.37 ± 8.93	48.83 ± 32.78	3.27 ± 3.00	52.44 ± 12.48	**1.31 ± 1.97**	**29.81 ± 30.11**
rank=28	3.31 ± 9.32	47.33 ± 31.25	2.87 ± 2.76	49.06 ± 11.82	1.35 ± 2.06	30.09 ± 30.87

Table 2: Comparison of the mean and standard deviation of MSE and relative L2 error for different rank sizes on the Darcy, Navier-Stokes, and Kolmogorov datasets.



(a) Darcy Flow (b) Navier-Stokes Equation (c) Kolmogorov Flow

Figure 3: Evaluation of R^2-FFNO model performance with varying low-rank factorization sizes (k) across three datasets: (a) Darcy Flow, (b) Navier-Stokes Equation, and (c) Kolmogorov Flow.

4.5. Ablation Study: the Contributions of Reduced-Rank Factorization and Data Augmentation

We conducted a comprehensive ablation study to isolate the individual contributions of our two primary components: reduced-rank factorization and data augmentation. The results in Table 3 clearly show that both reduced-rank factorization alone (R^2-FFNO) and data augmentation individually (FFNO(aug)) improve performance over the baseline FFNO.

Both reduced-rank factorization alone (R^2-FFNO) and data augmentation individually (FFNO(aug)) improve performance over the baseline FFNO. For the Darcy Flow dataset, reduced-rank factorization achieves a 12.8% reduction in MSE and a 4.6% reduction in relative L2, while data augmentation provides a 6.7% improvement in MSE and a 7.8% reduction in relative L2. In the Navier-Stokes dataset, data augmentation shows more substantial gains across the board with a 28.4% reduction in MSE and a 16.2% reduction in relative L2, while reduced-rank factorization maintains similar performance. For Kolmogorov Flow, both components contribute modestly but consistently. Reduced-rank factorization reduces MSE by 3.1% and relative L2 by 2.2%, while data augmentation reduces MSE by 1.6% and relative L2 by 2.2%.

Our proposed complete model R^2-FFNO(aug), which combines both techniques, consistently achieves the best results across all three datasets. The combined approach yields substantial overall improvements: 34.2% MSE reduction and 18.1% relative L2 reduction for Darcy Flow; 46.3% MSE reduction and 27.6% relative L2 reduction for Navier-Stokes; and 31.8% MSE reduction and 18.5% relative L2 reduction for Kolmogorov Flow. Importantly, these substantial final improvements exceed the sum of individual contributions, confirming the synergistic effects between reduced-rank factorization and data augmentation. This confirms that both components are crucial to the overall effectiveness of our method and that their effects are synergistic.

	Darcy		**Navier-Stokes**		**Kolmogorov**	
	MSE	**relative L2**	**MSE**	**relative L2**	**MSE**	**relative L2**
FFNO	4.94 ± 16.64	59.25 ± 39.17	4.30 ± 3.71	60.76 ± 13.17	1.92 ± 2.95	36.35 ± 36.07
FFNO(aug)	4.61 ± 14.96	54.64 ± 38.90	3.08 ± 2.76	51.02 ± 11.83	1.89 ± 2.93	35.48 ± 36.87
R^2FFNO	4.31 ± 11.35	56.51 ± 33.24	4.26 ± 3.62	60.43 ± 13.10	1.86 ± 2.83	35.49 ± 35.78
R^2FFNO(aug)	**3.25 ± 9.24**	**48.49 ± 29.53**	**2.31 ± 2.25**	**43.96 ± 10.65**	**1.31 ± 1.99**	**29.56 ± 30.39**

Table 3: Ablation study isolating the performance effects of data augmentation and reduced-rank factorization. This table compares the performance of four model variants: the baseline FFNO, FFNO with data augmentation (FFNO(aug)), R^2-FFNO with reducedrank factorization only, and the full R^2-FFNO(aug) model combining both strategies.

5. Conclusion and Discussion

This work introduces the R^2-FFNO method to address overparameterization in FFNO architectures through systematic reduced-rank factorization of spectral kernels. To further enhance its robustness and mitigate high-frequency spectral bias, we also employed a data augmentation strategy involving the addition of high-frequency data variations via Gaussian noise. Comprehensive experimental evaluation demonstrates that R^2-FFNO achieves

competitive or superior performance compared to state-of-the-art models, with reduced prediction errors and enhanced stability. Our analysis highlights the critical importance of careful rank selection, as it directly impacts the balance between computational efficiency and predictive accuracy. Notably, we observe performance saturation after reaching the optimal rank, with degraded performance beyond this threshold, suggesting the existence of an optimal complexity-accuracy trade-off.

Future research directions include extending R^2-FFNO to irregular domains, such as structured meshes or unstructured point clouds, potentially through the application of coordinate mapping techniques or integration with approaches like GINO (Li et al., 2023b) and Geo-FNO. Combining R^2-FFNO with Physics-Informed Neural Networks (PINNs) (Raissi et al., 2019) presents an intriguing direction, as the operator learning capabilities of R^2-FFNO combined with the physics-constraining properties of PINNs could yield more robust and data-efficient PDE solvers. Furthermore, investigating the integration of R^2-FFNO with WNO, which captures multiscale spatial and frequency information, could address challenges associated with high-frequency modeling and enhance multi-scale representation capabilities.

References

Kamyar Azizzadenesheli, Nikola Kovachki, Zongyi Li, Miguel Liu-Schiaffini, Jean Kossaifi, and Anima Anandkumar. Neural operators for accelerating scientific simulations and design. *Nature Reviews Physics*, 2024.

Claudio Canuto, M Yousuff Hussaini, Alfio Quarteroni, and Thomas A Zang. Spectral methods inffluid dynamics, 1988.

Zhi-Min Chen. Enhanced and unenhanced dampings of the kolmogorov flow. *Journal of Differential Equations*, 2024.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations (ICLR)*, 2022.

John Guibas, Morteza Mardani, Zongyi Li, Andrew Tao, Anima Anandkumar, and Bryan Catanzaro. Adaptive fourier neural operators: Efficient token mixers for transformers. In *International Conference on Learning Representations (ICLR)*, 2022.

Isom H. Herron and Michael R. Foster. *Partial Differential Equations in Fluid Dynamics*. Cambridge University Press, 2008.

Keith Julien and Mike Watson. Efficient multi-dimensional solution of pdes using chebyshev spectral methods. *Journal of Computational Physics*, 2009.

Jean Kossaifi, Nikola Kovachki, Zongyi Li, David Pitt, Miguel Liu-Schiaffini, Robert Joseph George, Boris Bonev, Kamyar Azizzadenesheli, Julius Berner, and Anima Anandkumar. A library for learning neural operators, 2024.

Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: learning maps between function spaces with applications to pdes. *The Journal of Machine Learning Research*, 2024.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Multipole graph neural operator for parametric partial differential equations. In *International Conference on Neural Information Processing Systems (NeurIPS)*, 2020a.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Graph kernel network for partial differential equations. In *International Conference on Learning Representations (ICLR) Workshop*, 2020b.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations (ICLR)*, 2021.

Zongyi Li, Daniel Zhengyu Huang, Burigede Liu, and Anima Anandkumar. Fourier neural operator with learned deformations for pdes on general geometries. *Journal of Machine Learning Research*, 24(388):1–26, 2023a.

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. Geometry-informed neural operator for large-scale 3d PDEs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023b.

Chaoyu Liu, Davide Murari, Lihao Liu, Yangming Li, Chris Budd, and Carola-Bibiane Schönlieb. Enhancing fourier neural operators with local spatial features. *arXiv:2503.17797*, 2025.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations (ICLR)*, 2017.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 2021.

Andreas Öchsner. *Partial Differential Equations of Classical Structural Members*. Springer Cham, 2020.

A Paszke. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.

Shaoxiang Qin, Fuyuan Lyu, Wenhui Peng, Dingyang Geng, Ju Wang, Naiping Gao, Xue Liu, and Liangzhu Leon Wang. Toward a better understanding of fourier neural operators: Analysis and improvement from a spectral perspective. *arXiv:2404.07200*, 2024.

Md Ashiqur Rahman, Zachary E. Ross, and Kamyar Azizzadenesheli. U-no: U-shaped neural operators. *Transactions on Machine Learning Research*, 2024.

Maziar Raissi, 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*, 378:686–707, 2019.

Bogdan Raonić, Roberto Molinaro, Tim De Ryck, Tobias Rohner, Francesca Bartolucci, Rima Alaifari, Siddhartha Mishra, and Emmanuel de Bézenac. Convolutional neural operators for robust and accurate learning of pdes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Gregory C Reinsel, Raja P Velu, and Kun Chen. *Multivariate Reduced-Rank Regression: Theory, Methods and Applications*, volume 225. Springer Nature, 2 edition, 2022. ISBN 978-1-0716-2793-8. doi: https://doi.org/10.1007/978-1-0716-2793-8.

Thomas Stocker. *Introduction to Climate Modelling*. Springer Berlin, Heidelberg, 2011.

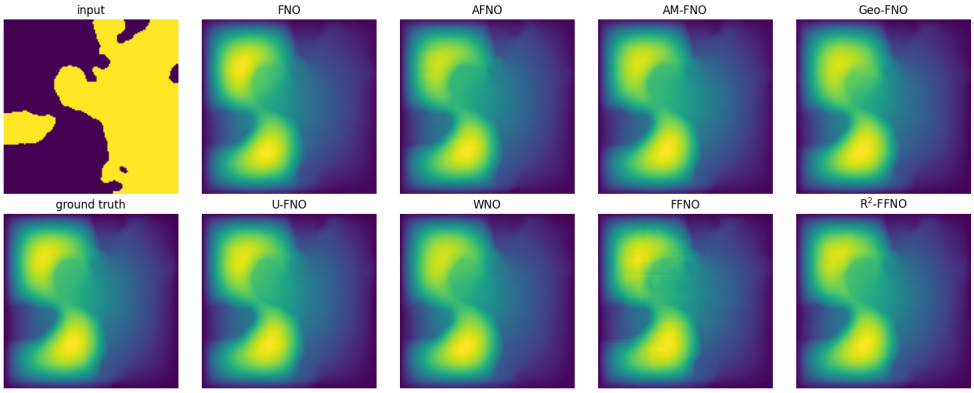
Alasdair Tran, Alexander Mathews, Lexing Xie, and Cheng Soon Ong. Factorized fourier neural operators. In *International Conference on Learning Representations (ICLR)*, 2023.

Tapas Tripura and Souvik Chakraborty. Wavelet neural operator for solving parametric partial differential equations in computational mechanics problems. *Computer Methods in Applied Mechanics and Engineering*, 2023.

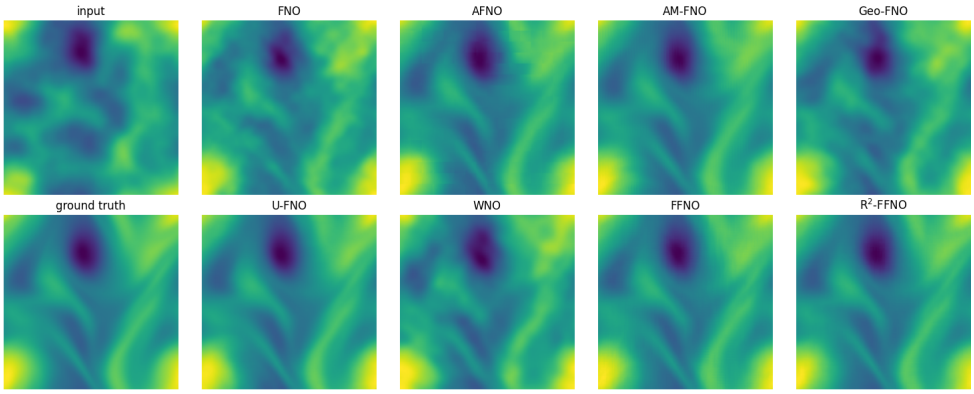
Gege Wen, Zongyi Li, Kamyar Azizzadenesheli, Anima Anandkumar, and Sally M. Benson. U-fno—an enhanced fourier neural operator-based deep-learning model for multiphase flow. *Advances in Water Resources*, 2022.

Zipeng Xiao, Siqi Kou, Hao Zhongkai, Bokai Lin, and Zhijie Deng. Amortized fourier neural operators. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

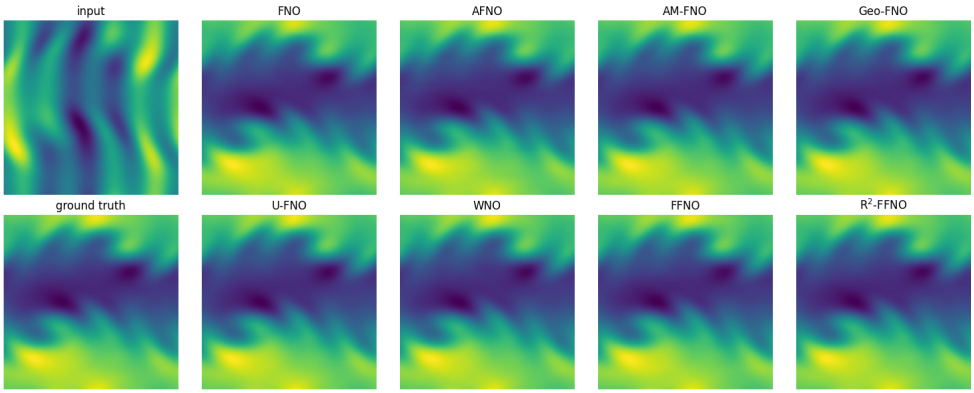
Appendix A. Prediction of Models



(a) Darcy Flow



(b) Navier-Stokes Equation



(c) Kolmogorov Flow

Figure 4: Visual comparison of predictions across different models for a representative sample from each dataset. For each dataset, the figure displays the input field, the corresponding ground truth solution, and the predictions generated by various models.