OPERATOR LEARNING WITH FOURIER NEURAL OPERATORS: THEORETICAL GUARANTEES AND APPLICATIONS

Lappeenranta-Lahti University of Technology LUT

Master's Program in Computational Engineering, Master's Thesis

2025

Eliane Mopo Youmssi

Examiners: Professor Tapio Helin
Post Doc. Abhishake Abhishake

ABSTRACT

Lappeenranta-Lahti University of Technology LUT
School of Engineering Science
Computational Engineering

Eliane Mopo Youmssi

Operator learning with Fourier Neural Operators: Theoretical guarantees and applications

Master's thesis
2025
58 pages, 12 figures, 3 tables
Examiners: Professor Tapio Helin and Post Doc. Abhishake Abhishake

Keywords: Fourier neural operator, Darcy flow, universal approximation theorem, discretization-invariance

This thesis investigates the use of Fourier Neural Operators (FNOs) to simulate steady fluid movement through porous materials, with a specific emphasis on the two-dimensional Darcy flow problem. The aim is to assess the effectiveness of FNOs in approximating solutions to partial differential equations (PDEs) purely through data, without needing explicit understanding of the underlying equations when making predictions. The study considers theoretical elements such as the universal approximation theorem and the discretization-invariance trait of FNOs, which bolster their ability to adapt to various grid sizes. To examine real-world performance, FNOs were trained on synthetic data produced by traditional numerical methods and evaluated using input fields with distinct structural characteristics, such as modified and transformed Gaussian random fields. The findings reveal that FNOs excel with smooth and orderly input fields, achieving low relative errors and little bias in predictions. However, when faced with irregular or discontinuous inputs, accuracy diminishes, showing sensitivity to the input's structure. This research underscores the capability and resilience of FNOs for solving PDEs and highlights their promise in computational physics and engineering, where efficient and precise modeling is crucial.

ACKNOWLEDGEMENTS

This thesis represents the end of a demanding yet rewarding journey. Working on FNOs has not been an easy task, but I am proud of what I have achieved and thankful for the chance to explore such a cutting-edge research area.

I owe deep gratitude to my supervisor and mentor, Professor Tapio Helin, for his unwavering support, insightful advice, and for believing in my abilities even when I was uncertain. His concern for my well-being, especially during our many meetings in the chilly Finnish winter, taught me that kindness and mentorship go hand in hand. I am also incredibly grateful to my family for their endless love and support, particularly my parents, whose regular phone calls offered me strength and comfort. I would like to acknowledge everyone who contributed to this project in any way, especially the researchers and PhD students in the Computational Engineering Department at LUT University.

Last, but not least, I am sincerely thankful to the African Institute for Mathematical Sciences (AIMS) Rwanda and LUT University for providing me with the opportunity to pursue this Master's degree through their generous scholarships.

Lappeenranta, May 8, 2025

Eliane Mopo Youmssi

LIST OF ABBREVIATIONS

D	A domain in $\mathbb{R}^d$
$\mathcal{N}$	A neural operator
σ	Activation function
$K, K', \widetilde{K}$	Compact sets
$\mathcal{G}$	Continuous operator
$\mathcal{F}_N, \mathcal{F}_N^{-1}$	Discrete Fourier and inverse discrete Fourier transforms
FFT	Fast Fourier Transform
$\mathcal{F}, \mathcal{F}^{-1}$	Fourier and inverse Fourier transforms
$\widehat{v}_k$	Fourier coefficients of a function v
FNO	Fourier Neural Operator
$\mathcal{J}_N$	Grid point indices: $\mathcal{J}_N := \{0, \cdots, 2N\}^d$
H_N^s	$H_N^s \subset H^s$ trigonometric polynomials of degree $\leq N$
$\Im$	Imaginary part of a complex number
$\mathcal{A}(D; \mathbb{R}^{d_a})$	Input function space
$\mathcal{K}$	Integral kernel operator
$\mathcal{P}$	Lifting operator
$⨍_{\mathbb{T}^d}$	Mean integral over the torus $\mathbb{T}^d$: $⨍_{\mathbb{T}^d} f(x)\,dx = \frac{1}{\lvert\mathbb{T}^d\rvert}\int_{\mathbb{T}^d} f(x)\,dx$
MSE	Mean Square Error
NN	Neural Network
NO	Neural Operator
N	Non-negative integer
d_a, d_u, d_v	Number of components of input, output and lifting
$\mathcal{U}(D; \mathbb{R}^{d_u})$	Output function space
P_N	Orthogonal Fourier projection operator
$\Theta \subset \mathbb{R}^p$	Parameter space
κ_θ	Parameterized kernel function
PDE	Partial Differential Equation
$\mathbb{T}^d$	Periodic torus identified with $[0, 2\pi]^d$
$\mathcal{Q}$	Projector operator
$\Re$	Real part of a complex number
$\mathcal{G}\vert_{\widetilde{K}}$	Restriction of $\mathcal{G}$ to $\widetilde{K}$
$\mathcal{K}_N$	Set of Fourier wavenumbers: $\mathcal{K}_N := \{k \in \mathbb{Z}^d \mid \lvert k\rvert_\infty \leq N\}$
H^s	Sobolev space of order s
d	Spatial dimension of the domain
$\mathcal{L}_t$	t-th neural operator layer

CONTENTS

1 INTRODUCTION **6**
1.1 Background 6
1.2 Objectives and delimitations 7
1.3 Structure of the thesis 8

2 LITERATURE REVIEW **9**

3 THEORETICAL FOUNDATIONS OF FNOs **12**
3.1 Preliminary Mathematical Concepts 12
3.2 General NOs 14
3.3 Fourier Neural Operators (FNOs) 20
3.4 Universal Approximation Theorem for FNOs 22

4 EXPERIMENTS **36**
4.1 Data 37
4.2 Evaluation Criteria 38
4.3 Description of Experiments 39
4.4 Results 40

5 DISCUSSION **52**
5.1 Current study 52
5.2 Future work 53

6 CONCLUSION **55**

REFERENCES **56**

1 INTRODUCTION

1.1 Background

Partial differential equations (PDEs) play a central role in modeling physical, biological, and engineering systems. They arise in a wide variety of applications such as fluid dynamics, material science, weather prediction, and medical imaging [1, 2]. Solving these equations accurately and efficiently is a longstanding challenge, especially when the systems are high-dimensional or time-dependent.

Traditionally, numerical techniques like the finite difference, finite volume, and finite element methods have been used to approximate PDE solutions [1]. However, these methods can be computationally expensive when applied repeatedly across varying input parameters or domain geometries, which is common in tasks like real-time simulation, optimization, and uncertainty quantification [3, 4].

In recent years, a new paradigm has emerged: *operator learning*. Instead of approximating a single solution, operator learning seeks to approximate the *solution map* that takes the input of a PDE (e.g., a coefficient function or boundary condition) to its corresponding solution [5, 6]. This approach enables fast and flexible inference once the operator has been learned from data.

A major breakthrough in this field was the introduction of Fourier Neural Operators (FNOs) by [7]. FNOs use the Fast Fourier Transform (FFT) to learn mappings between infinite-dimensional function spaces. By operating in the Fourier domain, they are able to efficiently capture global spatial features and generalize across varying PDE inputs. Moreover, they exhibit a desirable property known as *discretization-invariance*, allowing them to transfer across different resolution grids [8]. These strengths have made FNOs a key focus in the rapidly developing field of neural operators [9].

This thesis focuses on the mathematical underpinnings and practical application of FNOs for solving PDEs, with a particular emphasis on the two-dimensional Darcy flow problem. The aim is to understand how FNOs approximate solution operators, what theoretical guarantees support their design, and how they can be effectively implemented for real-world problems.

To better illustrate the type of complex phenomena that FNOs are designed to handle,

consider the visualization in Figure 1. Although this figure comes from the study of the Navier-Stokes equations, it highlights similar challenges faced when working with PDEs such as the Darcy flow problem, especially in terms of solving for fluid dynamics in complex domains.

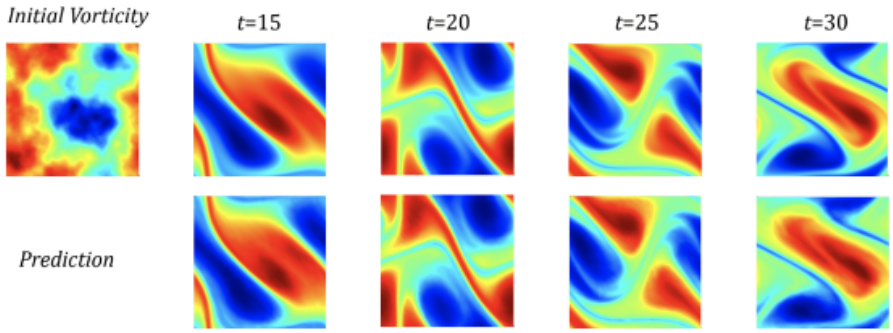


Figure 1. Snapshots of a Navier-Stokes solution at different time steps, illustrating time-dependent fluid dynamics, [7].

1.2 Objectives and delimitations

The main aim of this thesis is to investigate the application of FNOs to solving steady-state PDEs, with a focus on the Darcy flow problem. This work explores both the theoretical properties and practical implementation of FNOs in the context of this specific class of PDEs.

The key objectives of this thesis are as follows:

1. To investigate how FNOs can be applied to solve the Darcy flow problem, specifically in the context of modeling steady-state fluid flow in porous media.
2. To analyze the theoretical guarantees of FNOs, focusing on their discretization-invariance property and the universal approximation theorem, which ensures the ability of FNOs to approximate continuous operators between function spaces.
3. To evaluate the performance of FNOs by comparing the predicted solutions obtained through FNOs to the ground truth solutions, computed using traditional numerical methods such as finite element methods (FEM) or finite difference methods (FDM).

4. To assess the error, bias, and standard deviation of the predicted solutions in comparison to the ground truth data, providing insights into the accuracy and reliability of FNO-based methods for solving Darcy flow problems.

While this thesis primarily focuses on the application of FNOs to the Darcy flow problem, several related topics are intentionally excluded. Specifically, the study does not consider time-dependent problems or transient flow, as the focus is on steady-state equations. Additionally, the work is limited to the Darcy flow problem and does not extend to other types of PDEs such as those arising from electromagnetism or more complex fluid dynamics. Advanced topics such as control theory, uncertainty quantification, or the application of FNOs to multi-phase flows are also outside the scope of this thesis. Furthermore, the thesis does not delve into the development or modification of the FNO architecture itself, but instead focuses on its application and theoretical understanding.

1.3 Structure of the thesis

The thesis is structured as follows: Chapter 1 provides an introduction to the research problem, the objectives of the thesis, and an overview of the structure. Chapter 2 reviews the literature on operator learning, with a particular emphasis on Fourier Neural Operators (FNOs) and related models. Chapter 3 discusses the theoretical foundations of FNOs, beginning with key preliminary concepts such as the Fourier transform and the theory of Banach spaces, followed by a detailed exploration of FNOs' discretization-invariance property and the universal approximation theorem. Chapter 4 presents the practical implementation of FNOs for solving the two-dimensional Darcy flow problem. It covers the data generation process, the evaluation criteria adopted, the experimental setup, and a detailed presentation of the results. Chapter 5 provides a critical examination of the results in light of the research questions introduced earlier. It clarifies the contributions of the current study, identifies its limitations, and proposes meaningful directions for future work. Finally, Chapter 6 concludes the thesis with a summary of findings presented earlier.

2 LITERATURE REVIEW

Operator learning has a rich history starting with the seminal work of Tianping Chen and Hong Chen [5], who first proposed operator approximation using neural networks. This framework adopted a branch net for inputs and a trunk net for outputs to learn mappings between infinite-dimensional common input and output spaces, and the results showed a universal approximation property for nonlinear operators. While this was a breakthrough, the method was computationally expensive and did not scale well for high dimensional problems, so it was not practically applicable for larger problems.

After these initial developments, Zongyi Li et al. [7] proposed Fourier neural operators that represented a large step forward in operator learning. Utilizing the FFT, FNOs parameterized the kernel of the integral operator in the frequency domain so that operators between function spaces can be efficiently approximated. Remarkable success of this approach was shown on a family of PDEs including Darcy flow and Navier-Stokes equations. Nonetheless, the authors noted that FNOs struggled to generalize to non-periodic or very multiscale PDEs, and tended to overfit in some cases, especially when the training data were scarce and poorly regularized.

To address these challenges, authors in [10] proposed the multi-scale graph neural operator, which utilized a graph structure combined with multipole expansions. This innovation enhanced the representation and approximation capabilities of neural operators, enabling them to handle more intricate PDEs. While this method improved the generalization of neural operators to a broader set of problems, it did not fully resolve issues related to memory-intensive computations particularly for large-scale systems requiring high memory and processing power. Parallel to these developments, [6] introduced DeepONets, which employed a separate branch-trunk architecture to discretize operators, offering an alternative to FNOs for problems with irregular geometries.

Recognizing the need for stronger methods, Gaurav G. et al. [11] introduced a multiwavelet-based neural operator. This innovative approach used multiwavelet transforms to approximate the kernel, which not only provided a more compact representation but also helped reduce overfitting. While this method showed better performance than FNOs in certain applications, like steady-state problems, it faced challenges with time-dependent PDEs. This pointed out the ongoing demand for techniques that can effectively model dynamic systems, where the timing of changes is crucial.

In 2022, [12] enhanced operator learning models by incorporating advanced techniques

for kernel approximation. Their work significantly boosted the accuracy and robustness of neural operators, particularly for problems involving complex shapes and irregular domains. However, they acknowledged that achieving real-time predictions for high-dimensional PDEs is still a challenge, mainly due to the hefty computational costs associated with accurately approximating solutions in real-time situations.

The following year, [13] addressed the challenge of generalizing to high-dimensional PDE systems. They introduced adaptive mesh refinement techniques, which allowed neural operators to better capture the intricate details in the solutions of PDEs. Although this was a significant step forward, it also came with increased computational demands, emphasizing the ongoing trade-off between accuracy and computational efficiency that continues to affect many modern neural operator methods.

To provide a comprehensive overview of the field, [14] conducted an extensive review of operator learning models, which included FNOs, DeepONets, and multiwavelet-based operators. Their analysis highlighted the strengths and weaknesses of these models and provided valuable insights into the challenges faced by current methods, such as their inability to efficiently handle certain types of complex, high-dimensional problems. Their work also outlined potential avenues for future research in operator learning, particularly with respect to enhancing generalization, scalability, and efficiency.

In a related effort, [15] made important theoretical contributions by establishing a universal approximation theorem for neural operators. This work provided a solid theoretical foundation for operator learning, demonstrating that neural operators can indeed approximate nonlinear mappings between function spaces. While their findings indicate a promising future for neural operators in areas like system identification, real-time predictions, and dynamic simulations, [15] also cautioned that the practical implications of their results required further exploration to fully realize their potential in industrial and engineering applications.

Tackling the complexities of dynamic systems, [16] presented a neural operator architecture specifically tailored for time-dependent PDEs. Their approach integrated temporal dynamics, allowing for continuous-in-time predictions that broadened the use of neural operators in time-varying systems. However, they noted that scaling to extremely large domains posed a significant challenge, and more innovations would be necessary to make real-time, large-scale simulations feasible.

Most recently, [17] introduced a fascinating concept called pseudo-differential neural operators (PDNOs), which are an exciting new twist on FNOs. By utilizing pseudo-

differential integral operators (PDIOs) and symbol networks, PDNOs have taken FNOs to the next level, achieving impressive accuracy on tough challenges like Darcy flow and the Navier-Stokes equations. The authors of [17] tackled many of the common pitfalls seen in earlier methods, such as overfitting and issues with generalization, setting a fresh standard for operator learning models. Their research showcases how PDNOs can effectively manage complex, high-dimensional problems with more precision and efficiency than previous neural operator models.

The journey of developing operator learning models has been remarkable, with each breakthrough building on the achievements and insights gained from earlier techniques. From the pioneering work of [5] to the cutting-edge strategies of [17], neural operators have emerged as a robust tool for approximating solutions to intricate PDEs. The advancements mentioned above underscore the transformative potential of neural operators in addressing challenging mathematical problems, especially in areas like fluid dynamics, system identification, and real-time simulations. Yet, hurdles related to scalability, generalization, and efficiency still exist, underscoring the need for innovative solutions to fully unlock the capabilities of neural operators.

3 THEORETICAL FOUNDATIONS OF FNOs

This chapter begins by recalling fundamental mathematical concepts that serve as preliminaries for the rest of the work. It then proceeds to develop the analytical framework and main results that support the core of this study.

3.1 Preliminary Mathematical Concepts

We begin by providing an overview of key functional analysis concepts essential for establishing the main results of this study. The section relies on sources like [18–22].

To facilitate the understanding of the results that follow, we briefly recall the definitions of some functional spaces that are used throughout the main proofs of this work.

Definition 3.1. *A* Hilbert space H *is a complete normed space where the norm arises from an inner product* $\langle \cdot, \cdot \rangle : H \times H \to \mathbb{K}$*, defined by* $\|x\| = \sqrt{\langle x, x \rangle}$ *for all* $x \in H$*.*

Hilbert spaces, such as ℓ^2 and $L^2(\Omega)$, are Banach spaces with additional geometric structure, including orthonormal bases and the validity of the Pythagorean theorem, making them central to both theoretical and applied mathematics.

Definition 3.2. *For* $1 \leq p < \infty$*, the sequence space* ℓ^p *consists of all sequences* $x = (x_n)_{n \in \mathbb{N}}$ *such that*

$$\|x\|_{\ell^p} = \left(\sum_{n=1}^{\infty} |x_n|^p \right)^{1/p} < \infty.$$

For $p = \infty$*, the space* ℓ^∞ *consists of all bounded sequences with norm:*

$$\|x\|_{\ell^\infty} = \sup_{n \in \mathbb{N}} |x_n|.$$

These spaces are crucial in the study of functional analysis and sequence convergence.

Definition 3.3. *The space* $L^p(\Omega)$*, where* $\Omega \subset \mathbb{R}^d$ *is a measurable set, is defined as the set of measurable functions* $f : \Omega \to \mathbb{R}$ *(or* $\mathbb{C}$*) such that*

$$\|f\|_{L^p} = \left(\int_{\Omega} |f(x)|^p \, dx \right)^{1/p} < \infty, \quad 1 \leq p < \infty.$$

For $p = \infty$, the norm is defined as $\|f\|_{L^\infty} = \operatorname{ess\,sup}_{x\in\Omega}|f(x)|$.

These spaces are widely used in the analysis of PDEs, particularly in the study of weak solutions.

Definition 3.4. *Let $\Omega \subset \mathbb{R}^d$ be an open set. The* Sobolev space $W^{s,p}(\Omega)$*, for $s \in \mathbb{N}$ and $1 \leq p \leq \infty$, consists of functions $u \in L^p(\Omega)$ whose weak derivatives $D^\alpha u$ exist and belong to $L^p(\Omega)$ for all multi-indices α with $|\alpha| \leq k$. The norm on $W^{s,p}(\Omega)$ is defined as*

$$\|u\|_{W^{s,p}(\Omega)} = \left(\sum_{|\alpha|\leq s} \|D^\alpha u\|^p_{L^p(\Omega)} \right)^{1/p}.$$

Sobolev spaces are essential in the theory of weak solutions to PDEs, where they provide a natural setting for variational methods.

Remark 1. *In the special case $p = 2$, we write $W^s(\Omega)$ instead of $W^{s,2}(\Omega)$. The spaces $W^s(\Omega)$ are Hilbert spaces with the inner product*

$$\langle u, v\rangle_{W^s(\Omega)} = \sum_{|\alpha|\leq s} \int_\Omega D^\alpha u(x) D^\alpha v(x)\, dx.$$

This case is particularly important in applications due to the rich structure of Hilbert spaces. In these spaces, the geometry is deeply connected to the energy functional of the system, making them pivotal in many applied fields like fluid dynamics and elasticity.

We next revisit the notion of compact sets and highlight a few key results that play a crucial role in the proofs developed later in this work.

Definition 3.5. *Let $S \subset H$ be a subset of a Hilbert space H. An* open cover *of S is a collection of open sets $\{U_i\}_{i\in I}$ such that $S \subseteq \bigcup_{i\in I} U_i$, meaning that every point in S belongs to at least one of the sets U_i.*

Definition 3.6. *Let H be a Hilbert space. A subset $K \subset H$ is said to be* compact *if, for every open cover $\{U_i\}_{i\in I}$ of K, there exists a finite sub-cover $\{U_{i_1}, U_{i_2}, \ldots, U_{i_n}\}$ such that $K \subseteq \bigcup_{k=1}^n U_{i_k}$. In other words, every open cover of K has a finite sub-cover.*

Some properties of compact sets include:

Proposition 1. *Any finite reunion of compact subsets of a Hilbert H is compact.*

Proof. To prove this, it suffices to show that the proposition is true for two compact subsets K_1 and K_2 of H. So, let $\{U_i\}_{i\in I}$ be an open cover of $K_1 \cup K_2$. By the definition of compactness, there exist finite sub-covers $\{U_{i_1}, U_{i_2}, \ldots, U_{i_m}\}$ covering K_1 and $\{U_{j_1}, U_{j_2}, \ldots, U_{j_k}\}$ covering K_2. Thus, we can combine the sub-covers of K_1 and K_2 into a finite sub-cover of $K_1 \cup K_2$. Therefore, $K_1 \cup K_2$ is covered by a finite sub-cover, proving that $K_1 \cup K_2$ is compact and concludes the proof. □

Proposition 2. *Let $\mathcal{G} : H_1 \to H_2$ be a continuous linear operator, and let $K \subset H_1$ be a compact set. Then the image $\mathcal{G}(K)$ is also compact in H_2.*

Proof. Let $\{U_i\}_{i\in I}$ be an open cover of $\mathcal{G}(K)$. Since $\mathcal{G}$ is continuous, for each open set U_i there exists an open set V_i such that $\mathcal{G}^{-1}(U_i) \subseteq V_i$. But K is compact, so there exists a finite sub-cover $\{V_{i_1}, V_{i_2}, \ldots, V_{i_n}\}$ of K. Applying $\mathcal{G}$ to this finite sub-cover yields to a finite sub-cover $\{U_{i_1}, U_{i_2}, \ldots, U_{i_n}\}$ of $\mathcal{G}(K)$, proving that $\mathcal{G}(K)$ is compact and concludes the proof. □

Proposition 3. *Any compact subset of a Hilbert space is closed and bounded.*

For the remainder of Section 3, the work uses [15] as the main reference.

3.2 General NOs

We then present the specific mathematical formulation of the problem addressed in this thesis. This includes a detailed description of the operator learning framework and rigorous statements of the core results, supported by proofs where appropriate.

Since the computational part of this work focuses on the implementation of the two-dimensional Darcy flow problem, as considered in [9], let us introduce the family of PDEs of the form:

$$\begin{aligned}(L_a u)(x) &= f(x), \quad x \in D,\\ u(x) &= 0, \quad x \in \partial D,\end{aligned}$$

where $a \in \mathcal{A}$ is the input function and $u \in \mathcal{U}$ is the output function, i.e the solution of the PDE. Given such a PDE, an instinctive operator we can think of is $\mathcal{G}^\dagger := L_a^{-1} f : \mathcal{A} \to \mathcal{U}$,

which maps the input function a to the output solution u. In the case of the Darcy flow problem, L_a is identified as $-\nabla(a\nabla)$. From an implementation perspective, we assume we have $n \in \mathbb{N}$ pairs of input-output functions $\{a^{(i)}, u^{(i)}\}_{i=1}^{n}$ to train the model, where the $a^{(i)}$ are i.i.d samples from a probability distribution and the $u^{(i)}$ are most often computed using numerical methods (the push-forwards under $\mathcal{G}^\dagger$).

We later demonstrate that such operator $\mathcal{G}^\dagger$, can be approximated by a FNO. Before doing so, we first introduce the main mathematical framework and provide a formal definition of a neural operator.

Let $D \subset \mathbb{R}^d$ be a domain. We consider the Banach spaces $\mathcal{A}(D;\mathbb{R}^{d_a})$ and $\mathcal{U}(D;\mathbb{R}^{d_u})$ defined as follows:

$$\mathcal{A}(D;\mathbb{R}^{d_a}) = \left\{a : D \to \mathbb{R}^{d_a}, d_a \in \mathbb{N}\right\}$$

$$\mathcal{U}(D;\mathbb{R}^{d_u}) = \left\{u : D \to \mathbb{R}^{d_u}, d_u \in \mathbb{N}\right\}$$

and the approximation of operators $\mathcal{G} : \mathcal{A}(D;\mathbb{R}^{d_a}) \to \mathcal{U}(D;\mathbb{R}^{d_u})$. Here, $\mathcal{A}(D;\mathbb{R}^{d_a})$ denotes the space of vector-valued input functions; for instance, the set of coefficient functions in the Darcy flow equation, while $\mathcal{U}(D;\mathbb{R}^{d_u})$ denotes the corresponding space of vector-valued output functions, such as the solutions associated with those coefficients. A well-know example of such spaces is the space of Sobolev spaces $H^s(D;\mathbb{R}^{d_a})$, with $s \geq 0$.

With this framework in place, we are now ready to define what a neural operator is.

Definition 3.7. *A* Neural Operator *(NO) is a mapping $\mathcal{N}$ defined as follows:*

$$\begin{array}{rccl} \mathcal{N} & : \mathcal{A}(D;\mathbb{R}^{d_a}) & \to & \mathcal{U}(D;\mathbb{R}^{d_u}) \\ & a & \mapsto & \mathcal{N}(a) = \mathcal{Q} \circ \mathcal{L}_T \circ \mathcal{L}_{T-1} \circ \cdots \circ \mathcal{L}_1 \circ \mathcal{P}(a) \end{array}.$$

$\mathcal{P} \in \mathbb{R}^{d_v \times d_a}$ is the lifting operator that lifts the co-domain of the input function to a higher-dimensional space and $\mathcal{Q} \in \mathbb{R}^{d_u \times d_v}$ is the projection operator that projects the co-domain of the intermediate function to the output co-dimension.

The layers $\mathcal{L}_t$ are of the form:

$$\mathcal{L}_t(v)(x) = \sigma\left(W_t v(x) + b_t(x) + \mathcal{K}((a,\theta_t)v)(x)\right), \quad \forall x \in D, t = 1, \cdots, T,$$

where $\sigma : \mathbb{R} \to \mathbb{R}$ *is a non-linear activation function, $W_t \in \mathbb{R}^{d_v \times d_v}$ represent the*

weight matrices parametrized by a point-wise NN and $\mathcal{K}(a,\theta)$ represents the integral kernel operator defined as:

$$\begin{array}{rccl} \mathcal{K}: & \mathcal{A}\times\Theta & \to & L\left(\mathcal{U}(D;\mathbb{R}^{d_v}),\mathcal{U}(D;\mathbb{R}^{d_v})\right) \\ & (a,\theta) & \mapsto & (\mathcal{K}((a,\theta)v)(x)=\int_D \kappa_\theta(x,y;a(x),a(y))v(y)\,dy \quad \forall x\in D, \end{array}$$

$b_t \in \mathcal{U}(\mathbb{T}^d;\mathbb{R}^{d_v})$ define the bias functions and $\kappa_\theta : \mathbb{R}^{2(d+d_a)} \to \mathbb{R}^{d_v\times d_v}$ is the parameterized kernel function.

The following lemma presents an important property of NOs, which is instrumental in the subsequent proofs. Its proof is omitted here but can be found in the reference cited.

Lemma 1 (Lemma 46, [15])**.** *A composition of NOs can be represented by a neural operator.*

Unlike traditional neural networks, which are inherently tied to a fixed input resolution, NOs possess a remarkable property of discretization-invariance, allowing them to learn mappings between function spaces independent of the underlying grid or resolution. An example of discretization-invariance, where the model's performance remains consistent across different discretizations is shown in Figure 2. However, as highlighted in [9], achieving this property requires the number of model parameters to remain fixed; otherwise, the parameter count would grow unboundedly as the discretization becomes finer. Moreover, the model should satisfy the three following properties:

1. Accepts any set of points in the input domain.
2. Can be evaluated at any point of the output domain.
3. Converges to a continuum operator as the discretization is refined.

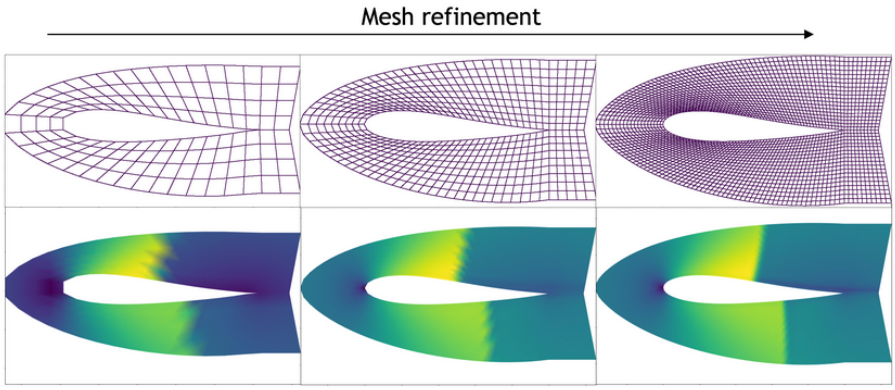


Figure 2. Discretization Invariance: An operator whose predictions remain consistent as the discretization of the input mesh is refined, [9].

To show that neural operators are discretization-invariance, let us frame it within a more mathematical context.

Definition 3.8. *Let $D \subset \mathbb{R}^d$ be a domain. A sequence of sets $D_1 \subset D_2 \subset \cdots \subset D$, with $|D_L| = L$ for any $L \in \mathbb{N}$ such that, for any $\varepsilon > 0$, there exists a number $L(\varepsilon) \in \mathbb{N}$ satisfying:*

$$D \subset \bigcup_{x \in D_L} \{y \in \mathbb{R}^d : \|y - x\|_2 < \varepsilon\},$$

is called a discrete refinement of D. *Moreover, an element D_L of such a refinement is called* a discretization of D.

Definition 3.9. *Let $\Theta \subset \mathbb{R}^p$ be a parameter space, $\mathcal{A}$ a Banach space of $\mathbb{R}^m$-valued functions on D. Given a parametric class of operators $\mathcal{G} : \mathcal{A} \times \Theta \to \mathcal{U}$ and a discrete refinement $(D_n)_{n=1}^{\infty}$ of D, $\mathcal{G}$ is said to be* discretization-invariant *if there exists a sequence of maps $\widehat{\mathcal{G}}_1, \widehat{\mathcal{G}}_2, \cdots$, where $\widehat{\mathcal{G}}_L : \mathbb{R}^{Ld} \times \mathbb{R}^{Lm} \times \Theta \to \mathcal{U}$ such that, for any $\theta \in \Theta$ and any compact set $K \subset \mathcal{A}$,*

$$\lim_{L \to \infty} R_K(\mathcal{G}(\cdot, \theta), \widehat{\mathcal{G}}_L(\cdot, \cdot, \theta), D_L) = 0.$$

In the above, R_K represents the discretized uniform risk defined as

$$R_K(\mathcal{G}, \widehat{\mathcal{G}}, D_L) = \sup_{a \in K} \|\widehat{\mathcal{G}}(D_L, a|_{D_L}) - \mathcal{G}(a)\|_{\mathcal{U}},$$

for some map $\widehat{\mathcal{G}} : \mathbb{R}^{Ld} \times \mathbb{R}^{Lm} \to \mathcal{U}$.

With this background in place, we now introduce the following theorem. But first, let us explain the notation $\mathrm{NO}_n(\sigma_1, \sigma_2, \sigma_3; D)$. This represents the set of neural operators

with depth n, defined on a domain $D \subset \mathbb{R}^d$, constructed using three continuous activation functions: $\sigma_1, \sigma_2, \sigma_3$. Each of these functions serves a unique purpose within the architecture: σ_1 is applied after the initial layer, the lifting layer; σ_2 is utilized within the main operator layers, typically integral layers or spectral layers, where the core computation and learning process occur; and σ_3 is used in the final layer, the projection layer.

Theorem 1 (Theorem 8, [9]). *Let $D \subset \mathbb{R}^d$ be a domain, $\sigma_1, \sigma_2, \sigma_3$ three continuous functions on $\mathbb{R}$. Consider also $\mathcal{A}$ and $\mathcal{U}$ two real-valued Banach function spaces on D that can be continuously embedded in $C(\bar{D})$. Thus, for any $n \in \mathbb{N}$, the set of neural operators $NO_n(\sigma_1, \sigma_2, \sigma_3; D)$, considered as operators from $\mathcal{A} \to \mathcal{U}$, exhibits discretization-invariance.*

Proof. We can assume without loss of generality that $\mathcal{A} = \mathcal{U} = C(\bar{D})$ and start by defining a sequence of discrete approximations for the input space that converges to the full function space. This is, Define a sequence of discretizations $(D_j)_{j\in\mathbb{N}}$, where each D_j consists of a finite set of points $\{y_1, y_2, \ldots, y_j\} \subset D$. For each D_j, let $\{P_j^{(1)}, \ldots, P_j^{(j)}\}$ be a partition of D such that:

1. The sets $P_j^{(k)}$ are pairwise disjoint.
2. Each partition contains exactly one point from D_j and has positive measure.
3. The union of all partitions covers D: $\bigcup_{k=1}^{j} P_j^{(k)} = D$.

Assume for simplicity that we have a single-layer neural operator (the case with three layers is handled similarly):

$$\mathcal{G}(a)(x) = \sigma_1 \left(\int_D \kappa(x, y) a(y) dy + b(x) \right),$$

where $\kappa : \bar{D} \times \bar{D} \to \mathbb{R}$ is a learned kernel function, $b : \bar{D} \to \mathbb{R}$ is a bias term, and $\sigma_1 : \mathbb{R} \to \mathbb{R}$ is a continuous activation function.

Consider also the discretized operator $\mathcal{G}_j$ by replacing the integral with a Riemann sum:

$$\mathcal{G}_j(a)(x) = \sigma_1 \left(\sum_{k=1}^{j} \kappa(x, y_k) a(y_k) |P_j^{(k)}| + b(x) \right).$$

The goal is to show that $\mathcal{G}_j$ converges uniformly to $\mathcal{G}$, i.e.,

$$\sup_{a \in K} \|\mathcal{G}_j(a) - \mathcal{G}(a)\|_{\mathcal{U}} \to 0 \quad \text{as } j \to \infty.$$

Since K is compact, there exists a constant $M > 0$ such that:

$$\sup_{a \in K} \|a\|_{C(\bar{D})} \leq M.$$

Define the bound:

$$R = 2(M|D|\|\kappa\|_{C(\bar{D}\times\bar{D})} + \|b\|_{C(\bar{D})}).$$

By the uniform continuity of σ_1, there exists a modulus of continuity ω such that:

$$|\sigma_1(z_1) - \sigma_1(z_2)| \leq \omega(|z_1 - z_2|), \quad z_1, z_2 \in [-R, R],$$

where $\omega(z) \to 0$ as $z \to 0$.

Thus, it suffices to show that, for all $\varepsilon > 0$, there exists n_0 such that for any $m > n_0$, the following inequality holds:

$$\sup_{a \in K} \sup_{x \in \bar{D}} \left| \int_D \kappa(x, y) a(y) dy - \sum_{k=1}^{m} \kappa(x, y_k) a(y_k) |P_m^{(k)}| \right| < \epsilon.$$

Since K is compact, there exist $a_1, \ldots, a_N \in K$ such that for any $a \in K$, there exists a_n with:

$$\|a - a_n\|_{C(\bar{D})} \leq \frac{\epsilon}{4|D|\|\kappa\|_{C(\bar{D}\times\bar{D})}}.$$

By the convergence of Riemann sums, there exists q such that for any $m \geq q$:

$$\sup_{x \in \bar{D}} \left| \sum_{k=1}^{m} \kappa(x, y_k) |P_m^{(k)}| - \int_D \kappa(x, y) dy \right| < |D|\|\kappa\|_{C(\bar{D}\times\bar{D})}.$$

For each a_n, there exists p_n such that for any $m \geq p_n$:

$$\sup_{x \in \bar{D}} \left| \sum_{k=1}^{m} \kappa(x, y_k) a_n(y_k) |P_m^{(k)}| - \int_D \kappa(x, y) a_n(y) dy \right| < \frac{\epsilon}{4}$$

and since

$$\sup_{x \in \bar{D}} \left| \int_D \kappa(x, y)(a(y) - a_n(y)) dy \right| \leq |D|\|\kappa\|_{C(\bar{D}\times\bar{D})} \|a - a_n\|_{C(\bar{D})}$$

we have:

$$\begin{aligned}\sup_{x\in\bar{D}}\left|\sum_{k=1}^{m}\kappa(x,y_k)(a_n(y_k)-a(y_k))|P_m^{(k)}|\right| &\leq \|a_n-a\|_{C(\bar{D})}\sup_{x\in\bar{D}}\left|\sum_{k=1}^{m}\kappa(x,y_k)|P_m^{(k)}|\right| \\ &\leq \|a_n-a\|_{C(\bar{D})}\left(\sup_{x\in\bar{D}}\left|\sum_{k=1}^{m}\kappa(x,y_k)|P_m^{(k)}|\right|\right. \\ &\quad \left.-\left|\int_D\kappa(x,y)dy\right|+\sup_{x\in\bar{D}}\left|\int_D\kappa(x,y)dy\right|\right) \\ &\leq 2|D|\|\kappa\|_{C(\bar{D}\times\bar{D})}\|a_n-a\|_{C(\bar{D})}.\end{aligned}$$

Thus, for $m \geq \max(q, p_1, \ldots, p_N)$ and by the triangle inequality, we obtain:

$$\begin{aligned}\sup_{x\in\bar{D}}\left|\int_D\kappa(x,y)a(y)dy-\sum_{k=1}^{m}\kappa(x,y_k)a(y_k)|P_m^{(k)}|\right| &\leq \frac{\epsilon}{4}+\frac{\epsilon}{4}+2\frac{\epsilon}{4} \\ &= \epsilon.\end{aligned}$$

This shows that $\mathcal{G}_j(a)$ converges uniformly to $\mathcal{G}(a)$, which concludes the proof.

□

3.3 Fourier Neural Operators (FNOs)

In this section, we introduce the Fourier neural operator framework and describe its underlying architecture.

Throughout the remainder of Chapter 3, we consider the domain $D = \mathbb{T}^d = [0, 2\pi]^d$, the d-dimensional torus. We begin by recalling key concepts from Fourier analysis that will be fundamental to the formulation of FNOs.

Definition 3.10. *Let $v \in L^2(\mathbb{T}^d)$, its Fourier and inverse Fourier transforms are respectively defined as*

$$\mathcal{F}(v)(k) = \frac{1}{(2\pi)^d}\int_{\mathbb{T}^d} v(x)e^{-i\langle k,x\rangle}\,dx, \quad \forall k \in \mathbb{Z}^d$$

and

$$\mathcal{F}^{-1}(\widehat{v})(x) = \sum_{k\in\mathbb{Z}^d} \widehat{v}_k e^{i\langle k,x\rangle}, \quad \forall x \in \mathbb{T}^d.$$

The coefficient $\widehat{v}_k = \mathcal{F}(v)(k)$ is called the k-th Fourier coefficient of v.

The discrete counterpart is defined as follows:

Definition 3.11. *Let $v \in \mathbb{R}^{\mathcal{J}_N}$, the discrete Fourier transform and its inverse are respectively defined as*

$$\mathcal{F}_N(v)(k) = \frac{1}{(2N+1)^d} \sum_{j\in\mathcal{J}_N} v_j e^{-\frac{2\pi i\langle j,k\rangle}{N}}, \quad \mathcal{J}_N := \{0,\cdots,2N\}^d$$

and

$$\mathcal{F}_N^{-1}(\widehat{v})(j) = \sum_{k\in\mathcal{K}_N} \widehat{v}_k e^{\frac{2\pi i\langle j,k\rangle}{N}}, \quad \mathcal{K}_N := \left\{k \in \mathbb{Z}^d \mid |k|_\infty \le N\right\}.$$

We also define the orthogonal Fourier projection operator as well as the set of Fourier wavenumbers used in the proofs.

Definition 3.12. The orthogonal Fourier projection operator $P_N : H^s(\mathbb{T}^d) \to H_N^s(\mathbb{T}^d)$, $s \ge 0$ *is defined as:*

$$P_N\left(\sum_{k\in\mathbb{Z}^d} c_k e^{i\langle k,x\rangle}\right) = \sum_{k\in\mathcal{K}_N} c_k e^{i\langle k,x\rangle}, \quad \forall (c_k)_{k\in\mathbb{Z}^d} \in \ell^2(\mathbb{Z}^d),$$

Definition 3.13 (FNOs)**.** Fourier Neural Operators *(FNOs) are a specific class of neural operators in which the kernel integral operator is parameterized using the convolution theorem (see [23] for details), leading to the following formulation:*

$$(\mathcal{K}(\theta)v)(x) = \mathcal{F}^{-1}\left(P_\theta(k)\cdot\mathcal{F}(v)(k)\right)(x), \quad \forall x \in \mathbb{T}^d, k \in \mathbb{Z}^d,$$

where $P_\theta(k) = \mathcal{F}(\kappa_\theta)(k) \in C^{d_v\times d_v}$ is a full matrix that should satisfy $P_\theta(-k) = P_\theta(k)^\dagger$ to ensure $(\mathcal{K}(\theta)v)(x)$ to be real-valued for real-valued $v(x)$.

With these adjustments, $\mathcal{L}_t$ from Definition 3.7 becomes:

$$\mathcal{L}_t(v)(x) = \sigma\left(W_t v(x) + b_t(x) + \mathcal{F}^{-1}\left(P_t(k)\cdot\mathcal{F}(v)(k)\right)(x)\right), \forall x \in \mathbb{T}^d, t = 1,\cdots,T,$$

where $P_t : \mathbb{Z}^d \to \mathbb{C}^{d_v \times d_v}$ represent the coefficients of a non-local, linear mapping via the Fourier transform.

To better understand the workings of NOs and the role of the Fourier layer, Figure 3 illustrates the architecture of NOs, with the first part (top) showing the processing of the input $a(x)$ through a lifting operator, Fourier layers, and a projector operator to produce the output $u(x)$. The second part (bottom) provides a detailed view of the Fourier layer, highlighting the Fourier transform, its inverse, activation function, and the learned weight that enable the operator to approximate the solution to the PDE.

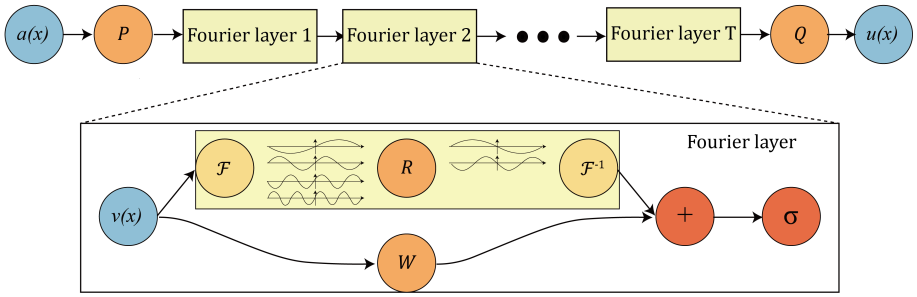


Figure 3. Architecture of NOs and Fourier layer, [7].

Since FNOs are a special case of NOs, they are also discretization-invariant; and in addition to this remarkable property, FNOs possess the universal approximation property, which we explore in the following subsection.

3.4 Universal Approximation Theorem for FNOs

This subsection is dedicated to establishing the universal approximation property of FNOs, a key theoretical result in this work. We begin by recalling the following definition, which is used throughout the proof of the theorem.

Definition 3.14. *Let H be a Hilbert space. An operator $\mathcal{G} : H \to L^2$ is said to be* uniformly continuous on a compact set $K \subset H$ *if there exists a modulus of continuity $\omega : [0, \infty) \to [0, \infty)$ such that*

$$\|\mathcal{G}(a) - \mathcal{G}(a')\|_{L^2} \leq \omega\left(\|a - a'\|_H\right), \quad \forall a, a' \in K.$$

Fourier neural operator models achieve *universal approximation*, meaning they can learn

and represent any continuous operator that maps functions to functions. This is an extension of the classical universal approximation theorem for neural networks, which applies to mappings between finite-dimensional spaces (see Theorem 2 for the mathematical formulation).

The key reason why they can achieve this lies in their layer structure, which includes nonlinear activation functions. These activations introduce the necessary complexity for the model to capture a wide range of function-to-function relationships. Without nonlinearity, the model would be limited to simple linear mappings, reducing its expressive power. Thus, by composing multiple layers with nonlinear activations, FNOs become highly expressive and flexible.

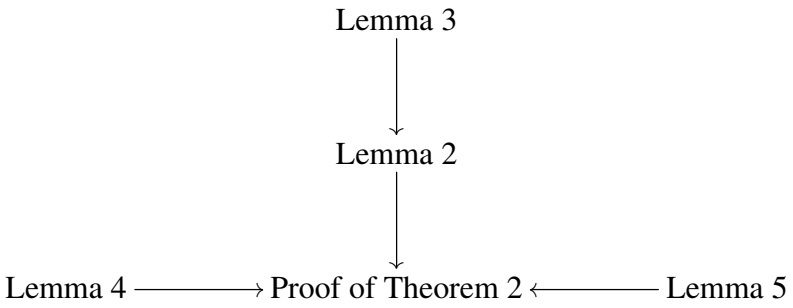
For the rest of this chapter, we denote $H^s(\mathbb{T}^d;\mathbb{R}^{d_a})$ by H^s , $H^{s'}(\mathbb{T}^d;\mathbb{R}^{d_u})$ by $H^{s'}$ and $H^s(\mathbb{T}^d;\mathbb{T}^d)$ by $H^s(\mathbb{T}^d)$ for simplicity.

Theorem 2 (Universal approximation theorem, Theorem 5, [15])**.** *Let s and s' be two nonnegative numbers and $\mathcal{G} : H^s \to H^{s'}$ be a continuous operator. Then, for any $\varepsilon > 0$, there exists a continuous FNO $\mathcal{N} : H^s \to H^{s'}$ satisfying:*

$$\sup_{a\in K} \|\mathcal{G}(a) - \mathcal{N}(a)\|_{H^{s'}} \leq \varepsilon.$$

For ease of notation, we set $d_a = d_u = 1$. The case $d_a, d_u > 1$ follows analogously.

To prove Theorem 2, we proceed in two main steps: The first one consists of showing that there exists a continuous operator $\mathcal{G}_N : H^s(\mathbb{T}^d) \to L^2(\mathbb{T}^d), N \in \mathbb{N}$ that uniformly approximates $\mathcal{G}$. The second one is to show that this operator $\mathcal{G}_N$ can itself be uniformly approximated by a FNO $\mathcal{N} : L^2(\mathbb{T}^d) \to L^2(\mathbb{T}^d)$; the conclusion then follows by restricting $\mathcal{N}$ to $H^s(\mathbb{T}^d)$. The evidence is organized through a series of step-by-step findings, which are connected and outlined in the diagram below. These findings are explained and demonstrated in the sections that follow.



Lemma 2 (Lemma 6, [15]). *If Theorem 2 holds for $s' = 0$, then it holds for $s' \geq 0$. That is, for any continuous operator $\mathcal{G} : H^s \to H^{s'}$ and $K \subset H^s$ compact, if there exist FNO approximations of operators $H^s \to L^2$, then for any $\varepsilon > 0$, FNO approximations of operators $H^s \to H^{s'}$ exist with accuracy ε.*

The proof of Lemma 2 relies on the result of Lemma 3, which we state and prove below:

Lemma 3 (Lemma 43, [15]). *Let s' be a nonnegative number and N a positive integer. Let $K \subset H^{s'}$ be a compact and assume there is an integer $m > s'$ such that $\sigma \in C^m$. Then, $\forall \varepsilon > 0$, there exists $\mathcal{L} : H^{s'} \to H^{s'}$ a FNO with single-layer such that*

$$sup_{v \in K} \|P_N v - \mathcal{L}(v)\|_{H^{s'}} \leq \varepsilon.$$

Proof. We begin by noting that the Fourier projection operator $P_N : H^{s'} \to H^{s'}$ is continuous and since K is compact, $P_N K \subset H^{s'}$ is compact. Because P_N maps into a finite-dimensional subspace of $H^{s'}$ spanned by smooth trigonometric functions and because norms are equivalent in finite-dimensional spaces, there exists a positive constant $C_0 = C_0(N, K)$ such that

$$\sup_{v \in K} \|P_N v\|_{L^\infty} \leq C_0, \quad \sup_{v \in K} \|P_N v\|_{H^m} \leq C_0.$$

Next, consider $x_0 \in \mathbb{R}$ such that $\sigma'(x_0) \neq 0$ and

$$\psi_h(x) = \frac{\sigma(x_0 + hx) - \sigma(x_0 - hx)}{2h\sigma'(x_0)}, \quad h > 0.$$

It is straightforward to verify that $\psi_h \in C^m$, and because $\sigma \in C^m$ and $[-C_0, C_0]$ is

compact, there exists a constant $C_1 = C_1(\sigma, C_0) > 0$ such that

$$\|\psi_h\|_{C^m([-C_0,C_0])} \leq C_1, \quad \forall h \in (0,1].$$

Moreover, by the second-order Taylor expansion of ψ_h on $(x_0 - hx, x_0 + hx)$ with $x \in [-C_0, C_0]$, we have

$$|\psi_h(x) - x| \leq ch, \quad \forall h \in (0,1],$$

for some constant $c > 0$.

Since $m > s'$, $P_N v \in H^m$ implying $\psi_h \circ P_N v \in H^m$. It follows that, there exists $C_2 = C_2(C_1, C_0) > 0$ such that

$$\|\psi_h(P_N v)\|_{H^m} \leq C_2, \quad \forall v \in K.$$

That is,

$$\|\mathcal{L}_h(v)\|_{H^m} \leq C_2, \quad \forall v \in K,$$

with $\mathcal{L}_h(v) = \psi_h(P_N v)$.

Thanks to the Gagliardo-Nirenberg interpolation inequality, the one allowing interpolation of non-integer Sobolev spaces (see [24] for more details), stating that for all $v \in K$ and for $\theta = 1 - \frac{s'}{m} > 0$, the following holds:

$$\begin{aligned}
\|\mathcal{L}_h(v) - P_N v\|_{H^{s'}} &\leq \|\mathcal{L}_h(v) - P_N v\|_{L^2}^{1-\theta} \|\mathcal{L}_h(v) - P_N v\|_{H^m}^{\theta} \\
&= \|\psi_h(P_N v) - P_N v\|_{L^2}^{1-\theta} \|\mathcal{L}_h(v) - P_N v\|_{H^m}^{\theta} \\
&\leq (ch)^{1-\theta} (C_2 h)^{\theta} \\
&= Ch^{\theta}, \quad C = c^{1-\theta} C_2^{\theta} > 0.
\end{aligned}$$

By letting $h \to 0$, it is enough to take $\mathcal{L} = \mathcal{L}_h$ to conclude the proof. □

We can now use this result to prove Lemma 2.

Proof of Lemma 2. Since $\mathcal{G}$ is continuous and K is compact, $\mathcal{G}(K) \subset H^{s'}$ is compact and

therefore, there exists $N \in \mathbb{N}$ so that

$$\sup_{a \in K} \|\mathcal{G}(a) - P_N \mathcal{G}(a)\|_{H^{s'}} \leq \frac{\varepsilon}{2}.$$

Let $\delta > 0$. By assumption, there exists a FNO $\widetilde{\mathcal{N}} : H^s \to L^2$ satisfying

$$\sup_{a \in K} \|P_N \mathcal{G}(a) - \widetilde{\mathcal{N}}(a)\|_{H^{s'}} \leq \delta. \tag{1}$$

Generally, for $s' > s$, $\widetilde{\mathcal{N}}$ does not define a mapping $H^s \to H^{s'}$. To handle this issue, we compose with an additional FNO layer $\widetilde{\mathcal{L}} : L^2 \to H^{s'}$. Lemma 3 allows for the existence of a FNO with single layer satisfying

$$\widetilde{\mathcal{L}}(v) = \widetilde{\mathcal{L}}(P_N v), \forall v,$$

and defines a continuous operator $H^s \to H^{s'}$ so that

$$\sup_{v \in K'} \|P_N v - \widetilde{\mathcal{L}}(v)\|_{H^{s'}} \leq \delta, \tag{2}$$

with $K' = P_N \widetilde{\mathcal{N}}(K) \subset H_{s'}$ compact by the properties of compact sets introduced in Subsection 3.1.

Therefore, the FNO $\mathcal{N} = \widetilde{\mathcal{L}} \circ \widetilde{\mathcal{N}}$ is a continuous operator $H^s \to H^{s'}$ as composition of $\widetilde{\mathcal{N}} : H^s \to L^2$ and $\widetilde{\mathcal{L}} : L^2 \to H^{s'}$ which are both continuous operators.

For any $a \in K$, we have:

$$\begin{aligned}
\|P_N \mathcal{G}(a) - \mathcal{N}(a)\|_{H^{s'}} &\leq \|P_N \mathcal{G}(a) - P_N \widetilde{\mathcal{N}}(a)\|_{H^{s'}} + \|P_N \widetilde{\mathcal{N}}(a) - \mathcal{N}(a)\|_{H^{s'}} \\
&\leq C N^{s'} \|P_N \mathcal{G}(a) - P_N \widetilde{\mathcal{N}}(a)\|_{L^2} + \|P_N \widetilde{\mathcal{N}}(a) - \widetilde{\mathcal{L}}(P_N \widetilde{\mathcal{N}}(a))\|_{H^{s'}} \\
&\leq C N^{s'} \delta + \|P_N v - \widetilde{\mathcal{L}}(v)\|_{H^{s'}}, \quad v = P_N \widetilde{\mathcal{N}}(a) \in K' \\
&\leq C N^{s'} \delta + \delta \\
&= (C N^{s'} + 1)\delta.
\end{aligned}$$

In the above, the first inequality is due to the triangular inequality; the second one is because $\|P_N v\|_{H^{s'}} \leq C N^{s'} \|P_N v\|_{L^2}$ for $C = C(\mathbb{T}^d, s') > 0$ and because $\mathcal{N}(a) = \widetilde{\mathcal{L}}(\widetilde{\mathcal{N}}(a)) = \widetilde{\mathcal{L}}(P_N \widetilde{\mathcal{N}}(a))$. The third inequality is because of Inequality 1 and also because

$$\|P_N v - \widetilde{\mathcal{L}}(v)\|_{H^{s'}} = \|P_N \widetilde{\mathcal{N}}(a) - \widetilde{\mathcal{L}}(P_N \widetilde{\mathcal{N}}(a))\|_{H^{s'}} \leq \delta$$

holds by Inequality 2, and lastly, the fourth inequality is also due to Inequality 2.

Now, because $\delta > 0$ was arbitrary, we can choose it so that it satisfies $(CN^{s'} + 1)\delta \leq \frac{\varepsilon}{2}$.

That being said, $\forall \delta > 0$ satisfying $(CN^{s'} + 1)\delta \leq \frac{\varepsilon}{2}$, we have $\|P_N\mathcal{G}(a) - \mathcal{N}(a)\|_{H^{s'}} \leq \frac{\varepsilon}{2}$.

Therefore, there exists a FNO $\mathcal{N} : H^s \to H^{s'}$ so that $\forall a \in K$,

$$\begin{aligned}\|\mathcal{G}(a) - \mathcal{N}(a)\|_{H^{s'}} &\leq \|\mathcal{G}(a) - P_N\mathcal{G}(a)\|_{H^{s'}} + \|P_N\mathcal{G}(a) - \mathcal{N}(a)\|_{H^{s'}}, \\ &\leq \frac{\varepsilon}{2} + \frac{\varepsilon}{2} \\ &= \varepsilon,\end{aligned}$$

implying $\sup_{a\in K} \|\mathcal{G}(a) - \mathcal{N}(a)\|_{H^{s'}} \leq \varepsilon$, which concludes the proof. □

Lemma 4 (Lemma 7, [15])**.** *Let $B > 0$ and N an integer. For all $\varepsilon > 0$, there exists a FNO*

$$\begin{array}{rccc}\mathcal{N}: & L^2(\mathbb{T}^d) & \to & L^2(\mathbb{T}^d; \mathbb{R}^{d_v}) \\ & v & \mapsto & \{\mathcal{N}(v)_{l,k}\}\end{array},$$

with constant output functions (constant as function of $x \in \mathbb{T}^d$) such that

$$\Re(\widehat{v}_k) - \mathcal{N}(v)_{1,k}\|_{L^\infty} \leq \varepsilon$$

and

$$\Im(\widehat{v}_k) - \mathcal{N}(v)_{2,k}\|_{L^\infty} \leq \varepsilon,$$

for all $k \in \mathcal{K}_N$ and $\|v\|_{L^2} \leq B$, with $\widehat{v}_k \in \mathbb{C}$ being the k-th Fourier coefficient of v.

Proof. The proof of this lemma can be done in three main steps.

Firstly, we aim to construct a FNO $\mathcal{N}_1 : L^2(\mathbb{T}^d; \mathbb{R}) \to L^2(\mathbb{T}^d; \mathbb{R}^{d_v})$ satisfying

$$\|\mathcal{N}_1(v)_{1;k} - P_N v(x)\cos(\langle k, x\rangle)\|_{L^\infty} < \varepsilon$$

and

$$\|\mathcal{N}_1(v)_{2;k} - P_N v(x)\sin(\langle k, x\rangle)\|_{L^\infty} < \varepsilon,$$

for all $k \in \mathcal{K}_N$.

To do so, we start by defining

$$\begin{array}{rccl} \mathcal{R}_1: & L^2(\mathbb{T}^d;\mathbb{R}) & \to & L^2(\mathbb{T}^d;\mathbb{R}^{d_v}) \\ & v(x) & \mapsto & w(x) = (v(x),0,v(x),0)_{k\in\mathcal{K}_N} \in \mathbb{R}^{d_v} \end{array},$$

where $d_v = 4|\mathcal{K}_N|$.

Next, we define the first FNO layer $\widehat{\mathcal{L}}_1$ by

$$\widehat{\mathcal{L}}_1(\widehat{v})(x) := W\widehat{w}(x) + b(x) + \mathcal{F}^{-1}(\mathcal{F}\widehat{w})(x),$$

where $W \equiv 0, \quad b(x) = \{0, \cos(\langle k, x\rangle), 0, \sin(\langle k, x\rangle))\}_{k\in\mathcal{K}_N}$ and

$$P(k) = \begin{cases} 1_{d_v\times d_v} & \text{if } |k| \le N, \\ 0 & \text{otherwise.} \end{cases}$$

Thus, $\widehat{\mathcal{L}}_1$ becomes:

$$\widehat{\mathcal{L}}_1(\widehat{v})(x) = \left\{ (P_N\widehat{w}_{1,k}(x), \cos(\langle k, x\rangle), P_N\widehat{w}_{3,k}(x), \sin(\langle k, x\rangle)) \right\}_{k\in\mathcal{K}_N}.$$

But we have $\|v\|_{L^2} \le B$ by assumption, and $\widehat{w}_{1,k}(x), \widehat{w}_{3,k}(x) = v(x)$ by construction, which leads to

$$\|P_N\widehat{w}_{1,k}\|_{L^\infty} = \|P_N\widehat{w}_{3,k}\|_{L^\infty} = \|P_N v\|_{L^\infty} \le C\|P_N v\|_{L^2} \le CB,$$

with $C = C(N) \propto N^{\frac{d}{2}}$ a constant.

The universal approximation theorem for NN (see [25]) allows for the existence of a NN

$$\widehat{\mathcal{N}} : [-CB, CB] \times [-1, 1] \times [-CB, CB] \times [-1, 1] \to \mathbb{R}^2,$$

with activation function σ, so that $\max_{(a,b,c,d)} \left|\widehat{\mathcal{N}}(a, b, c, d) - (ab, cd)\right| < \varepsilon$, for $a, c \in [-CB, CB]$, and $b, d \in [-1, 1]$.

Taking

$$\begin{array}{rccl} \mathcal{N}_1 \quad : & L^2(\mathbb{T}^d;\mathbb{R}) & \to & L^2(\mathbb{T}^d;\mathbb{R}^{d_v}) \\ & v(x) & \mapsto & \widetilde{w}(x) = \widehat{\mathcal{L}}_1(\mathcal{R}_1(v))(x) \quad \mapsto \widehat{\mathcal{N}}(\widetilde{w}(x)) \end{array}$$

suffices to conclude the first step of the proof, since such a $\widehat{\mathcal{L}}_1$ can be represented by a FNO.

Secondly, since $v(x)$ is real-valued and

$$P_N v(x) = \sum_{k \in \mathcal{K}_N} b_k e^{i\langle k,x\rangle},$$

the Fourier coefficients $\widehat{v}_k$ and their conjugates $\overline{\widehat{v}_k}$ satisfy $\widehat{v}_{-k} = \overline{\widehat{v}_k}$ and we have

$$⨍_{\mathbb{T}^d} P_N v(x) \cos(\langle k, x\rangle) dx = \Re(\widehat{v}_k), \quad ⨍_{\mathbb{T}^d} P_N v(x) \sin(\langle k, x\rangle) dx = -\Im(\widehat{v}_k).$$

More precisely, the above implies that

$$\mathcal{F}[P_N v(x) \cos(\langle k, x\rangle)](0) = \Re(\widehat{v}_k), \quad \mathcal{F}[P_N v(x) \sin(\langle k, x\rangle)](0) = -\Im(\widehat{v}_k)$$

and consequently, the Fourier multiplier δ_0 is given by

$$\delta_0(k') = \begin{cases} 1, & \text{if } k' = 0 \\ 0, & \text{if otherwise} \end{cases}, \quad \forall k' \in \mathbb{Z}^d.$$

Together with $w(x) = \mathcal{N}_1(v)(x) = \{(w_{1,k}(x), w_{2,k}(x))\}_{k \in \mathcal{K}_N} \in L^2(\mathbb{T}^d; \mathbb{R}^{d_v})$, we have:

$$\|\mathcal{F}^{-1}\left(\delta_0(k')\mathcal{F}(w_{1,k})(k_0)\right)(x) - \Re(\widehat{v}_k)\|_{L^\infty} < \varepsilon \tag{3}$$

and

$$\|\mathcal{F}^{-1}\left(\delta_0(k')\mathcal{F}(w_{2,k})(k_0)\right)(x) + \Im(\widehat{v}_k)\|_{L^\infty} < \varepsilon, \tag{4}$$

for all $v \in L^2(\mathbb{T}^d)$, $\|v\|_{L^2} \leq B$, by the first step.

Equations 3 and 4 above allows us to define the second FNO layer $\widehat{\mathcal{L}}_2$ by

$$\widehat{\mathcal{L}}_2(w) = Ww(x) + b(x) + \mathcal{F}^{-1}\left(P\mathcal{F}w\right)(x),$$

where $W = 0, b = 0$ and

$$\begin{array}{rccl} P: & \mathbb{Z}^d & \to & \mathbb{C}^{d_v \times d_v} \\ & k' & \mapsto & [P(k')]_{(l,k),(\widetilde{l},\widetilde{k})} = \delta_0(k')\left\{\delta_{l=1}(l) - \delta_{l=2}(l)\right\} 1_{d_v \times d_v} \end{array},$$

with $d_v = 2|\mathcal{K}_N|$. But we have $\|v\|_{L^2} \leq B$, $\forall v \in L^2(\mathbb{T}^d)$ by assumption and $w(x) = \widehat{\mathcal{L}}_2(\mathcal{N}_1(v))(x) = \{(w_{1,k}(x), w_{2,k}(x))\}_{k \in \mathcal{K}_N}$ by construction, which leads to

$$\|w_{1;k}(x) - \Re(\widehat{v}_k)\|_{L^\infty} < \varepsilon \text{ and } \|w_{2;k}(x) - \Im(\widehat{v}_k)\|_{L^\infty} < \varepsilon. \tag{5}$$

Since $\widehat{\mathcal{L}}_2$ is missing the non-linearity σ, $\widehat{\mathcal{L}}_2 \circ \mathcal{N}_1$ does not define a FNO. To handle this, we can compose it with a suitable NN $\widetilde{\mathcal{N}}$ which approximates the identity. Inequalities in 5 imply that $|w_{l,k}| \leq |\widehat{v}_k| + \varepsilon \leq B + \varepsilon$, for all l, k, and therefore, the universal approximation theorem for NN allows for the existence of a NN $\widetilde{\mathcal{N}} : \mathbb{R}^{d_v} \to \mathbb{R}^{d_v}$ satisfying :

$$\|\widetilde{\mathcal{N}}(w) - w\|_{\ell^\infty} < \varepsilon, \quad \forall w \in \mathbb{R}^{d_v}, \|w\|_{\ell^\infty} \leq B + \varepsilon.$$

By doing so,

$$\begin{array}{rccc} \mathcal{N} = \widetilde{\mathcal{N}} \circ \widehat{\mathcal{L}}_2 : & L^2(\mathbb{T}^d; \mathbb{R}^{d_v}) & \to & L^2(\mathbb{T}^d; \mathbb{R}^{d_v}) \\ & w(x) & \mapsto & \widetilde{\mathcal{N}}(\widehat{\mathcal{L}}_2(w)(x)) \end{array}$$

defines a FNO.

Finally, since $\mathcal{N}_1$ and $\mathcal{N}_2$ define FNOs, $\mathcal{N} = \mathcal{N}_1 \circ \mathcal{N}_2 : L^2(\mathbb{T}^d) \to L^2(\mathbb{T}^d; \mathbb{R}^{d_v})$ also defines a FNO.

Moreover, $\mathcal{N}$ maps to constant functions and satisfies for all $\varepsilon > 0$:

$$\|\Re(\widehat{v}_k) - \mathcal{N}(v)_{1,k}\|_{L^\infty} < \varepsilon$$

and

$$\|\Im(\widehat{v}_k) - \mathcal{N}(v)_{2,k}\|_{L^\infty} < \varepsilon,$$

for all $v \in L^2(\mathbb{T}^d)$, $\|v\|_{L^2} \leq B$. This concludes the proof. □

Lemma 5 (Lemma 8, [15])**.** *Let $B > 0$ and N an integer. For all $\varepsilon > 0$, there exists a FNO $\mathcal{N} : L^2(\mathbb{T}^d; \mathbb{R}^{d_v}) \to L^2(\mathbb{T}^d)$ satisfying:*

$$\|v - \mathcal{N}(w)\|_{L^2} \leq \varepsilon, \quad \forall v \in L^2(\mathbb{T}^d), \quad \|v\|_{L^2} \leq B,$$

with $w(x) = \{(\Re(\widehat{v}_k), \Im(\widehat{v}_k))\}_{k \in \mathcal{K}_N}$ be a constant function.

Proof. The proof of Lemma 5 can be done in two main steps.

Firstly, let $d_v = 2|\mathcal{K}_N| = 2(2N+1)$, with $|\mathcal{K}_N|$ being the cardinal of $\mathcal{K}_N$ and $|\mathbb{T}^d| = (2\pi)^d$ be the Lebesgue measure of $\mathbb{T}^d$. We want to construct
$\mathcal{N} : L^2(\mathbb{T}^d; \mathbb{R}^{d_v}) \to L^2(\mathbb{T}^d; \mathbb{R}^{d_v})$ so that

$$\|\mathcal{N}_1(w)_{1,k} - P_N w_{1,k}(x)\cos(\langle k,x\rangle)\|_{L^\infty} < \varepsilon\left(d_v|\mathbb{T}_d|^{1/2}\right)^{-1}$$

and

$$\|\mathcal{N}_1(w)_{2,k} - P_N w_{2,k}(x)\sin(\langle k,x\rangle)\|_{L^\infty} < \varepsilon\left(d_v|\mathbb{T}_d|^{1/2}\right)^{-1}.$$

To do so, we start by defining

$$\mathcal{L}\circ\mathcal{R}(w) = \left\{\left(w_{1,k}(x), \cos(\langle k,x\rangle), w_{2,k}(x), \sin(\langle k,x\rangle)\right)\right\}_{k\in\mathcal{K}_N},$$

where

$$\begin{array}{rccl} \mathcal{R}: & L^2(\mathbb{T}^d;\mathbb{R}^{d_v}) & \to & L^2(\mathbb{T}^d;\mathbb{R}^{d_v}) \\ & \{(w_{1,k}(x), w_{2,k}(x))\}_{k\in\mathcal{K}_N} & \mapsto & \{(w_{1,k}(x), 0, w_{2,k}(x), 0)\}_{k\in\mathcal{K}_N} \end{array}$$

and $\mathcal{L} : L^2(\mathbb{T}^d;\mathbb{R}^{d_v}) \to L^2(\mathbb{T}^d;\mathbb{R}^{d_v})$ uniquely represents the FNO layer with $W = 0$, $b(x) = \{(0,\cos(\langle k,x\rangle),0,\sin(\langle k,x\rangle))\}_{k\in\mathcal{K}_N}$ and $P = 0$.

The universal approximation theorem for NNs allows for the existence of a neural operator

$$\widehat{\mathcal{N}} : [-B,B]\times[-1,1]\times[-B,B]\times[-1,1]\to\mathbb{R}^2,$$

so that $\max_{(a,b,c,d)}\left|\widehat{\mathcal{N}}(a,b,c,d) - (ab,cd)\right| < \varepsilon\left(d_v|\mathbb{T}_d|^{1/2}\right)^{-1}$, for $a,c\in[-B,B]$, and $b,d\in[-1,1]$.

Taking $\mathcal{N}_1(w) = \widehat{\mathcal{N}}\circ\mathcal{L}\circ\mathcal{R}(w)$ suffices to conclude the first step of the proof, since $\|w_{1,k}\|_{L^\infty}, \|w_{2,k}\|_{L^\infty} \neq B$ by assumption and $\mathcal{N}_1$ defined in this way represents a FNO as composition of FNOs.

Next, let us define the projector operator $\mathcal{Q}$ as:

$$\begin{array}{rccl} \mathcal{Q}: & L^2(\mathbb{T}^d;\mathbb{R}^{d_v}) & \to & L^2(\mathbb{T}^d) \\ & w(x) & \mapsto & \mathcal{Q}(w(x)) = \sum_{k\in\mathcal{K}_N}\left(w_{1,k}(x) - w_{2,k}(x)\right). \end{array}$$

Taking $\mathcal{N} = \mathcal{Q}\circ\mathcal{N}_1$ suffices to conclude the proof.

Actually, for any $v\in L^2(\mathbb{T}^d)$ and $w = \{(\Re(\widehat{v}_k),\Im(\widehat{v}_k))\}_{k\in\mathcal{K}_N}$, we have:

$$
\begin{aligned}
\|v - \mathcal{N}(w)\|_{L^2} &= \sum_{k \in \mathcal{K}_N} \|\mathcal{N}_1(w)_{1;k} - \mathcal{N}_1(w)_{2;k}\|_{L^2} \\
&\leq \sum_{k \in \mathcal{K}_N} \|\mathcal{N}_1(w)_{1,k} - \Re(\widehat{v}_k)\cos(\langle k, x\rangle)\|_{L^2} \\
&\quad + \sum_{k \in \mathcal{K}_N} \|\mathcal{N}_1(w)_{2,k} - \Im(\widehat{v}_k)\sin(\langle k, x\rangle)\|_{L^2} \\
&\leq \frac{1}{2} d_v |\mathbb{T}^d|^{\frac{1}{2}} \max_{k \in \mathcal{K}_N} \|\mathcal{N}_1(w)_{1,k} - w_{1,k}\cos(\langle k, x\rangle)\|_{L^\infty} \\
&\quad + \frac{1}{2} d_v |\mathbb{T}^d|^{\frac{1}{2}} \max_{k \in \mathcal{K}_N} \|\mathcal{N}_1(w)_{1,k} - w_{1,k}\sin(\langle k, x\rangle)\|_{L^\infty} \\
&< \frac{\varepsilon}{2} + \frac{\varepsilon}{2} \\
&= \varepsilon.
\end{aligned}
$$

This concludes the proof. □

Having established the necessary intermediate results, we are now ready to prove Theorem 2.

Proof of Theorem 2. By Lemma 2, it is enough to show that if the theorem holds for $s' = 0$, then it holds for $s' \geq 0$.

Since $K \subset H^s(\mathbb{T}^d)$ is compact, then $\widetilde{K} = K \cup \bigcup_{N \in \mathbb{N}} P_N K$ is compact by the properties of compact sets introduced in Subsection 3.1 . Moreover, since $\mathcal{G}$ is continuous, $\mathcal{G}|_{\widetilde{K}}$ is uniformly continuous.

Now, consider the operator

$$
\begin{array}{rccc}
\mathcal{G}_N : & H^s(\mathbb{T}^d) & \to & L^2(\mathbb{T}^d) \\
& a & \mapsto & P_N \mathcal{G}(P_N a)
\end{array},
$$

we have:

$$
\begin{aligned}
\|\mathcal{G}(a) - \mathcal{G}_N(a)\|_{L^2} &\leq \|\mathcal{G}(a) - P_N\mathcal{G}(a)\|_{L^2} + \|P_N\mathcal{G}(a) - P_N\mathcal{G}(P_N a)\|_{L^2} \\
&\leq \|\mathcal{G}(a) - P_N\mathcal{G}(a)\|_{L^2} + \|\mathcal{G}(a) - \mathcal{G}(a)\|_{L^2} \\
&\leq \sup_{v \in \mathcal{G}(\widetilde{K})} \|(1 - P_N)v\|_{L^2} + w\Big(\sup_{a \in \widetilde{K}} \|(1 - P_N)a\|_{H^s}\Big).
\end{aligned}
$$

In the above, the first inequality is due to the triangular inequality; the second is because P_N is a linear, bounded and lipschitz projection and the last one is because $\mathcal{G}|_{\widetilde{K}}$ is uniformly continuous.

Since $\widetilde{K}$ is compact and $\mathcal{G}$ is continuous, $\mathcal{G}(\widetilde{K})$ is also compact, implying:

$$\limsup_{N\to\infty} \sup_{v\in\mathcal{G}(\widetilde{K})} \|(1-P_N)v\|_{L^2} = 0 = \limsup_{N\to\infty} \sup_{a\in\widetilde{K}} \|(1-P_N)a\|_{H^s}$$

In particular, there exists $\exists N \in \mathbb{N}$ such that

$$\|\mathcal{G}(a) - \mathcal{G}_N(a)\|_{L^2} \leq \frac{\varepsilon}{2} \quad \forall a \in K \subset \widetilde{K}$$

and, therefore,

$$\sup_{a\in K} \|\mathcal{G}(a) - \mathcal{G}_N(a)\|_{L^2} \leq \frac{\varepsilon}{2}. \tag{6}$$

Let us define the Fourier conjugate (Fourier dual) operator

$$\begin{array}{rccl} \widehat{\mathcal{G}}_N: & \mathbb{C}^{\mathcal{K}_N} & \to & \mathbb{C}^{\mathcal{K}_N} \\ & \widehat{a}_k & \mapsto & \mathcal{F}(\mathcal{G}_N(\Re(\mathcal{F}_N^{-1}(\widehat{a}_k))) \end{array}, \mathbb{C}^{\mathcal{K}_N} \equiv \mathbb{R}^{d_v}, d_v = 2\left|\mathcal{K}_N\right|,$$

which satisfies the identity

$$\mathcal{G}_N(a) = \mathcal{F}_N^{-1} \circ \widehat{\mathcal{G}}_N \circ \Big(\mathcal{F}(P_N a)\Big), \quad \forall a \in L^2(\mathbb{T}^d). \tag{7}$$

For the rest of the proof, we are aiming to show that there exists a FNO $\mathcal{N}: L^2(\mathbb{T}^d) \to L^2(\mathbb{T}^d)$, such that

$$\sup_{a\in K} \|\mathcal{G}_N(a) - \mathcal{N}(a)\|_{L^2} \leq \frac{\varepsilon}{2},$$

and therefore $\mathcal{N}|_{H^s(\mathbb{T}^d)}$ will be okay to conclude the proof.

Doing this pass by individually approximate by a FNO the terms in Equation 7 ($\mathcal{F}_N^{-1}$, $\widehat{\mathcal{G}}_N$ and $\mathcal{F}_N \circ P_N$). And to ensure that each FNO in this decomposition maps its domain into

the domain of the next FNO. Let us introduce $R_K, R_{\widehat{K}}, R_{\widehat{\mathcal{G}}} > 0$, such that

$$\begin{cases} K \subset B_{R_K}(0) := \{\|u\|_{L^2} \leq R_K\} \subset L^2(\mathbb{T}^d) \\ \\ \mathcal{F}_N \circ P_N(B_{R_K}(0)) \subset \Big[-\frac{R_{\widehat{K}}}{2}, \frac{R_{\widehat{K}}}{2}\Big]^{d_v} \\ \\ \mathcal{G}_N\Big([-R_{\widehat{K}}, R_{\widehat{K}}]^{d_v}\Big) \subset \Big[-\frac{R_{\widehat{G}}}{2}, \frac{R_{\widehat{G}}}{2}\Big]^{d_v} \end{cases} \tag{8}$$

1. FNO approximation of $\mathcal{F}_N^{-1}$: For the FNO approximation of $\mathcal{F}_N^{-1}$, we interpret

$$\mathcal{F}_N^{-1} : [-R, R]^{d_v} \to L^2(\mathbb{T}^d)$$

as

$$\begin{array}{rccc} \mathcal{F}_N^{-1} : & L^2(\mathbb{T}^d; [-R, R]^{d_v}) & \to & L^2(\mathbb{T}^d) \\ & (\Re(\widehat{v}_k), \Im(\widehat{v}_k))_{|k|\leq N} & \mapsto & v(x) \end{array},$$

where $(\Re(\widehat{v}_k), \Im(\widehat{v}_k))_{|k|\leq N} \subset [-R, R]^{d_v}$ is a constant function in $L^2(\mathbb{T}^d; [-R, R]^{d_v})$.

Thus, Lemma 5 allows for the existence of a FNO
$\mathcal{N}_1 : L^2(\mathbb{T}^d; \mathbb{R}^{d_v}) \to L^2(\mathbb{T}^d)$ satisfying:

$$\|\mathcal{F}_N^{-1}(\widehat{v}) - \mathcal{N}_1(\widehat{v})\|_{L^2} \leq \frac{\varepsilon}{6}, \quad \forall \widehat{v} \in L^2(\mathbb{T}^d; [-R, R]^{d_v})$$

2. FNO approximation of $\widehat{\mathcal{G}}_N$: Since $\widehat{\mathcal{G}}_N$ is continuous and $[-R_{\widehat{K}}, R_{\widehat{K}}]^{d_v}$ is compact, there exists a finite-dimensional canonical NN
$\mathcal{N}_2 : \mathbb{R}^{d_v} \to \mathbb{R}^{d_v}$ such that

$$\sup_{\widehat{v} \in [-R_{\widehat{K}}, R_{\widehat{K}}]^{d_v}} \|\widehat{\mathcal{G}}_N(\widehat{v}) - \mathcal{N}_2(\widehat{v})\|_{L^2} \leq \frac{\varepsilon}{6}.$$

But System 8 includes $\mathcal{G}_N\Big([-R_{\widehat{K}}, R_{\widehat{K}}]^{d_v}\Big) \subset \Big[-\frac{R_{\widehat{G}}}{2}, \frac{R_{\widehat{G}}}{2}\Big]^{d_v}$, so choosing $\mathcal{N}_2$ with sufficiently high accuracy guaranties

$$\mathcal{N}_2\Big([-R_{\widehat{K}}, R_{\widehat{K}}]^{d_v}\Big) \subset \Big[-R_{\widehat{G}}, R_{\widehat{G}}\Big]^{d_v}.$$

Remark 2.

$$\begin{array}{rccc} \mathcal{N}_2 : & L^2(\mathbb{T}^d; \mathbb{R}^{d_v}) & \to & L^2(\mathbb{T}^d; \mathbb{R}^{d_v}) \\ & v & \mapsto & \mathcal{N}_2(v(x)) \end{array}$$

is a FNO with only local layers of the form:

$$v_l(x) \mapsto \sigma\left(A_l v_l(x) + b_l\right), \quad (A_l \in \mathbb{R}^{d_v \times d_v}, b_l \in \mathbb{R}^{d_v}),$$

with $d_v = |\mathcal{K}_N|$ *is the cardinal of* $\mathcal{K}_N$.

3. FNO approximation of $\mathcal{F}_N \circ P_N$: Similarly to the approximation of $\mathcal{F}_N^{-1}$, for the approximation of $\mathcal{F}_N \circ P_N$, we interpret

$$\mathcal{F}_N \circ P_N : B_{R_K}(0) \subset L^2(\mathbb{T}^d) \to \mathbb{R}^{d_v}$$

as

$$\begin{array}{rccc} \mathcal{F}_N \circ P_N : & B_{R_K}(0) & \to & L^2(\mathbb{T}^d; \mathbb{R}^{d_v}) \\ & v & \mapsto & (\Re(\widehat{v}_k), \Im(\widehat{v}_k))_{|k| \leq N} \end{array},$$

where $(\Re(\widehat{v}_k), \Im(\widehat{v}_k))_{|k| \leq N} \subset \mathbb{R}^{d_v}$ is a constant function in $L^2(\mathbb{T}^d; \mathbb{R}^{d_v})$. Thus Lemma 4 allows for the existence of o FNO $\mathcal{N}_3 : B_{R_K}(0) \to L^2(\mathbb{T}^d; \mathbb{R}^{d_v})$ satisfying:

$$\|\mathcal{F}_N \circ P_N(v) - \mathcal{N}_3(v)\|_{L^2} \leq \frac{\varepsilon}{6}, \quad \forall v \in B_{R_K}(0).$$

But System 8 includes $\mathcal{F}_N \circ P_N(B_{R_K}(0)) \subset \left[-\frac{R_{\widehat{G}}}{2}, \frac{R_{\widehat{G}}}{2} \right]^{d_v}$, so choosing $\mathcal{N}_3$ with sufficiently high accuracy guaranties

$$\mathcal{N}_3(B_{R_K}(0)) \subset \left[-R_{\widehat{G}}, R_{\widehat{G}} \right]^{d_v}.$$

Therefore, taking $\mathcal{N} = \mathcal{N}_1 \circ \mathcal{N}_2 \circ \mathcal{N}_3$, we can conclude that

$$\sup_{a \in K} \|\mathcal{G}_N(a) - \mathcal{N}(a)\|_{L^2} \leq \frac{\varepsilon}{2}. \tag{9}$$

Putting together Inequalities 6 and 9, we have:

$$\begin{aligned} \sup_{a \in K} \|\mathcal{G}(a) - \mathcal{N}(a)\|_{L^2} &\leq \sup_{a \in K} \|\mathcal{G}(a) - \mathcal{G}_N(a)\|_{L^2} + \sup_{a \in K} \|\mathcal{G}_N(a) - \mathcal{N}(a)\|_{L^2} \\ &\leq \frac{\varepsilon}{2} + \frac{\varepsilon}{2} \\ &= \varepsilon. \end{aligned}$$

This concludes the proof. □

4 EXPERIMENTS

This section investigates the performance of Fourier neural operators on solving the two-dimensional Darcy flow equation, a model describing the steady-state flow of fluid through a porous medium. The governing partial differential equation is defined as:

$$\begin{cases} -\nabla \cdot (a(x)\nabla u(x)) & = f(x), \quad x \in D, \\ u(x) & = 0, \quad x \in \partial D, \end{cases}$$

where: D is the spatial domain (typically $D = [0, 2\pi]^2$, i.e $d = 2$ in our setting), $a(x)$ is the spatially varying diffusion coefficient (also called the permeability of the medium), $u(x)$ is the pressure field or solution to be predicted and $f(x)$ is a known source term (typically fixed as 1 in this work).

This elliptic PDE is solved under homogeneous Dirichlet boundary conditions. The objective is to learn a mapping from the coefficient function $a(x)$ to the solution $u(x)$ using FNOs, and assess the generalization and approximation capabilities of the model on various datasets.

In this chapter, we detail the implementation of Fourier neural operators to solve the two-dimensional Darcy flow problem. To achieve this, we use the code from the work of [9] (available on the GitHub page [26]), making specific modifications and additions to address the requirements of our study. The initial implementation includes the generation of the coefficient function, as described in their work (Appendix A.3.2), and the analysis of the results by visualizing the ground truth values, the FNO approximation, the related error, and the coefficient function.

First, we utilize the original method to generate the coefficient function, followed by visualizing the results through various plots showing the ground truth, error, and the model's approximation. These results were compared to evaluate the performance of the approximation in terms of efficiency and accuracy.

Next, a variant was introduced in the generation of the coefficient function to test the robustness of the approach. We modified how the coefficients are generated to explore the impact of these adjustments on the model's performance. The results obtained from this new method are also presented and compared with those from the initial implementation.

The chapter is structured as follows: we begin by presenting the data generation process

and the evaluation criteria; then, explaining the details of the base model implementation, and concluding with a detailed presentation of the results.

4.1 Data

The datasets used in the experiments are all defined on the two-dimensional spatial domain $[0, 2\pi]^2$, discretized into a 421×421 uniform grid. For each dataset, a total of 1000 training samples and 100 testing samples are generated. The coefficient function $a(x)$ for each dataset is generated using different methods, as described below, and the corresponding pressure field $u(x)$ is computed by solving the Darcy flow equation using a finite difference method. Specifically, the equation is discretized and solved using a sparse matrix solver, with Dirichlet boundary conditions applied at the domain boundaries. The resulting data, including both the coefficient functions $a(x)$ and the corresponding solutions $u(x)$, are stored in `.mat` files for model training and evaluation.

We use three different datasets, each serving a specific purpose in our analysis. The first one is the same as in the paper; we use this for reference to validate our implementation and ensure consistency with previously published results. The second dataset is designed to test how the FNO architecture handles higher contrast in the coefficient function $a(x)$, providing insight into the model's robustness in more challenging scenarios. The third dataset is used to test more smooth coefficients and evaluate the model's performance when the input field exhibits higher regularity.

Dataset 1: Thresholded Gaussian Random Field (Original dataset from paper [7])

In this dataset, the coefficient function $a(x)$ is generated from a centered Gaussian random field with covariance operator

$$\mathcal{C} = \tau^{2\alpha-2}(-\Delta + \tau^2 I)^{-\alpha},$$

where Δ is the Laplace operator, $\tau = 3 > 0$ and $\alpha = 2 > d/2 = 1$ are parameters controlling the smoothness of the field. The random field is simulated using the Karhunen–Loève expansion.

Let $g(x)$ denote a sample from the Gaussian random field. To enforce ellipticity while introducing moderate contrast between regions, the coefficient function $a(x)$ is defined

by thresholding:

$$a(x) = \begin{cases} 12, & \text{if } g(x) \geq 0, \\ 3, & \text{if } g(x) < 0. \end{cases}$$

Dataset 2: Thresholded Gaussian Random Field (Modified version of the original dataset)

In this dataset, the coefficient function $a(x)$ is generated in the same way as in Dataset 1, but instead of taking values 12 and 3, it takes more widely spaced values: 100 and 10. That is, if $g(x)$ denote a sample from the Gaussian random field, $a(x)$ is defined by thresholding:

$$a(x) = \begin{cases} 100, & \text{if } g(x) \geq 0, \\ 10, & \text{if } g(x) < 0. \end{cases}$$

Dataset 3: Exponentiated Gaussian Random Fields

In this dataset, the coefficient function $a(x)$ is again generated from a Gaussian random field, as in Datasets 1 and 2. However, instead of applying a thresholding operation, we exponentiate the field to ensure that the ellipticity condition is satisfied. That is, if $g(x)$ denote a sample from the Gaussian random field, $a(x)$ is defined by:

$$a(x) = e^{g(x)}.$$

4.2 Evaluation Criteria

To evaluate how well the FNO model predicts solutions for our two-dimensional Darcy flow problem, we use several assessment metrics on the test dataset.

Firstly, the *relative L^2 error* gauges the discrepancy between the model's predictions and the actual (ground truth) solutions across the entire domain. It is calculated as:

$$\text{RelL2} = \frac{\|u_{\text{pred}} - u_{\text{true}}\|_2}{\|u_{\text{true}}\|_2},$$

where u_{pred} stands for the predicted solution and u_{true} represents the ground truth. This

measure tells us how closely the predicted solutions match the true ones in terms of overall error size.

Additionally, we assess the *bias*, indicating any consistent difference between predicted and actual solutions. The bias is the average of the differences between predicted and true values across the test set:

$$\text{Bias} = \frac{1}{N}\sum_{i=1}^{N}(u_{\text{pred},i} - u_{\text{true},i}),$$

with N being the number of test points, and $u_{\text{pred},i}$ and $u_{\text{true},i}$ being the predicted and true values at those points, respectively. A bias near zero suggests the model does not systematically overestimate or underestimate the solutions.

Finally, the *standard deviation* captures the spread or variability of the error between the predicted and true solutions. It is defined as:

$$\text{Std} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(u_{\text{pred},i} - u_{\text{true},i} - \text{Bias})^2}.$$

The standard deviation gives insight into how consistently the model makes predictions. A lower standard deviation means the errors are more uniformly distributed, whereas a higher standard deviation indicates greater variability in errors.

Together, these metrics provide a well-rounded view of the FNO model's performance, including its accuracy (through the relative L^2 error) and reliability (via bias and standard deviation) in solving the two-dimensional Darcy flow problem.

4.3 Description of Experiments

For the experiments, we use primarily the codes provided on the GitHub page [26], which we try to present in this section.

Once the data are generated as described in Subsection 4.1, we can proceed with the experiments, that follow an identical protocol: the same model architecture and training procedure, with only the input data varying between runs. This consistency ensures fair comparison and isolates the impact of data variability on performance.

The FNO2D model is chosen for its ability to efficiently approximate PDE solutions by learning in Fourier space. The architecture combines spectral convolutions with pointwise linear transformations, a design motivated by the observation that many PDE solutions exhibit low-frequency dominance, making Fourier modes a natural basis for parameterization. The model uses 12 Fourier modes per dimension (sufficient to capture dominant solution features while avoiding overfitting) and a channel width of 32, balancing expressivity and computational cost. Inputs and outputs are normalized to unit Gaussian distributions to stabilize training by ensuring consistent scales across different datasets.

Training data consist of 1,000 samples of coefficient-solution pairs $\{a_i, u_i\}$, sampled at 421×421 resolution and down-sampled to 85×85 grids (stride $r = 5$) to reduce memory requirements without significant information loss. The model processes these inputs by concatenating them with a fixed spatial grid encoding the rescaled coordinates, which provides the network with knowledge of absolute positions critical for problems where location influences solution behavior. An independent test set of 100 samples evaluates the model.

The training procedure uses the Adam optimizer with an initial learning rate of 0.001 and L^2 weight decay (10^{-4}) to prevent overfitting. The learning rate is halved every 100 epochs to refine convergence near optima. Batch sizes of 20 ensures stable gradient estimates while fitting GPU memory constraints. All experiments run for 500 epochs sufficient for loss convergence on CUDA enables hardware to accelerate the computationally intensive spectral transforms.

By maintaining identical hyperparameters and code across datasets, the experiments isolate the effect of data characteristics on model performance, providing clear insights into the FNO's generalization capabilities.

4.4 Results

In this section, we present the results we obtained from the experiments we investigated on each dataset.

Original dataset

The average relative error is 9.2688×10^{-3}. Table 1 lists the five test samples from Dataset 2 with the lowest and highest relative L^2 errors. The best-performing sample (index 59) yields a relative error of only 3.9054×10^{-3} , indicating excellent agreement between the predicted and true pressure fields. On the other hand, the worst case (index 61) has a higher, though still moderate, error of 2.6905×10^{-2}. In all reported cases, the errors remain below 3%, which reflects the model's strong ability to learn mappings from highly structured input data.

Table 1. Top 5 indices with the best and worst predictions: Dataset 1

Best index	**RelL2**	**Worst index**	**RelL2**
59	3.9054×10^{-3}	61	2.6905×10^{-2}
96	4.0169×10^{-3}	15	2.3555×10^{-2}
88	4.4161×10^{-3}	4	2.3416×10^{-2}
3	4.4548×10^{-3}	85	2.2230×10^{-2}
7	4.6230×10^{-3}	8	2.0992×10^{-2}

The least well-performing test sample from Dataset 1 is shown in Fig 4. Although the visual agreement between the ground truth and the prediction of the FNO is smooth, the prediction error clearly exhibits such discrepancies, especially near the complex interfaces in the coefficient field. This coefficient pattern, which is highly complex and with irregular shapes and edges, varies sharply between 3 and 12, and results in a more challenging approximation of the underlying PDE solution. The high error in those areas exposed the shortcomings of the model in adapting to a sudden spatial change.

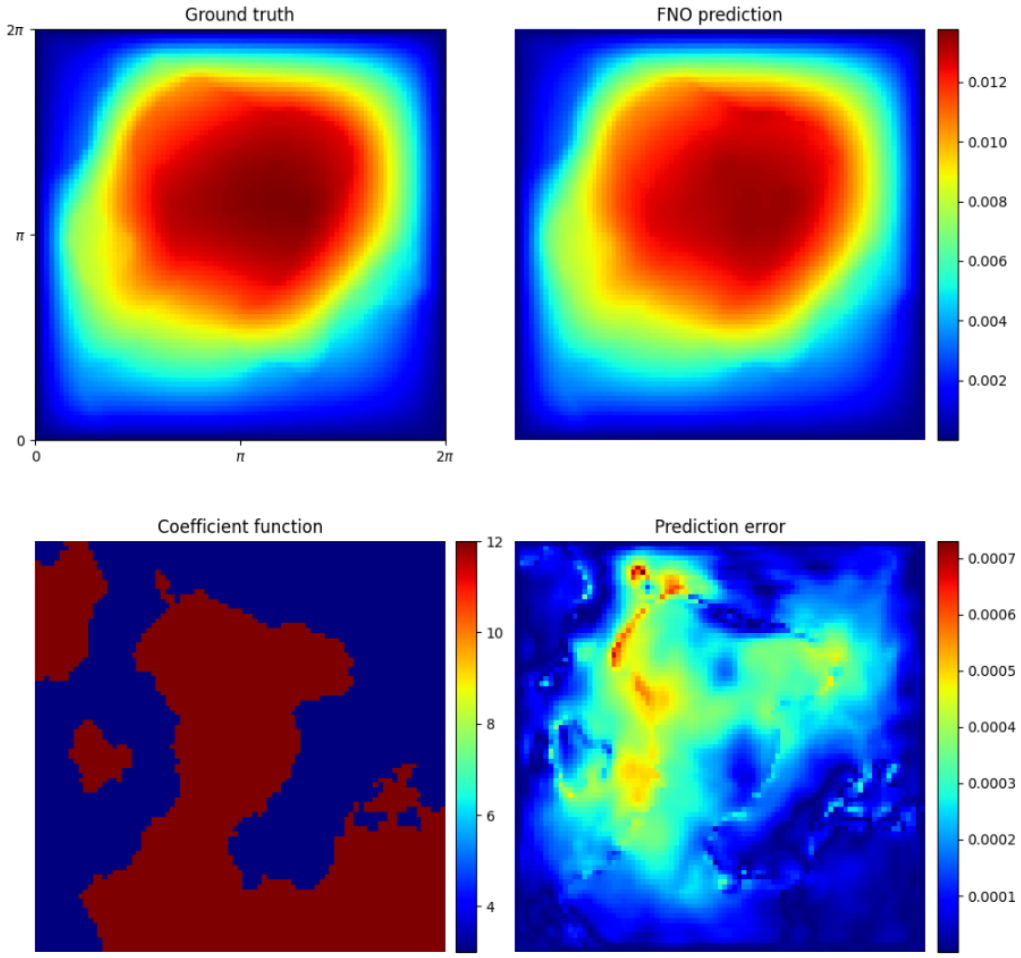


Figure 4. Dataset 1 worst sample. Top left: Ground truth solution (output) to the PDE. Top right: Prediction by the FNO model. Bottom left: Coefficient function (input) to the PDE. Bottom right: Pointwise prediction error between the ground truth and the FNO prediction

Figure 5 shows the best sample obtained, which has a simpler more horizontally stratified structure of the coefficient field. In this case the fine-grid FNO model approximates a perfect solution almost pointwise and the error is very small primarily resulting from the interface between the coefficient layers. This means that simple, low-complexity geometries are easier to generalize for the model.

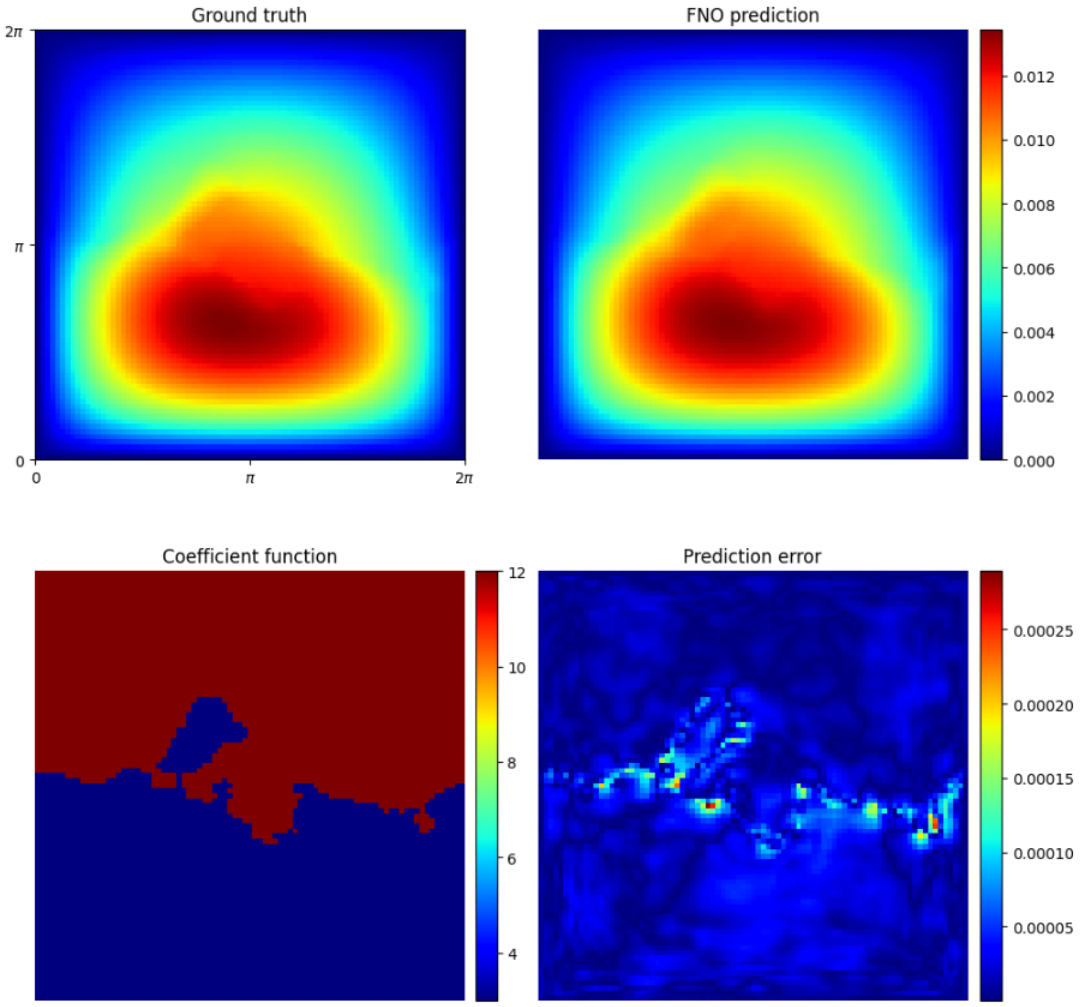


Figure 5. Dataset 1 best sample. Top left: Ground truth solution (output) to the PDE. Top right: Prediction by the FNO model. Bottom left: Coefficient function (input) to the PDE. Bottom right: Pointwise prediction error between the ground truth and the FNO prediction

Finally, Figure 6 shows the total bias and standard deviation across the entire test set. The bias stays relatively consistent across the range, suggesting a minor tendency of the model to underestimate. The standard deviation rises slightly in areas with abrupt coefficient changes, indicating greater local uncertainty where the solution is more responsive. Overall, these findings demonstrate the solid performance of the FNO model on this dataset, achieving better accuracy in predictions on samples where the coefficient field is more orderly and geometrically straightforward.

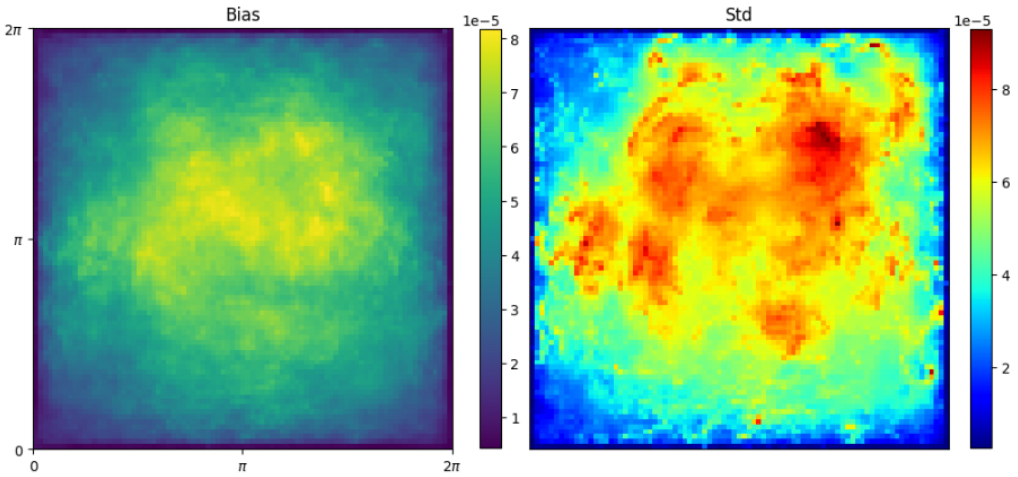


Figure 6. Dataset 1. Left: Marginal bias (mean absolute error) over the test samples. Right: Marginal standard deviation of the absolute error across the test samples.

Modified version of the original dataset

The mean relative error for this dataset is 1.8582×10^{-2}. Table 2 shows the five examples with the smallest and largest relative L^2 errors for Dataset 2. The most accurate predictions have very low errors, all less than 6.7190×10^{-3}, demonstrating that the model effectively applies to several test cases with impressive accuracy. Conversely, the least accurate predictions have much higher relative errors, especially sample 84, which shows a relative error of 1.1963×10^{-1}. This is more than ten times larger than the errors of the best predictions.

This difference implies that, although the FNO captures the overall mapping proficiently, certain examples prove tougher to handle. These challenging cases may be associated with areas having abrupt interfaces, intricate spatial arrangements, or varying patterns in the coefficient function $a(x)$ that are tough to learn. Since the dataset features a stark contrast between the thresholded values of $a(x)$, with values of 100 and 10, it is likely that significant changes in the coefficients intensify the complexity of the resulting solution $u(x)$, reducing the model's precision in specific extreme cases.

Table 2. Top 5 indices with the best and worst predictions: Dataset 2

Best index	**RelL2**	**Worst index**	**RelL2**
73	5.5932×10^{-3}	84	1.1963×10^{-1}
58	6.4034×10^{-3}	57	5.4849×10^{-2}
7	6.5771×10^{-3}	44	5.3269×10^{-2}
43	6.5969×10^{-3}	97	5.2807×10^{-2}
14	6.7189×10^{-3}	8	4.5981×10^{-2}

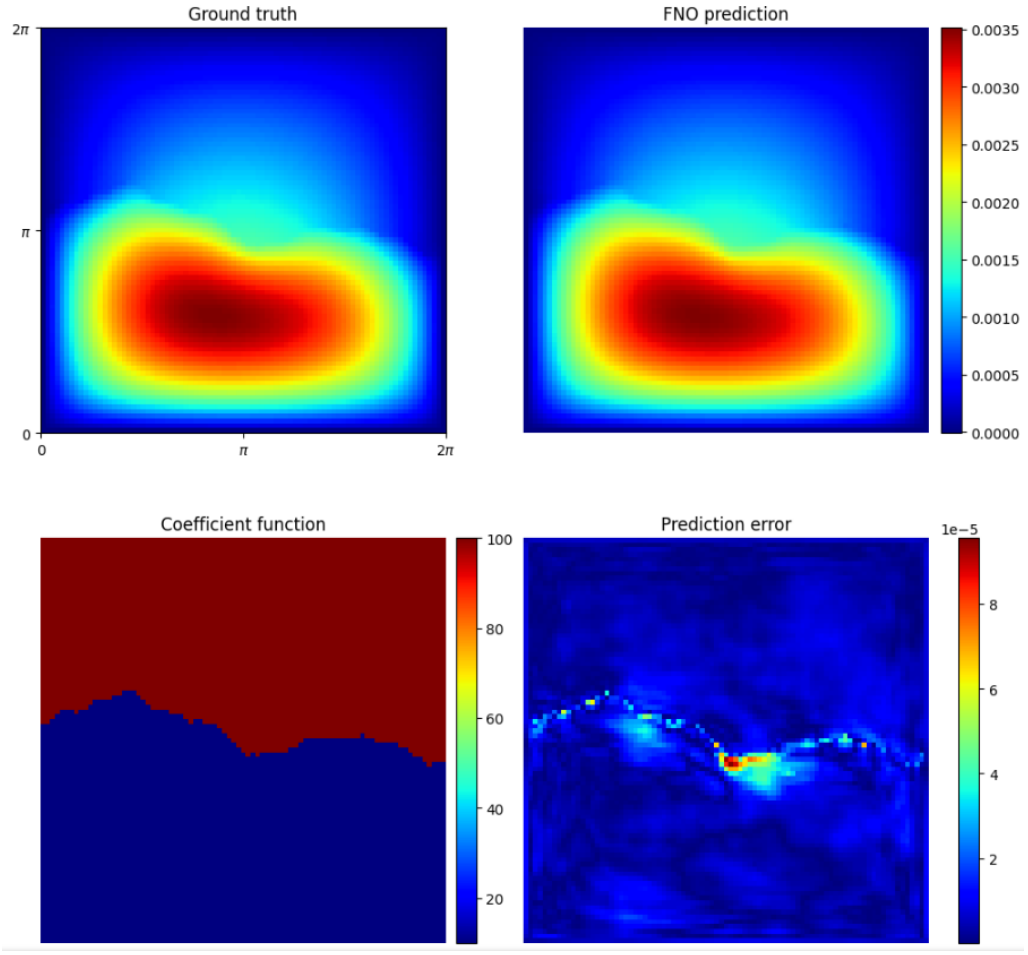


Figure 7. Dataset 2 best sample. Top left: Ground truth solution (output) to the PDE. Top right: Prediction by the FNO model. Bottom left: Coefficient function (input) to the PDE. Bottom right: Pointwise prediction error between the ground truth and the FNO prediction

Figure 7 showcases the top-performing forecast from Dataset 2. In this scenario, the coefficient layout is straightforward, featuring a distinct horizontal boundary dividing two consistent areas. This structural simplicity seems to be advantageous for the model: the

FNO forecast is nearly identical to the ground truth, without visible distortion or anomalies. The individual errors are minimal, primarily localized along the boundary of the coefficients, which is typical in PDE solutions with abrupt changes. This demonstrates that when the coefficient's structure is smooth and harmoniously fits the domain, the model can achieve highly precise results.

Conversely, Figure 8 highlights the poorest-performing forecast. Here, the coefficient function is more intricate and uneven, characterized by sharp, disjoint transitions without a clear geometric arrangement. Although the FNO manages to capture the general shape of the solution, it falls short in depicting finer details, particularly in areas with rapid changes in coefficients. The individual errors are higher in terms of both size and spread, concentrated mostly around discontinuities but also found in more stable zones, suggesting a broader difficulty with this setup. This underscores how irregular geometric coefficients can pose tougher forecasting challenges, even in the absence of minor-scale noise.

Figure 9 gives a statistical summary of the model's performance across the entire test set. The marginal bias stays near zero throughout the domain, confirming that the model does not consistently overestimate or underestimate. The variation in absolute error is generally low but slightly increased in areas typically containing coefficient boundaries, where the solution shows greater sensitivity. This signals the model's heightened uncertainty near abrupt changes, a common limitation for many PDE solvers, whether data-driven or otherwise.

These insights emphasize how the model's accuracy in this dataset is influenced by the geometric intricacy of the input coefficients.

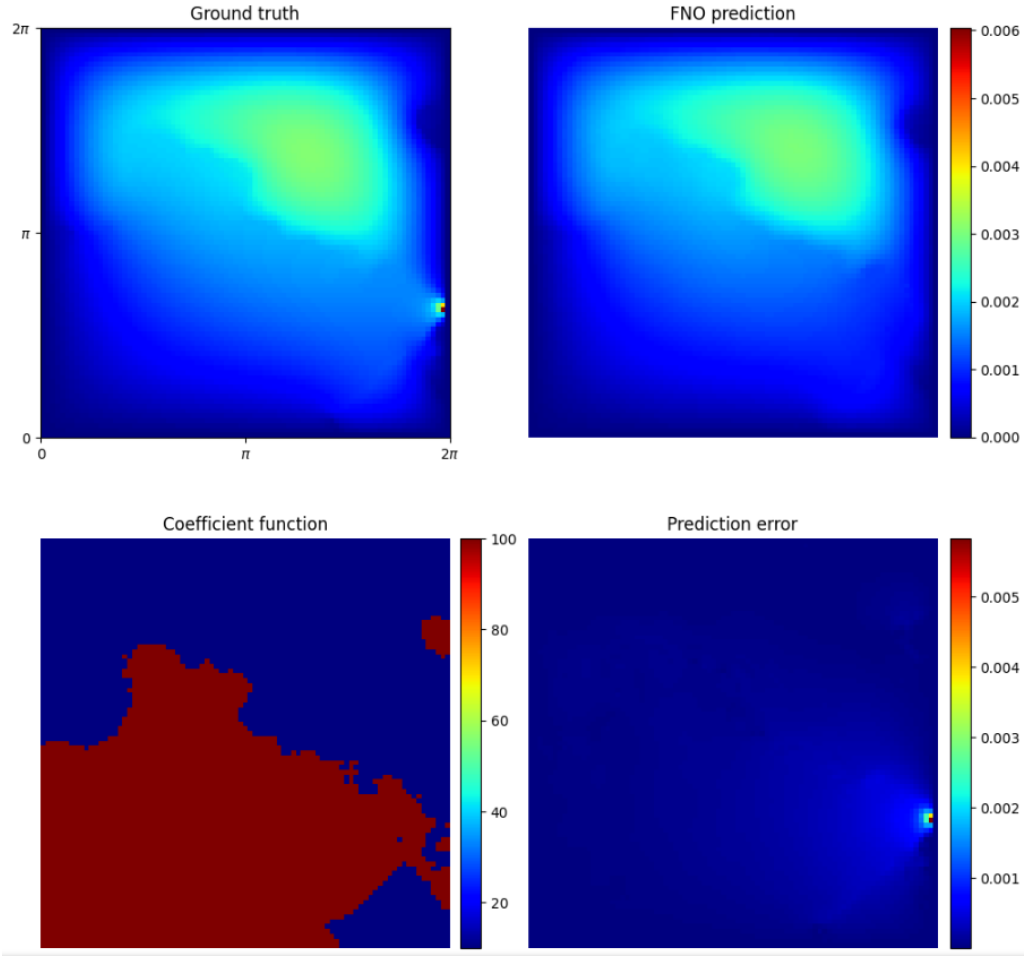


Figure 8. Dataset 2 worst sample. Top left: Ground truth solution (output) to the PDE. Top right: Prediction by the FNO model. Bottom left: Coefficient function (input) to the PDE. Bottom right: Pointwise prediction error between the ground truth and the FNO prediction

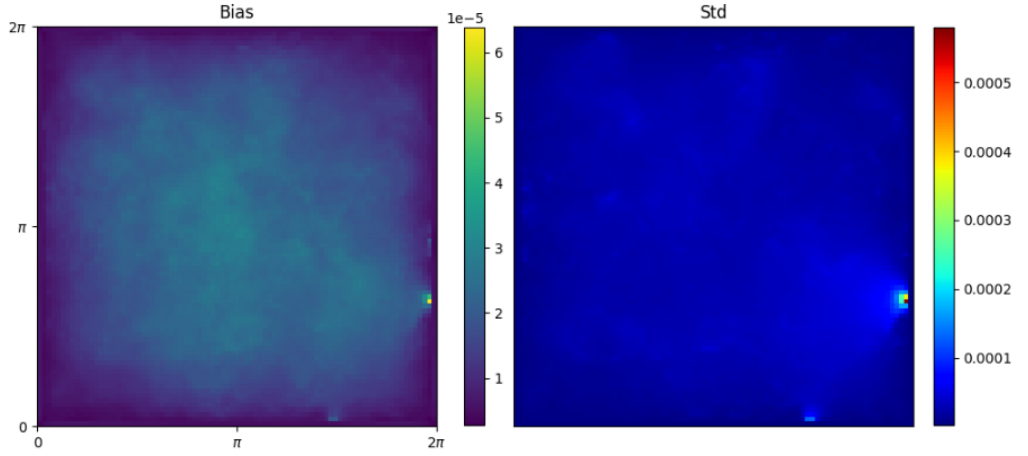


Figure 9. Dataset 2. Left: Marginal bias (mean absolute error) over the test samples. Right: Marginal standard deviation of the absolute error across the test samples.

Exponentiated Gaussian random fields

The average relative error obtained for this dataset is 4.6534×10^{-4}. Table 3 shows that the model achieves remarkably low relative L^2 errors across all test samples, with the best error being 2.9286×10^{-4} and the worst only 1.0220×10^{-3}. The small range and uniformly low error values reflect the fact that the learning task is significantly easier in this setting. The smoothness and continuity of the exponential coefficients likely reduce the complexity of the solution operator, making it easier for the FNO to approximate.

Compared to Datasets 1 and 2 with thresholded coefficients, the errors here are much smaller and more uniform. This suggests that abrupt transitions and high-contrast discontinuities (e.g., values jumping from 3 to 12 or 10 to 100) present greater challenges for learning, while smooth coefficient profiles allow for better generalization and stability.

Table 3. Top 5 indices with the best and worst predictions: Dataset 3

Best index	RelL2	Worst index	RelL2
75	2.9286×10^{-4}	10	1.0220×10^{-3}
16	2.9287×10^{-4}	44	8.7044×10^{-4}
69	3.0469×10^{-4}	95	8.5785×10^{-4}
8	3.0628×10^{-4}	13	8.2323×10^{-4}
71	3.1023×10^{-4}	90	7.8706×10^{-4}

Figure 10 highlights the sample with the poorest prediction outcome. In the top row, you can see the actual results compared with FNO's prediction, displaying a similar smooth and symmetrical pattern, which suggests that the model successfully mirrors the overall shape of the solution. On the flip side, the bottom row unveils a complex coefficient function characterized by high spatial variability, especially noticeable in the upper left and center-left areas due to sudden changes. This complexity likely results in higher prediction errors evident in the associated error map, where the error isn't spread evenly but instead shows concentrated peaks in certain areas, indicating regions where the model struggles to accurately capture the solution due to the irregular input coefficients.

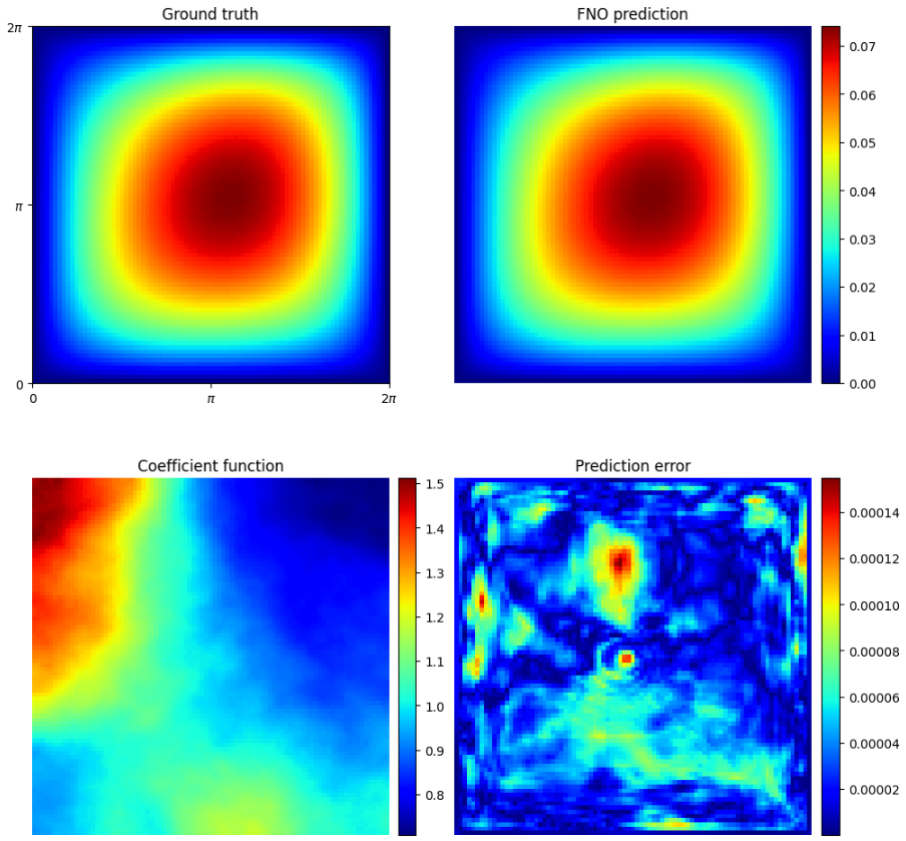


Figure 10. Dataset 3 worst sample. Top left: Ground truth solution (output) to the PDE. Top right: Prediction by the FNO model. Bottom left: Coefficient function (input) to the PDE. Bottom right: Pointwise prediction error between the ground truth and the FNO prediction

In contrast, Figure 11 showcases the sample with the best prediction. Once again, the top row reflects a strong alignment between the actual data and the FNO's output. This time around, the coefficient function, seen in the bottom-left panel, looks smoother and more uniformly distributed compared to the prior sample, although it still presents moderate variability. The resulting prediction error stands at a notably lower level and is more evenly spread, showing only a tiny peak near the center. This signifies that the FNO efficiently manages inputs with moderate variability, leading to an accurate and trustworthy prediction.

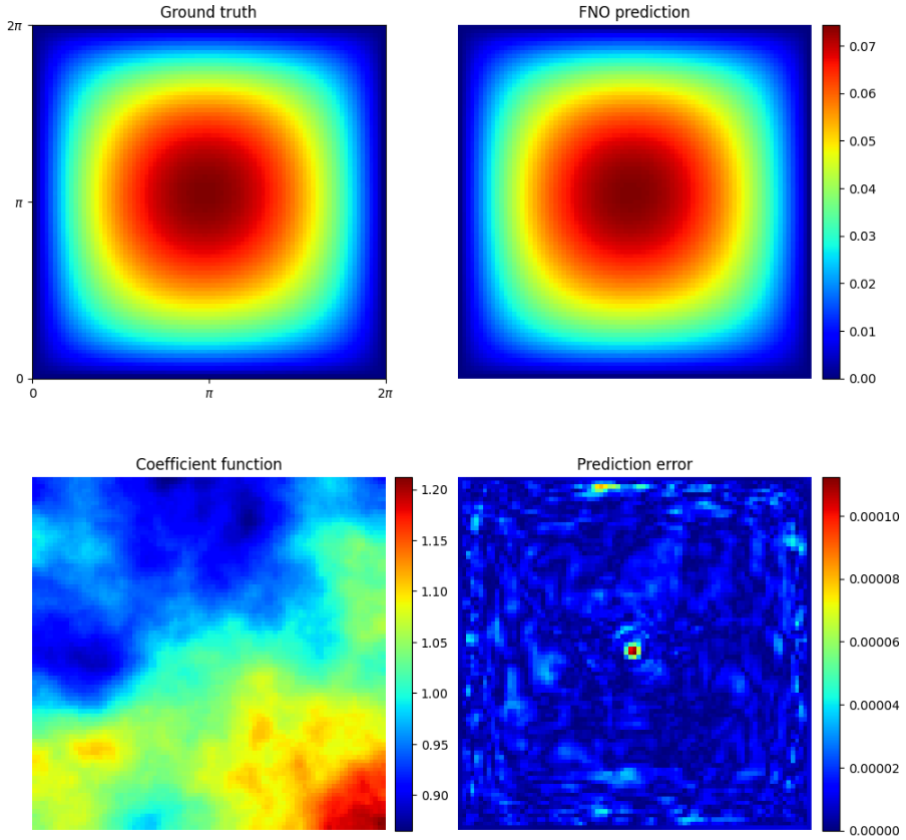


Figure 11. Dataset 3 best sample. Top left: Ground truth solution (output) to the PDE. Top right: Prediction by the FNO model. Bottom left: Coefficient function (input) to the PDE. Bottom right: Pointwise prediction error between the ground truth and the FNO prediction

Figure 12 offers an overview of the model's general performance by illustrating the pixel-wise bias and standard deviation of the prediction errors throughout the test set. The bias remains very minimal and is concentrated at the center, indicating that, on average, the model avoids consistently over- or under-predicting across most areas. Although small, the standard deviation is somewhat higher in the center and along the edges, suggesting a slight increase in prediction variability in these regions, possibly due to the model's sensitivity to specific patterns in the input coefficients.

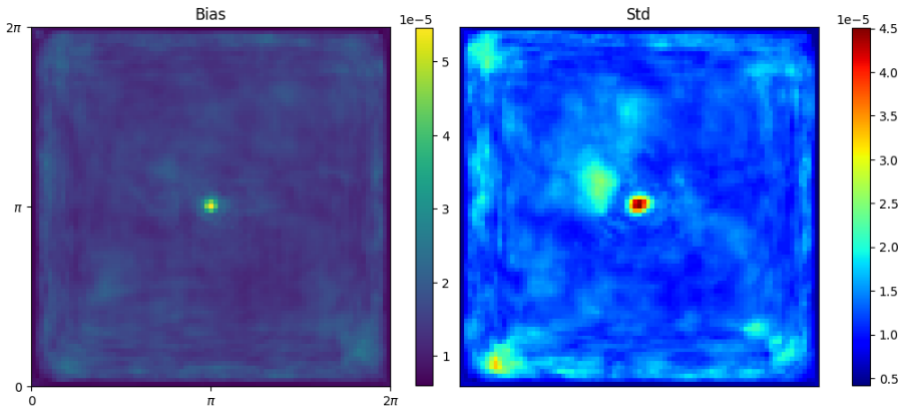


Figure 12. Dataset 3. Left: Marginal bias (mean absolute error) over the test samples. Right: Marginal standard deviation of the absolute error across the test samples.

5 DISCUSSION

5.1 Current study

This research aimed to investigate how well Fourier Neural Operators (FNOs) can be applied to and perform in solving the Darcy flow problem. The study was organized around four key research questions, each addressed based on the experimentation outcomes and their analysis.

Firstly, regarding the use of FNOs for modeling steady-state fluid movement in porous materials, the research effectively showed that FNOs can learn to map input coefficients (permeability fields) to pressure solutions governed by the Darcy equation. For all datasets, the FNO was trained to approximate the solution operator without needing to know the PDE structure or discretization approach during inference. The architecture was developed and trained exclusively with data generated from traditional methods, particularly the finite difference method, underscoring the data-driven nature of this approach.

Secondly, the theoretical foundation of FNOs was reviewed and discussed in Chapter 3. The model's ability to remain consistent across different grid resolutions was leveraged in this study, even as training and testing data shared the same resolution; the operator was learned at the function level. The universal approximation theorem, as presented in the fundamental work by Kovachki et al. [15], supports the model's ability to approximate continuous solution operators between infinite-dimensional function spaces, providing a theoretical basis for the performance observed.

Thirdly, in terms of prediction accuracy, a detailed evaluation was performed on three different datasets, each characterized by distinct statistical and geometric properties of the coefficient fields:

- **Dataset 1** consisted of thresholded Gaussian random fields with binary values, leading to distinct but regular interfaces. The FNO showed consistent performance with relative L^2 errors between 3.9×10^{-3} and 2.7×10^{-2}, showing moderate sensitivity to spatial gaps but generally high prediction reliability.
- **Dataset 2** also involved thresholded Gaussian random fields but with a higher contrast between binary values, creating spatially smooth but non-uniform patterns that resulted in more performance variability. The highest error reached 1.2×10^{-1}, significantly higher than in Dataset 1, while the best performance stayed below

5.6×10^{-3}. Visual assessments revealed that irregular coefficient shapes challenge the model's generalization, particularly when faced with sharp transitions or detailed elements.

- **Dataset 3** was created from exponentiated Gaussian random fields, resulting in smooth, continuous spatial patterns. In this scenario, the FNO performed exceptionally well, with the highest prediction error below 1.1×10^{-3}. The overall smoothness and lack of abrupt changes in input appeared to make learning easier, as evident from the minimal errors, bias, and variance.

Throughout all datasets, the model exhibited low bias and standard deviation in smooth areas, with most errors occurring near coefficient discontinuities. Statistical visualizations in Figure 9 and others for Datasets 1 and 3 revealed that model uncertainty is concentrated around sensitive transitions, highlighting a well-known limitation in both numerical and learning-based PDE solvers. The results suggest that increasing contrast in the coefficient function lead to reduced accuracy. In particular, for extreme threshold values such as 100 and 3, the learning process breaks down with errors exceeding 58%.

In conclusion, this research's main contribution is an in-depth empirical analysis of FNOs applied to the Darcy flow problem, demonstrating how the geometric nature of input coefficients notably affects model precision. The entire research process, including all experiments, analysis, and adaptation of code, was carried out independently by the author. While the foundational implementation of the FNO architecture is based on publicly available code from established GitHub repositories, these implementations were thoroughly modified and adapted to meet the specific needs of this project. The thesis does not claim authorship of the original codebases but focuses on the originality of their adaptation and the fresh empirical insights obtained.

5.2 Future work

Several promising pathways emerge from this research. Firstly, expanding the experiments to include noisy or multi-scale coefficients would enhance our understanding of the model's resilience in real-world conditions, particularly for geophysical or biomedical applications. Secondly, explicitly testing the discretization-invariance property by training and evaluating across various resolutions could further confirm the theoretical strengths of FNOs.

Another logical expansion would involve incorporating uncertainty quantification techniques, like Bayesian neural operators or ensemble approaches, to provide confidence measures alongside predictions. Additionally, hybrid approaches that blend data-driven models like FNOs with physics-informed constraints or traditional solvers could boost reliability, especially in challenging scenarios.

Lastly, this study concentrated on the two-dimensional steady-state Darcy problem. Examining time-dependent or three-dimensional scenarios would widen the scope and assess the scalability of FNOs for more complex PDE systems.

6 CONCLUSION

This thesis explored how Fourier Neural Operators (FNOs) can be used to tackle the two-dimensional steady-state Darcy flow problem. The study aimed at appreciating the suitability of FNOs for modeling complicated fluid motions through porous materials and assessing both their theoretical and practical capabilities.

The FNO model was set up and trained on synthetic datasets using a finite difference method. It was able to precisely map input coefficient fields to pressure solutions without needing direct information about the underlying PDE or numerical method during prediction time. The model's effectiveness was measured using three different datasets, each featuring unique structural and statistical properties, which highlighted a clear link to the complexity of the input.

Interestingly, the FNO showed optimal performance when the input coefficients presented structured or smooth patterns, whereas its accuracy dropped when encountering geometric inconsistencies or sharp changes. These results are consistent with the theoretical claims made by Li et al. [7], illustrating that FNOs have a strong ability to express and can successfully approximate solution operators for PDEs, especially when input functions are well-structured. This is further supported by the consistently low relative errors along with minimal bias and limited variability in predictions in most scenarios.

In summary, this research showed that FNOs are a promising data-driven approach for solving elliptic PDEs like the Darcy flow, particularly useful in applications needing repeated assessments. The findings serve as a groundwork for future research looking to expand the technique to more intricate, high-dimensional, or dynamic systems.

REFERENCES

[1] Alfio Quarteroni, Riccardo Sacco, and Fausto Saleri. *Numerical mathematics*, volume 37. Springer Science & Business Media, 2010.

[2] Lawrence C Evans. *Partial differential equations*, volume 19. American Mathematical Society, 2022.

[3] Bing Yu et al. The deep ritz method: A deep learning-based numerical algorithm for solving variational problems. *Communications in Mathematics and Statistics*, 6(1):1–12, 2018.

[4] Yinhao Zhu, Nicholas Zabaras, Phaedon-Stelios Koutsourelakis, and Paris Perdikaris. Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data. *Journal of Computational Physics*, 394:56–81, 2019.

[5] Tianping Chen and Hong Chen. Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems. *IEEE transactions on neural networks*, 6(4):911–917, 1995.

[6] Lu Lu, Pengzhan Jin, and George Em Karniadakis. Deeponet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. *arXiv preprint arXiv:1910.03193*, 2019.

[7] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

[8] Samuel Lanthaler, Siddhartha Mishra, and George E Karniadakis. Error estimates for deeponets: A deep learning framework in infinite dimensions. *Transactions of Mathematics and Its Applications*, 6(1):tnac001, 2022.

[9] 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. *Journal of Machine Learning Research*, 24(89):1–97, 2023.

[10] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Andrew Stuart, Kaushik Bhattacharya, and Anima Anandkumar. Multipole graph neural operator for parametric partial differential equations. *Advances in Neural Information Processing Systems*, 33:6755–6766, 2020.

[11] Gaurav Gupta, Xiongye Xiao, and Paul Bogdan. Multiwavelet-based operator learning for differential equations. *Advances in neural information processing systems*, 34:24048–24062, 2021.

[12] Md Ashiqur Rahman, Zachary E Ross, and Kamyar Azizzadenesheli. U-no: U-shaped neural operators. *arXiv preprint arXiv:2204.11127*, 2022.

[13] Kalpesh Sanjay More, Tapas Tripura, Rajdip Nayek, and Souvik Chakraborty. A bayesian framework for learning governing partial differential equation from data. *Physica D: Nonlinear Phenomena*, 456:133927, 2023.

[14] Somdatta Goswami, Aniruddha Bora, Yue Yu, and George Em Karniadakis. Physics-informed deep neural operator networks. In *Machine Learning in Modeling and Simulation: Methods and Applications*, pages 219–254. Springer, 2023.

[15] Nikola Kovachki, Samuel Lanthaler, and Siddhartha Mishra. On universal approximation and error bounds for fourier neural operators. *Journal of Machine Learning Research*, 22(290):1–76, 2021.

[16] Sung Woong Cho, Jae Yong Lee, and Hyung Ju Hwang. Learning time-dependent pde via graph neural networks and deep operator network for robust accuracy on irregular grids. *arXiv preprint arXiv:2402.08187*, 2024.

[17] Jin Young Shin, Jae Yong Lee, and Hyung Ju Hwang. Pseudo-differential neural operator: Generalized fourier neural operator for learning solution operators of partial differential equations. *arXiv preprint arXiv:2201.11967*, 2024.

[18] Haim Brézis. *Functional analysis, Sobolev spaces and partial differential equations*, volume 2. Springer, 2011.

[19] Gerald B Folland. *Real analysis: modern techniques and their applications*. John Wiley & Sons, 1999.

[20] Volodymyr Brayman, Andrii Chaikovskyi, Oleksii Konstantinov, Alexander Kukush, Yuliya Mishura, and Oleksii Nesterenko. *Functional Analysis and Operator Theory*. Springer, 2024.

[21] M Mursaleen and Abdullah K Noman. On some new sequence spaces of non-absolute type related to the spaces ℓ^p and ℓ^∞ i. *Filomat*, 25(2):33–51, 2011.

[22] Robert E Megginson. *An introduction to Banach space theory*, volume 183. Springer Science & Business Media, 2012.

[23] Eric W. Weisstein. Convolution theorem. MathWorld–A Wolfram Web Resource, n.d. Accessed: 2025-04-04.

[24] Haim Brézis. *Functional Analysis, Sobolev Spaces and Partial Differential Equations*, chapter 8, pages 233–234. Springer, 2011.

[25] Andrew R Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information theory*, 39(3):930–945, 1993.

[26] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. `https://github.com/scaomath/fourier_neural_operator`, 2020. Accessed: 2025-03-01.