Spectral vs. Fourier Neural Operators in Parametric PDE Modeling: Analysis and Experiments

Jizhou Guo
Zhiyuan College, Shanghai Jiao Tong University
sjtu18640985163@sjtu.edu.cn

Abstract

We compare Fourier Neural Operators (FNOs) and Spectral Neural Operators (SNOs) for learning solution operators of parameterized partial differential equations. While FNOs leverage truncated Fourier transforms for mesh-invariant convolutions and satisfy *universal approximation* property, they suffer from aliasing errors, impeding its performance in certain scenarios. SNOs instead expand inputs and outputs in a fixed global spectral basis, eliminating aliasing and yielding directly interpretable spectral coefficients. We prove universal approximation results for FNOs, analyze the sources of aliasing in FNOs, and describe how SNOs avoid these issues. Two methods are benchmarked on Burgers, KdV and KS equations. Our source code for numerical experiments is available at https://github.com/aster2024/SNO_vs_FNO.

1 Introduction

Many applications in science and engineering - ranging from aerodynamics design to subsurface flow modeling - require repeated solves of parameterized partial differential equations (PDEs) at high resolution. Classical numerical methods such as finite difference, finite element, or spectral solvers achieve high accuracy but incur steep computational costs when employed in inverse design, uncertainty quantification, or real-time control loops.

Data-driven surrogate models have emerged as a compelling alternative: once trained, they can evaluate the solution operator in milliseconds, enabling rapid parameter sweeps and real-time decision making. However, standard neural networks map between finite-dimensional discretizations and suffer from mesh dependence, lack of resolution invariance, and inability to query arbitrary points in the domain [Kovachki et al., 2021].

Neural operators [Anandkumar et al., 2020, Li et al., 2020] directly learn mappings between infinite-dimensional function spaces. The Fourier Neural Operator (FNO) Li et al. [2021] replaces integral kernels with convolutions in the Fourier domain, leveraging the FFT for quasi-linear complexity and demonstrating zero-shot super-resolution. Yet FNOs introduce a critical limitation: Nonlinear activations generate high-frequency harmonics that fold back into the truncated spectrum, leading to inconsistent multi-grid evaluations and reduced accuracy in high-precision regimes [Fanaskov and Oseledets, 2023, Bartolucci et al., 2023].

Spectral Neural Operators (SNOs) [Fanaskov and Oseledets, 2023] address such issue by expanding both input and output functions in a chosen global basis (e.g. Chebyshev or Fourier polynomials). The network then learns a mapping on the coefficient vectors, ensuring exact alias-free operations.

In this work, we carry out a systematic theoretical and empirical comparison of FNOs and SNOs:

- We illustrate the architecture of FNO, and prove universal approximation theorems for FNOs.

Course Project for MATH3550: Numerical Methods for Ordinary and Partial Differential Equations.

- We characterize the aliasing mechanisms in FNO layers by theoretical analysis and numerical experiments, and show that SNOs avoid such issue.
- Through numerical experiments on Burgers' equation, KdV equation and KS equation, we evaluate the performance of FNOs and SNOs.

2 Related Work

This section outlines the landscape of numerical PDE solvers, contrasting traditional methods with emerging data-driven approaches. We then delve into specific neural network-based strategies for PDE solving, highlighting their strengths and limitations, with a particular emphasis on the advantages offered by neural operators.

Traditional Solvers vs. Data-Driven Methods Conventional solvers, such as Finite Element Methods (FEM) Grossmann [2007] and Finite Difference Methods (FDM) Ciarlet [2002], Brenner [2008], discretize the spatial domain to approximate solutions to PDEs. These methods inherently involve a trade-off between computational cost and accuracy. Coarse discretizations offer faster computation but compromise accuracy, while fine discretizations provide higher accuracy at the expense of increased computational time. Complex PDE systems often necessitate very fine discretizations, posing significant computational challenges for traditional solvers. In contrast, data-driven methods offer the potential to learn the solution manifold directly from data, enabling significantly faster solution times compared to conventional solvers [Raissi et al., 2019, Kochkov et al., 2021]. Machine learning techniques are increasingly viewed as a key to revolutionizing scientific computing by providing fast, accurate approximations or enhancements to traditional solvers. However, many early machine learning approaches were limited by their dependence on specific discretizations.

Finite-Dimensional Operators These methods parameterize the solution operator as a deep convolutional neural network mapping between finite-dimensional Euclidean spaces Guo et al. [2016], Adler and Öktem [2017], Bhatnagar et al. [2019]. A major limitation is their mesh-dependent nature, requiring modifications and tuning for different resolutions and discretizations to achieve consistent error. Furthermore, they are restricted to the discretization size and geometry of the training data, preventing querying solutions at new points within the domain.

Neural-FEM This approach directly parameterizes the solution function as a neural network [Bar and Sochen, 2019, Raissi et al., 2019, Pan and Duraisamy, 2020]. While mesh-independent and accurate, Neural-FEM is designed to model only a specific instance of the PDE, rather than the general solution operator. Each new instance of the functional parameter or coefficient requires training a new neural network. This approach shares computational challenges with classical methods, as the optimization problem must be solved for every new instance. Moreover, it is limited to scenarios where the underlying PDE is known.

Neural Operators Neural operators represent a recent advancement, enabling the learning of mesh-free, infinite-dimensional operators using neural networks [Anandkumar et al., 2020, Li et al., 2020]. These operators overcome the mesh-dependent limitations of finite-dimensional operator methods by providing a single set of network parameters applicable to different discretizations, facilitating solution transfer between meshes. Furthermore, neural operators require training only once; obtaining solutions for new parameter instances involves only a forward pass through the network, mitigating the computational burden associated with Neural-FEM. Importantly, neural operators require no prior knowledge of the underlying PDE, relying solely on data. However, a challenge remains in developing efficient numerical algorithms for neural operators that can match the performance of convolutional or recurrent neural networks in finite-dimensional settings, primarily due to the cost of evaluating integral operators. This work addresses this challenge by leveraging the fast Fourier transform and spectral decomposition.

3 Problem Statement

Let $D \subset \mathbb{R}^d$ be a bounded, open domain and consider two separable Banach spaces

$$\mathcal{A} = \mathcal{A}(D; \mathbb{R}^{d_a}), \quad \mathcal{U} = \mathcal{U}(D; \mathbb{R}^{d_u}),$$

whose elements are parameter functions $a\colon D \to \mathbb{R}^{d_a}$ and state functions $u\colon D \to \mathbb{R}^{d_u}$, respectively. We assume there exists an (unknown) solution operator

$$\mathcal{F}^*\colon \mathcal{A} \longrightarrow \mathcal{U},$$

typically arising from a parameterized partial differential equation (PDE), so that for each $a \in \mathcal{A}$, the corresponding solution is $u = \mathcal{F}^*(a)$.

In practice, we observe a finite dataset

$$\{(a_j,\, u_j)\}_{j=1}^{N}, \quad a_j \sim \mu \quad \text{and} \quad u_j \approx \mathcal{F}^*(a_j),$$

where μ is an (unknown) probability measure on $\mathcal{A}$ and each u_j may be corrupted by noise. Our goal is to construct a parametric family of mappings

$$\mathcal{F}_\theta\colon \mathcal{A} \longrightarrow \mathcal{U}, \quad \theta \in \Theta \subset \mathbb{R}^p,$$

such that $\mathcal{F}_\theta \approx \mathcal{F}^*$ in the sense of minimizing the expected discrepancy

$$\min_{\theta\in\Theta} \mathbb{E}_{a\sim\mu}\big[\,\mathcal{L}\big(\mathcal{F}_\theta(a),\, \mathcal{F}^*(a)\big)\big],$$

for a suitable loss functional $\mathcal{L}\colon \mathcal{U} \times \mathcal{U} \to \mathbb{R}$ (e.g. an L^2 error).

Unlike classical PDE solvers or physics-informed networks, which solve one instance of the PDE at a time, our aim is to *learn the entire operator* $\mathcal{F}^*$ so that a single forward evaluation of $\mathcal{F}_\theta$ yields the solution u for any new a. This shift from per-instance solution to operator approximation offers substantial savings when multiple queries are needed.

Since both inputs and outputs are functions, we only have access to their pointwise evaluations on finite grids

$$D_j = \{x_{j,1}, \ldots, x_{j,n_j}\} \subset D, \quad a_j\big|_{D_j} \in \mathbb{R}^{n_j \times d_a}, \quad u_j\big|_{D_j} \in \mathbb{R}^{n_j \times d_u}.$$

A key requirement is *discretization-invariance*: once trained, $\mathcal{F}_\theta$ should accept data on any grid $D' \subset D$, possibly of different resolution or geometry, and produce a consistent approximation of the continuum solution. This property enables *zero-shot transfer* across meshes without retraining.

In summary, our problem is to design a learnable, finite-parameter mapping $\mathcal{F}_\theta$ that

- approximates the true operator $\mathcal{F}^*$ in the infinite-dimensional setting,
- is trained on a finite collection of discretized input–output pairs, and
- yields mesh-independent predictions on arbitrary point-sets in D.

4 Fourier Neural Operator

We describe the architecture of Fourier Neural Operator (FNO) [Li et al., 2021] for learning operators mapping coefficient functions $a(\cdot)$ to solution functions $u(\cdot)$ in parameterized PDEs.

Let $D \subset \mathbb{R}^d$ be the computational domain and define the input and output spaces

$$\mathcal{A} = \{\, a : D \to \mathbb{R}^{d_a}\}, \qquad \mathcal{U} = \{\, u : D \to \mathbb{R}^{d_u}\}.$$

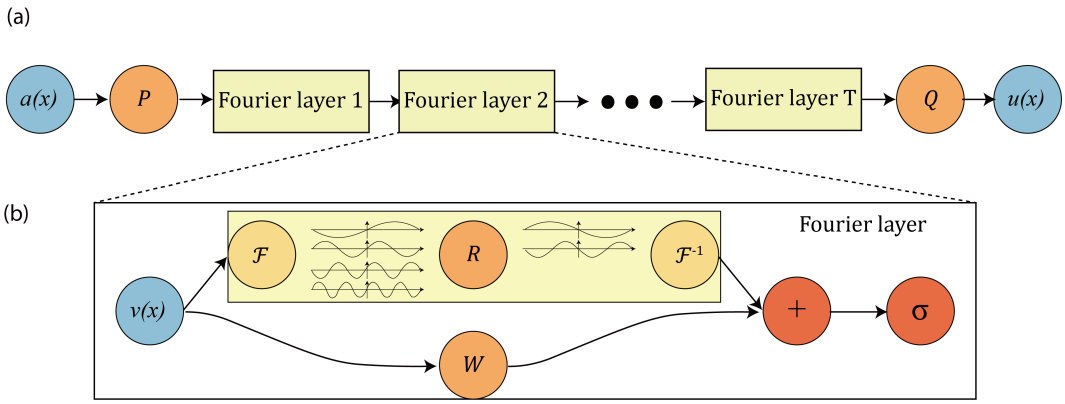
Our goal is to approximate the true solution operator $\mathcal{F}^* : \mathcal{A} \to \mathcal{U}$, $u = \mathcal{F}^*(a)$, from data $\{(a_j, u_j)\}$. We introduce a trainable map $\mathcal{F}_\theta$ defined by the following iterative procedure. First, a pointwise lift

$$v_0(x) = P\big(a(x)\big) \in \mathbb{R}^{d_v} \tag{4.1}$$

encodes the input into a higher-dimensional feature space via a small fully-connected network P. Then for $t = 0, \ldots, T-1$ we update[1]

$$v_{t+1}(x) \;=\; \sigma\Big(W_t\, v_t(x) \;+ b_t(x) +\; (\mathcal{K}_t v_t)(x)\Big), \tag{4.2}$$

[1] In the original paper of Li et al. [2021], there is actually no bias term $b_t(x)$. However the bias term exists in Kovachki et al. [2021], and to support our theoretical study in Section 5, we keep the bias term here.



(a) FNO Architecture: Starting from input a, the model: 1. Lifts a to a higher-dimensional channel space using a neural network P. 2. Applies four layers of integral operators and activation functions sequentially. 3. Projects back to the target dimension via a neural network Q, resulting in output u. **(b) Fourier Layer Details**: Beginning with input v: The upper branch applies the Fourier transform $\mathcal{G}$, performs a linear transform R on the lower Fourier modes while filtering out higher modes, and then applies the inverse Fourier transform $\mathcal{G}^{-1}$. The lower branch applies a local linear transform W.

Figure 1: **top:** FNO Architecture; **bottom:** Fourier Layer Details.

where $W_t \in \mathbb{R}^{d_v \times d_v}$ and $b_t(x)$ define a pointwise affine mapping (corresponding to weights and biases), σ is a pointwise activation (e.g. ReLU), and $\mathcal{K}_t$ is a global linear operator. Finally, the solution is recovered by the projection

$$u(x) = Q\big(v_T(x)\big), \tag{4.3}$$

with $Q : \mathbb{R}^{d_v} \to \mathbb{R}^{d_u}$ another small network. The architecture is illustrated in Figure 1 (a).

In the Fourier Neural Operator we choose $\mathcal{K}_t$ to be a convolution with a translation-invariant kernel $\kappa_t : \mathbb{R}^d \to \mathbb{R}^{d_v \times d_v}$. By the convolution theorem,

$$(\mathcal{K}_t v)(x) = \int_D \kappa_t(x-y)\, v(y)\, \mathrm{d}y = \mathcal{F}^{-1}\big(\widehat{\kappa}_t \cdot \widehat{v}\big)(x), \tag{4.4}$$

where $\widehat{v}(k) = \int_D v(x) e^{-2\pi i\, k\cdot x}\, \mathrm{d}x$. We parameterize only the low-frequency multipliers $\widehat{\kappa}(k) \in \mathbb{C}^{d_v \times d_v}$ for $|k_j| \le K_j$, truncating all higher modes to zero. Concretely, on a uniform discretization of D with $n_1 \times \cdots \times n_d$ points we compute

$$\widehat{v}_t = \mathrm{FFT}(v_t), \quad \widehat{v}'_t(k) = R(k)\, \widehat{v}_t(k), \quad v'_t = \mathrm{FFT}^{-1}(\widehat{v}'_t),$$

where $R(k) \equiv \widehat{\kappa}(k)$ for $|k_j| \le K_j$ and FFT denotes Fast Fourier Transform [Cooley and Tukey, 1965]. We provide an illustration for Fourier Layer in Figure 1 (b). This Fourier layer incurs $O(n \log n + K\, d_v^2)$ cost per update, with $n = \prod_j n_j$ and $K = \prod_j (2K_j + 1)$, while requiring grids to be equispaced to apply FFT. Additionally, Lingsch et al. [2024] proposes a method that replaces FFTs with small matrix–vector products to compute truncated Fourier (or spherical-harmonic) transforms directly on arbitrary, non-equispaced point clouds, resulting in $O(mN)$ complexity, where m is the dimension of the computed Fourier modes.

***Remark* 4.1.** The most straightforward example of a FNO is constructed as follows: Let $\widehat{\mathcal{N}} : \mathbb{R}^{d_a} \to \mathbb{R}^{d_u}$ be a standard finite-dimensional neural network with activation function σ. We can associate with $\widehat{\mathcal{N}}$ the mapping $\mathcal{N} : L^2(\mathbb{T}^d; \mathbb{R}^{d_a}) \to L^2(\mathbb{T}^d; \mathbb{R}^{d_u})$, defined by $a(x) \mapsto \widehat{\mathcal{N}}(a(x))$. It is easy to see that $\mathcal{N}$ is a FNO, as it can be expressed in the form

$$\widehat{\mathcal{N}} = \widehat{\mathcal{Q}} \circ \widehat{\mathcal{L}}_L \circ \cdots \circ \widehat{\mathcal{L}}_1 \circ \widehat{\mathcal{R}},$$

where $\widehat{\mathcal{R}}(y) = Ry$ with $R \in \mathbb{R}^{d_v \times d_a}$, and each layer $\widehat{\mathcal{L}}_\ell$ is of the form $\widehat{\mathcal{L}}_\ell(y) = \sigma(W_\ell y + b_\ell)$ for some $W_\ell \in \mathbb{R}^{d_v \times d_v}$, $b_\ell \in \mathbb{R}^{d_v}$. Here, $\widehat{\mathcal{Q}}$ is an affine output layer of the form $\widehat{\mathcal{Q}}(y) = Qy + q$ with $Q \in \mathbb{R}^{d_u \times d_v}$, $q \in \mathbb{R}^{d_u}$. By replacing the input y with a function $v(x)$, these layers become a specific

instance of the FNO lifting layer (4.1), the non-linear layers (4.2) (with $P_\ell \equiv 0$ and constant bias $b_\ell(x) \equiv b_\ell$), and the projection layer (4.3). Therefore, any finite-dimensional neural network can be regarded as a FNO as defined above.

Because $R(k)$ and W are independent of the mesh spacing, the same trained parameters can be applied to any coarser or finer uniform grid without retraining, enabling *zero-shot super-resolution*. The FFT-based convolution captures global dependencies efficiently, while the pointwise activation σ injects nonlinearity, allowing the network to approximate highly non-linear solution operators. In addition, we showcase the universal property in Section 5, i.e., FNOs can approximate any continuous operator to desired accuracy.

5 Universal Approximation Properties of FNOs

We now establish that FNOs possess *universal approximation* capabilities [Kovachki et al., 2021], meaning that for any given class of operators, we can construct an FNO that approximates it with arbitrary precision. The framework for our analysis is presented below:

***Setting* 5.1.** Let $d \in \mathbb{N}$ be a fixed spatial dimension, and let $D \subset \mathbb{R}^d$ represent a domain in $\mathbb{R}^d$. We focus on approximating operators of the form $\mathcal{G} : \mathcal{A}(D; \mathbb{R}^{d_a}) \to \mathcal{U}(D; \mathbb{R}^{d_u})$, where $a \mapsto u := \mathcal{G}(a)$. Here, the input $a \in \mathcal{A}(D; \mathbb{R}^{d_a})$ with $d_a \in \mathbb{N}$ is a function $a : D \to \mathbb{R}^{d_a}$ having d_a components, while the output $u \in \mathcal{U}(D; \mathbb{R}^{d_u})$ with $d_u \in \mathbb{N}$ is a function $u : D \to \mathbb{R}^{d_u}$ having d_u components. The spaces $\mathcal{A}(D; \mathbb{R}^{d_a})$ and $\mathcal{U}(D; \mathbb{R}^{d_u})$ are assumed to be Banach spaces (or appropriate subsets thereof). Common examples include the space of continuous functions $C(D; \mathbb{R}^{d_u})$ and Sobolev spaces $H^s(D; \mathbb{R}^{d_u})$ of order $s \geq 0$. The definition of the notations is included in Appendix A.

We proceed to demonstrate that FNOs possess *universal approximation* properties. Specifically, for any broad class of operators as specified in Setting 5.1, one can construct an FNO that approximates the target operator to any desired level of accuracy. This result is formalized in the following theorem[2].

***Theorem* 5.2** (Universal approximation property)**.** Let $s, s' \geq 0$ be given. Consider a continuous operator $\mathcal{G} : H^s(\mathbb{T}^d; \mathbb{R}^{d_a}) \to H^{s'}(\mathbb{T}^d; \mathbb{R}^{d_u})$ and let $K \subset H^s(\mathbb{T}^d; \mathbb{R}^{d_a})$ be a compact subset. Then for any tolerance $\epsilon > 0$, there exists an FNO $\mathcal{N} : H^s(\mathbb{T}^d; \mathbb{R}^{d_a}) \to H^{s'}(\mathbb{T}^d; \mathbb{R}^{d_u})$ of the form (4.4), which is continuous as an operator $H^s \to H^{s'}$, such that

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

Proof. For notational simplicity, we assume $d_a = d_u = 1$ throughout; the extension to general $d_a, d_u > 1$ follows by analogous arguments. We first observe the following lemma, which is proved in Appendix B.1:

***Lemma* 5.3.** If the universal approximation Theorem 5.2 holds for $s' = 0$, it holds for arbitrary $s' \geq 0$.

We define an operator $\mathcal{G}_N$ that maps functions from the Sobolev space $H^s(\mathbb{T}^d)$ to $L^2(\mathbb{T}^d)$ as follows:

$$\mathcal{G}_N(a) := P_N \mathcal{G}(P_N a), \tag{5.1}$$

where P_N is the orthogonal Fourier projection operator defined in Appendix A (A.6). Essentially, $\mathcal{G}_N$ can be viewed as the Fourier projection of the operator $\mathcal{G}$.

Next, we can demonstrate that for any specified $\epsilon > 0$, there exists a natural number $N \in \mathbb{N}$ such that

$$\|\mathcal{G}(a) - \mathcal{G}_N(a)\|_{L^2} \leq \epsilon, \quad \forall\, a \in K. \tag{5.2}$$

[2] In practice, as evaluating the Fourier transform is impossible since it requires computing an integral exactly, we should compute discrete Fourier Transform between layers of the FNO instead. Its theoretical framework is introduced in Kovachki et al. [2021]

Consequently, the problem is reduced to finding a Fourier Neural Operator (FNO) (4.4) capable of approximating the operator $\mathcal{G}_N$ with arbitrary precision.

We define the set $\mathcal{K}_N$ containing Fourier wavenumbers k:

$$\mathcal{K}_N := k \in \mathbb{Z}^d \mid |k|_\infty \leq N, \tag{5.3}$$

and introduce the Fourier conjugate (or dual) operator $\widehat{\mathcal{G}}_N : \mathbb{C}^{\mathcal{K}_N} \to \mathbb{C}^{\mathcal{K}_N}$:

$$\widehat{\mathcal{G}}_N(\widehat{a}_k) := \mathcal{F}_N\left(\mathcal{G}_N\left(\mathrm{Re}\left(\mathcal{F}_N^{-1}(\widehat{a}_k)\right)\right)\right), \tag{5.4}$$

such that the following identity holds:

$$\mathcal{G}_N(a) = \mathcal{F}_N^{-1} \circ \widehat{\mathcal{G}}_N \circ \mathcal{F}_N(P_N a), \tag{5.5}$$

This identity is valid for all real-valued functions $a \in L^2(\mathbb{T}^d)$. Here, $\mathcal{F}_N$ and $\mathcal{F}_N^{-1}$ represent the discrete Fourier transform and its inverse, respectively, as defined in Appendix A (A.9) and (A.10).

The subsequent steps in the proof involve utilizing the decomposition of the projection $\mathcal{G}_N$ in (5.5), which expresses it in terms of the discrete Fourier transform $\mathcal{F}_N \circ P_N$, the discrete inverse Fourier transform $\mathcal{F}_N^{-1}$, and the Fourier conjugate operator $\widehat{\mathcal{G}}_N$. We aim to approximate each of these operators using Fourier neural operators.

We begin by defining

$$\mathbb{R}^{2\mathcal{K}_N} = \left(\mathbb{R}^2\right)^{\mathcal{K}_N} (\simeq \mathbb{C}^{\mathcal{K}_N}), \tag{5.6}$$

as the set of coefficients $\{(v_{1,k}, v_{2,k})\}_{k\in\mathcal{K}_N}$, where $v_{\ell,k} \in \mathbb{R}$ are indexed by a tuple (ℓ, k), with $\ell \in \{1, 2\}$ and $k \in \mathcal{K}_N$. We interpret the operator $\mathcal{F}_N \circ P_N$ as a mapping $\mathcal{F}_N \circ P_N : a \mapsto \{(\mathrm{Re}(\widehat{a}_k), \mathrm{Im}(\widehat{a}_k))\}_{|k|\leq N}$, where the input is $a \in L^2(\mathbb{T}^d)$ and the output $\{\mathrm{Re}(\widehat{a}_k), \mathrm{Im}(\widehat{a}_k)\}_{|k|\leq N} \in \mathbb{R}^{2\mathcal{K}_N}$ is considered a *constant* function in $L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N})$. Approximating this operator is a direct consequence of the following lemma, which is proven in Appendix B.2:

***Lemma* 5.4.** Let $B > 0$ and $N \in \mathbb{N}$ be given. For any $\epsilon > 0$, there exists a FNO $\mathcal{N} : L^2(\mathbb{T}^d) \to L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N})$, $v \mapsto \{\mathcal{N}(v)_{\ell,k}\}$, with *constant* output functions (constant with respect to $x \in \mathbb{T}^d$), such that

$$\left.\begin{aligned} \|\mathrm{Re}(\widehat{v}_k) - \mathcal{N}(v)_{1,k}\|_{L^\infty} \leq \epsilon \\ \|\mathrm{Im}(\widehat{v}_k) - \mathcal{N}(v)_{2,k}\|_{L^\infty} \leq \epsilon \end{aligned}\right\} \quad \forall\, k \in \mathbb{Z}^d,\ |k|_\infty \leq N,$$

for all $\|v\|_{L^2} \leq B$, where $\widehat{v}_k \in \mathbb{C}$ represents the k-th Fourier coefficient of v.

In the next step, we approximate the (discrete) inverse Fourier transform $\mathcal{F}_N^{-1}$ using an FNO. Since FNOs operate on functions rather than constants, we interpret the mapping

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

as a mapping

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

where the input $\{\mathrm{Re}(\widehat{v}_k), \mathrm{Im}(\widehat{v}_k)\}_{|k|\leq N} \in [-R, R]^{2\mathcal{K}_N}$ is identified with a *constant* function in $L^2(\mathbb{T}^d; [-R, R]^{2\mathcal{K}_N})$. The existence of an FNO of the form (4.4) that can approximate the above mapping to a desired accuracy follows from the subsequent lemma, proven in Appendix B.3:

***Lemma* 5.5.** Let $B > 0$ and $N \in \mathbb{N}$ be given. For any $\epsilon > 0$, there exists a FNO $\mathcal{N} : L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}) \to L^2(\mathbb{T}^d)$, such that for any $v \in L^2_N(\mathbb{T}^d)$ with $\|v\|_{L^2} \leq B$, we have

$$\|v - \mathcal{N}(w)\|_{L^2} \leq \epsilon,$$

where $w(x) := \{(\mathrm{Re}(\widehat{v}_k), \mathrm{Im}(\widehat{v}_k))\}_{k\in\mathcal{K}_N}$, i.e., $w \in L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N})$ is a constant function containing the real and imaginary parts of the Fourier coefficients $\widehat{v}_k$ of v.

We first address the simplified case where $s' = 0$, leveraging Lemma 5.3. That is, given a continuous operator $\mathcal{G} : H^s(\mathbb{T}^d) \to L^2(\mathbb{T}^d)$, a compact set $K \subset H^s(\mathbb{T}^d)$, and tolerance $\epsilon > 0$, we aim to construct an FNO $\mathcal{N} : H^s(\mathbb{T}^d) \to L^2(\mathbb{T}^d)$ satisfying $\sup_{a\in K} \|\mathcal{G}(a) - \mathcal{N}(a)\|_{L^2} \leq \epsilon$.

Let P_N denote the orthogonal Fourier projection given by (A.6) for $N \in \mathbb{N}$. We begin by observing that since $K \subset H^s(\mathbb{T}^d)$ is compact, the extended set $\widetilde{K}$ defined by

$$\widetilde{K} := K \cup \bigcup_{N\in\mathbb{N}} P_N K$$

is also compact (this follows from standard arguments). The continuity of $\mathcal{G}$ implies that its restriction to $\widetilde{K}$ is uniformly continuous, meaning there exists a modulus of continuity function $\omega : [0, \infty) \to [0, \infty)$ such that

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

Using the definition of the projected operator $\mathcal{G}_N$ from (5.1), we obtain:

$$\begin{aligned}\|\mathcal{G}(a) - \mathcal{G}_N(a)\|_{L^2} &\leq \|\mathcal{G}(a) - P_N\mathcal{G}(a)\|_{L^2} + \|P_N\mathcal{G}(a) - P_N\mathcal{G}(P_N a)\|_{L^2} \\ &\leq \|\mathcal{G}(a) - P_N\mathcal{G}(a)\|_{L^2} + \|\mathcal{G}(a) - \mathcal{G}(P_N a)\|_{L^2} \\ &\leq \sup_{v\in\mathcal{G}(\widetilde{K})} \|(1 - P_N)v\|_{L^2} + \omega\left(\sup_{a\in\widetilde{K}} \|(1 - P_N)a\|_{H^s}\right). \end{aligned} \tag{5.7}$$

The compactness of $\widetilde{K}$ ensures that $\mathcal{G}(\widetilde{K})$ is also compact, which yields:

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

Therefore, there exists $N \in \mathbb{N}$ large enough such that

$$\|\mathcal{G}(a) - \mathcal{G}_N(a)\|_{L^2} \leq \epsilon, \quad \forall\, a \in K \subset \widetilde{K}. \tag{5.8}$$

The remainder of our proof focuses on constructing an FNO approximation for $\mathcal{G}_N$. Note that $\mathcal{G}_N$ defines a continuous mapping $\mathcal{G}_N : L^2(\mathbb{T}^d) \to L^2(\mathbb{T}^d)$ via $a \mapsto P_N\mathcal{G}(P_N a)$, and K remains compact when viewed as a subset of $L^2(\mathbb{T}^d)$. We will establish the existence of an FNO $\mathcal{N} : L^2(\mathbb{T}^d) \to L^2(\mathbb{T}^d)$ such that

$$\sup_{a\in K} \|\mathcal{G}_N(a) - \mathcal{N}(a)\|_{L^2} < \epsilon.$$

The restriction of $\mathcal{N}$ to $H^s(\mathbb{T}^d) \subset L^2(\mathbb{T}^d)$ then provides the desired approximation of $\mathcal{G}$:

$$\sup_{a\in K} \|\mathcal{G}(a) - \mathcal{N}(a)\|_{L^2} < 2\epsilon.$$

Since $\epsilon > 0$ was arbitrary, this establishes our claim.

As outlined in our proof strategy, the construction of an FNO approximating $\mathcal{G}_N$ relies on the decomposition (5.5), which involves the Fourier conjugate operator $\widehat{G}_N$ defined in (5.4). We demonstrate that each component in the decomposition (5.5) can be approximated by FNOs to arbitrary accuracy.

Given $\epsilon > 0$, we select constants $R_K, R_{\widehat{K}}, R_{\widehat{\mathcal{G}}} > 0$ such that

$$\begin{cases} K \subset B_{R_K}(0) := \{\|u\|_{L^2} \leq R_K\} \subset L^2(\mathbb{T}^d), \\ \mathcal{F}_N \circ P_N\left(B_{R_K}(0)\right) \subset \left[-\dfrac{R_{\widehat{K}}}{2}, \dfrac{R_{\widehat{K}}}{2}\right]^{2\mathcal{K}_N}, \\ \widehat{\mathcal{G}}_N\left([-R_{\widehat{K}}, R_{\widehat{K}}]^{2\mathcal{K}_N}\right) \subset \left[-\dfrac{R_{\widehat{\mathcal{G}}}}{2}, \dfrac{R_{\widehat{\mathcal{G}}}}{2}\right]^{2\mathcal{K}_N}. \end{cases} \tag{5.9}$$

These bounds ensure that each FNO in our composition maps its domain appropriately into the domain of the subsequent FNO. We now construct FNO approximations for each step in the composition $\mathcal{G}_N = \mathcal{F}_N^{-1} \circ \widehat{\mathcal{G}}_N \circ (\mathcal{F}_N \circ P_N)$:

FNO approximation of $\mathcal{F}_N^{-1}$: We begin by constructing an FNO approximation for the final step in our composition. We interpret the mapping

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

as an operator

$$\mathcal{F}_N^{-1} : \begin{cases} L^2(\mathbb{T}^d; [-R, R]^{2\mathcal{K}_N}) \to L^2(\mathbb{T}^d), \\ \{\mathrm{Re}(\widehat{v}_k), \mathrm{Im}(\widehat{v}_k)\}_{|k|\leq N} \mapsto v(x), \end{cases} \tag{5.10}$$

where the input $\{\mathrm{Re}(\widehat{v}_k), \mathrm{Im}(\widehat{v}_k)\}_{|k|\leq N} \in [-R, R]^{2\mathcal{K}_N}$ corresponds to a *constant* function in $L^2(\mathbb{T}^d; [-R, R]^{2\mathcal{K}_N})$. For non-constant inputs $v(x)$, we define the mapping (5.10) by applying $\mathcal{F}_N^{-1}$ to the constant function $x \mapsto \fint_{\mathbb{T}^d} v(\xi)\, d\xi$. By Lemma 5.5, this mapping can be approximated to arbitrary precision by an FNO $\mathcal{N}_{\mathrm{IFT}} : L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}) \to L^2(\mathbb{T}^d)$ satisfying

$$\|\mathcal{N}_{\mathrm{IFT}}(\widehat{v}) - \mathcal{F}_N^{-1}(\widehat{v})\|_{L^2} \leq \epsilon/3, \tag{5.11}$$

for all *constant* input functions $\widehat{v} \in L^2(\mathbb{T}^d; [-R, R]^{2\mathcal{K}_N})$.

FNO approximation of $\widehat{\mathcal{G}}_N$: We treat the Fourier conjugate operator $\widehat{\mathcal{G}}_N$ from (5.4) as a continuous mapping

$$\widehat{\mathcal{G}}_N : [-R_{\widehat{K}}, R_{\widehat{K}}]^{2\mathcal{K}_N} \subset \mathbb{R}^{2\mathcal{K}_N} \to \mathbb{R}^{2\mathcal{K}_N}.$$

Since $[-R_{\widehat{K}}, R_{\widehat{K}}]^{2\mathcal{K}_N}$ is compact, there exists a finite-dimensional standard neural network $\widehat{\mathcal{N}} : \mathbb{R}^{2\mathcal{K}_N} \to \mathbb{R}^{2\mathcal{K}_N}$ such that

$$\sup_{\widehat{v}\in[-R_{\widehat{K}}, R_{\widehat{K}}]^{2\mathcal{K}_N}} \|\widehat{\mathcal{G}}_N(\widehat{v}) - \widehat{\mathcal{N}}(\widehat{v})\|_{\ell^2} \leq \epsilon/3. \tag{5.12}$$

Moreover, by (5.9), we have

$$\widehat{G}_N\left([-R_{\widehat{K}}, R_{\widehat{K}}]^{2\mathcal{K}_N}\right) \subset \left[-\frac{R_{\widehat{\mathcal{G}}}}{2}, \frac{R_{\widehat{\mathcal{G}}}}{2}\right]^{2\mathcal{K}_N}.$$

By selecting a neural network approximation $\widehat{\mathcal{N}}$ with sufficient accuracy, we can additionally ensure that

$$\widehat{\mathcal{N}}\left([-R_{\widehat{K}}, R_{\widehat{K}}]^{2\mathcal{K}_N}\right) \subset \left[-R_{\widehat{\mathcal{G}}}, R_{\widehat{\mathcal{G}}}\right]^{2\mathcal{K}_N},$$

alongside condition (5.12). We observe that for $v \in L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N})$, the corresponding mapping

$$\widehat{\mathcal{N}} : L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}) \to L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}), \quad v(x) \mapsto \widehat{\mathcal{N}}(v(x)),$$

constitutes an FNO containing only *local* layers of the form

$$v_\ell(x) \mapsto \sigma\left(A_\ell v_\ell(x) + b_\ell\right), \quad (A_\ell \in \mathbb{R}^{d_v \times d_v},\ b_\ell \in \mathbb{R}^{d_v}),$$

where $d_v := 2|\mathcal{K}_N|$, corresponding to an FNO with all $P_\ell \equiv 0$ (see Remark 4.1). We therefore identify $\widehat{\mathcal{N}}$ with this specific FNO in what follows.

FNO approximation of $\mathcal{F}_N \circ P_N$: Finally, we interpret

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

as a mapping

$$\mathcal{F}_N \circ P_N : \begin{cases} B_{R_K}(0) \to L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}), \\ v \mapsto \{\mathrm{Re}(\widehat{v}_k), \mathrm{Im}(\widehat{v}_k)\}_{|k|\leq N}, \end{cases} \tag{5.13}$$

where the output $\{\mathrm{Re}(\widehat{v}_k), \mathrm{Im}(\widehat{v}_k)\}_{|k|\leq N} \in \mathbb{R}^{2\mathcal{K}_N}$ represents a *constant* function in $L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N})$. According to Lemma 5.4, this mapping can be approximated to arbitrary precision by an FNO $\mathcal{N}_{\mathrm{FT}} : B_{R_K}(0) \to L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N})$ (producing constant output functions). Specifically, denoting $\mathrm{Lip}(\widehat{\mathcal{N}})$ as the Lipschitz constant of the FNO constructed in the previous step, we can ensure that

$$\mathrm{Lip}(\widehat{\mathcal{N}})\, \|\mathcal{F}_N P_N v - \mathcal{N}_{\mathrm{FT}}(v)\|_{\ell^2} \leq \epsilon/3, \quad \forall\, v \in B_{R_K}(0), \tag{5.14}$$

and since (5.9) gives us

$$\mathcal{F}_N \circ P_N(B_{R_K}(0)) \subset \left[-\frac{R_{\widehat{K}}}{2}, \frac{R_{\widehat{K}}}{2}\right]^{2\mathcal{K}_N},$$

we can additionally guarantee that

$$\mathcal{N}_{\mathrm{FT}}\left(B_{R_K}(0)\right) \subset \left[-R_{\widehat{K}}, R_{\widehat{K}}\right]^{2\mathcal{K}_N}.$$

Error analysis for the composite FNO: We now define the composite FNO $\mathcal{N}(a) := \mathcal{N}_{\mathrm{IFT}} \circ \widehat{\mathcal{N}} \circ \mathcal{N}_{\mathrm{FT}}(a)$, where each component has been constructed above. We observe that

$$\begin{aligned}
&\sup_K \|\mathcal{G}_N - \mathcal{N}\|_{L^2} \\
&\quad \leq \sup_{B_{R_K}(0)} \left\| \mathcal{F}_N^{-1} \circ \widehat{\mathcal{G}}_N \circ \mathcal{F}_N \circ P_N - \mathcal{N}_{\mathrm{IFT}} \circ \widehat{\mathcal{N}} \circ \mathcal{N}_{\mathrm{FT}} \right\|_{L^2} \\
&\quad \leq \sup_{B_{R_K}(0)} \left\| \mathcal{F}_N^{-1} \circ \widehat{\mathcal{G}}_N \circ \mathcal{F}_N \circ P_N - \mathcal{F}_N^{-1} \circ \widehat{\mathcal{N}} \circ \mathcal{N}_{\mathrm{FT}} \right\|_{L^2} \\
&\qquad + \sup_{B_{R_K}(0)} \left\| \mathcal{F}_N^{-1} \circ \widehat{\mathcal{N}} \circ \mathcal{N}_{\mathrm{FT}} - \mathcal{N}_{\mathrm{IFT}} \circ \widehat{\mathcal{N}} \circ \mathcal{N}_{\mathrm{FT}} \right\|_{L^2} \\
&\quad \leq \sup_{B_{R_K}(0)} \left\| \widehat{\mathcal{G}}_N \circ \mathcal{F}_N \circ P_N - \widehat{\mathcal{N}} \circ \mathcal{N}_{\mathrm{FT}} \right\|_{L^2} \\
&\qquad + \sup_{\widehat{\mathcal{N}}\left(\mathcal{N}_{\mathrm{FT}}\left(B_{R_K}(0)\right)\right)} \left\| \mathcal{F}_N^{-1} - \mathcal{N}_{\mathrm{IFT}} \right\|_{L^2} \\
&\quad =: (I) + (II).
\end{aligned}$$

For term (II), we note that

$$\widehat{\mathcal{N}}\left(\mathcal{N}_{\mathrm{FT}}\left(B_{R_K}(0)\right)\right) \subset \widehat{\mathcal{N}}\left(\left[-R_{\widehat{K}}, R_{\widehat{K}}\right]^{2\mathcal{K}_N}\right) \subset \left[-R_{\widehat{\mathcal{G}}}, R_{\widehat{\mathcal{G}}}\right]^{2\mathcal{K}_N},$$

which, combined with (5.11), yields:

$$(II) \leq \sup_{\left[-R_{\widehat{\mathcal{G}}}, R_{\widehat{\mathcal{G}}}\right]^{2\mathcal{K}_N}} \left\| \mathcal{F}_N^{-1} - \mathcal{N}_{\mathrm{IFT}} \right\|_{L^2} \leq \epsilon/3.$$

To bound term (I), we decompose it as:

$$\begin{aligned}
(I) &= \sup_{B_{R_K}(0)} \left\| \widehat{\mathcal{G}}_N \circ \mathcal{F}_N \circ P_N - \widehat{\mathcal{N}} \circ \mathcal{N}_{\mathrm{FT}} \right\|_{L^2} \\
&\leq \sup_{B_{R_K}(0)} \left\| \widehat{\mathcal{G}}_N \circ \mathcal{F}_N \circ P_N - \widehat{\mathcal{N}} \circ \mathcal{F}_N \circ P_N \right\|_{L^2} \\
&\quad + \sup_{B_{R_K}(0)} \left\| \widehat{\mathcal{N}} \circ \mathcal{F}_N \circ P_N - \widehat{\mathcal{N}} \circ \mathcal{N}_{\mathrm{FT}} \right\|_{L^2} \\
&=: (Ia) + (Ib).
\end{aligned}$$

For subterm (Ia), we observe that

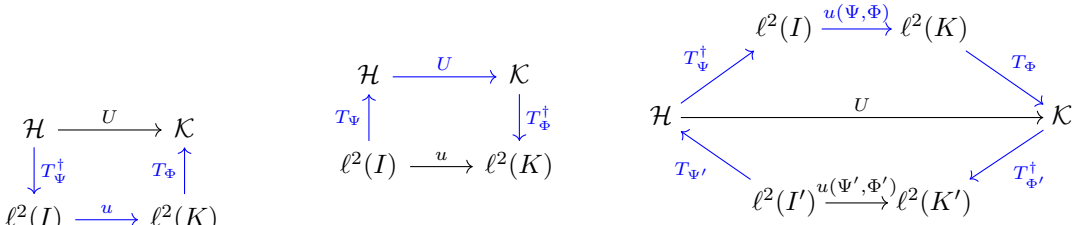
$$\mathcal{F}_N\left(P_N\left(B_{R_K}(0)\right)\right) \subset \left[-R_{\widehat{K}}, R_{\widehat{K}}\right]^{2\mathcal{K}_N},$$

leading to

$$(Ia) \leq \sup_{\left[-R_{\widehat{K}}, R_{\widehat{K}}\right]^{2\mathcal{K}_N}} \left\| \widehat{\mathcal{G}}_N - \widehat{\mathcal{N}} \right\|_{L^2} \leq \epsilon/3,$$

by virtue of (5.12). For subterm (Ib), we have:

$$(Ib) \leq \mathrm{Lip}(\widehat{\mathcal{N}}) \sup_{B_{R_K}(0)} \|\mathcal{F}_N \circ P_N - \mathcal{N}_{\mathrm{FT}}\|_{L^2} \leq \epsilon/3,$$



(a) An alias-free operator's diagram commutes.

(b) *ReNO*: an alias-free u can be constructed by discretizing the operator U for different discretizations given by Ψ, Φ.

(c) *ReNO*: discrete representations u, u' for different discretizations are equivalent.

Figure 2: An alias-free framework is presented, where U represents the continuous operator and u its discrete counterpart. Synthesis operators, along with their pseudo-inverses, serve as the critical link, enabling transformation between the continuous function space and the discrete space used for computation.

following from (5.14).

Combining all estimates, we conclude:

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

This demonstrates that the continuous operator $\mathcal{G}_N$ can be approximated by an FNO $\mathcal{N}$ to any prescribed accuracy $\epsilon > 0$. Together with (5.2), this completes our proof of the universal approximation theorem 5.2 for the special case $s' = 0$. The general case with $s' \geq 0$ follows directly from Lemma 5.3. □

6 Aliasing Errors in FNOs

FNOs possess universal approximation capabilities. However, in practice, evaluating the Fourier transform is impossible since it requires computing an integral exactly. Instead, in actual computations, we should compute discrete Fourier Transform between layers of the FNO instead. This introduces a critical challenge: aliasing error may lead to reduced accuracy in high-precision regimes [Fanaskov and Oseledets, 2023, Bartolucci et al., 2023], impeding the performance of FNOs. In this section, we analyze the aliasing errors within the FNO layers by theoretical analysis and empirical analysis.

6.1 Theoretical Analysis

6.1.1 Representation equivalent Neural Operators (ReNO)

Bartolucci et al. [2023] introduces an important concept: *Representation equivalent Neural Operator (ReNO)*. That is, given any pair (Ψ, Φ) of frame sequences corresponding to $\mathcal{H}$ and $\mathcal{K}$, we define a discrete-level mapping $u(\Psi, \Phi)\colon \operatorname{Ran} T_\Psi^\dagger \to \operatorname{Ran} T_\Phi^\dagger$, which operates on discrete function representations. Note that this mapping is parameterized by the frame sequences, which is natural since changing the discretization necessitates modifying the function definition accordingly. For notational simplicity, when the context is clear, we will denote $u(\Psi, \Phi)$ simply as u.

***Definition* 6.1. Representation equivalent Neural Operators (ReNO).** A pair (U, u) constitutes a ReNO when, for every frame sequence pair (Ψ, Φ) satisfying $\operatorname{Dom} U \subseteq \mathcal{M}_\Psi$ and $\operatorname{Ran} U \subseteq \mathcal{M}_\Phi$, the absence of aliasing occurs, meaning that the aliasing error operator equals zero:

$$\varepsilon(U, u, \Psi, \Phi) = 0. \tag{6.1}$$

This property is abbreviated as $\varepsilon(U, u) = 0$.

Put differently, the diagram shown in Figure 2a commutes for every applicable pair (Ψ, Φ). Under these circumstances, all discrete representations $u(\Psi, \Phi)$ are equivalent, signifying that they uniquely characterize the identical underlying operator U, provided that a continuous-discrete equivalence property is satisfied at the function space level. The domain and range requirements in Definition 6.1 ensure that the frames can sufficiently represent the input and output functions of U.

***Remark* 6.2.** When the aliasing error $\varepsilon(U, u, \Psi, \Phi)$ vanishes (as mandated by Definition 6.1), the requirement that $u(\Psi, \Phi)$ maps $\operatorname{Ran} T_\Psi^\dagger \subseteq \ell^2(I)$ into $\operatorname{Ran} T_\Phi^\dagger \subseteq \ell^2(K)$ leads to

$$u(\Psi, \Phi) = T_\Phi^\dagger \circ U \circ T_\Psi. \tag{6.2}$$

We note that this formulation of $u(\Psi, \Phi)$ ensures that the diagram in Figure 2b commutes. Specifically, after establishing the discrete representations Ψ, Φ for input and output functions, there exists a unique method for defining a discretization $u(\Psi, \Phi)$ that remains consistent with the continuous operator U, given by (6.2). In practical applications, we may encounter various discrete representations of input and output functions, which theoretically corresponds to changing reference frames in the function spaces.

It is crucial to note that to eliminate any aliasing error, the discrete representation of U *must* depend on the selected frame sequences. Therefore, u necessarily depends on Ψ and Φ, making it inherently discretization-dependent. The proof of Remark 6.2 is provided in B.4.

Specifically, Remark 6.2 directly provides a formula for transforming from one discrete representation to another:

$$u(\Psi', \Phi') = T_{\Phi'}^\dagger \circ T_\Phi \circ u(\Psi, \Phi) \circ T_\Psi^\dagger \circ T_{\Psi'}, \tag{6.3}$$

provided that both frame sequence pairs (Ψ, Φ) and (Ψ', Φ') satisfy the conditions specified in Definition 6.1. Consequently, the diagram in 2c commutes.

Formula (6.3) directly establishes Proposition 6.3, which creates a connection between aliasing and representation equivalence. This demonstrates the distinction from *discretization invariance,* extensively discussed in Kovachki et al. [2023]: whereas that concept establishes asymptotic consistency, representation equivalence enables direct comparison between any two specified discretizations and ensures their equivalence.

***Proposition* 6.3. Equivalence of ReNO discrete representations.** Consider (U, u) as a ReNO. For any two frame sequence pairs (Ψ, Φ) and (Ψ', Φ') that satisfy the conditions in Definition 6.1, we obtain

$$\tau(u, u') = 0,$$

where u' represents $u(\Psi', \Phi')$ by slight notational abuse.

Therefore, assuming that the discrete mapping at each discretization level is consistent with the underlying continuous operator, we obtain a unique method for expressing the operator at every discretization. Additionally, formula (6.3) bears close resemblance to similar formulas presented in Kovachki et al. [2023], Li et al. [2021] when evaluating *single shot super resolution*.

6.1.2 Fourier layer in FNOs

Our focus here is on the Fourier layer of FNOs, specifically

$$\mathcal{K}v = \mathcal{F}^{-1}(R \odot \mathcal{F})(v), \tag{6.4}$$

where $\mathcal{F}, \mathcal{F}^{-1}$ represent the Fourier transform and its inverse, and R denotes a low-pass filter. In Kovachki et al. [2023], the authors define the Fourier layer on the space $L^2(\mathbb{T})$ of 2-periodic functions and, with minor notational liberties, they refer to the mappings $\mathcal{F}\colon L^2(\mathbb{T}) \to \ell^2(\mathbb{Z})$ and $\mathcal{F}^{-1}\colon \ell^2(\mathbb{Z}) \to L^2(\mathbb{T})$,

$$\mathcal{F}w(k) = \langle w, e^{i\pi kx} \rangle, \qquad \mathcal{F}^{-1}(\{W_k\}_{k\in\mathbb{Z}}) = \sum_{k\in\mathbb{Z}} W_k e^{i\pi kx},$$

as the Fourier transform and inverse Fourier transform. Furthermore, the authors define the discrete version of (6.4) as $F^{-1}(R \odot F)$, where F, F^{-1} denote the discrete Fourier transform (DFT) and its inverse, assuming access only to pointwise function evaluations. However, *the space of 2-periodic functions is too extensive to permit any form of continuous-discrete equivalence (CDE)* when input v and output $\mathcal{K}v$ are represented through point samples. Therefore, we consider smaller subspaces of $L^2(\mathbb{T})$ that enable CDEs. Specifically, we demonstrate that FNO Fourier layers can be implemented as

Representation equivalent Operators (cf. Definition 6.1) between bandlimited and periodic functions. Let $K > 0$ and define $\mathcal{P}_K$ as the space of bandlimited 2-periodic functions

$$\mathcal{P}_K = \left\{ w(x) = \sum_{k=-K}^{K} W_k e^{i\pi k x} : \{W_k\}_{k=-K}^{K} \in \mathbb{C}^{2K+1} \right\}.$$

Every function $w \in \mathcal{P}_K$ can be uniquely represented by its Fourier coefficients $\{W_k\}_{k=-K}^{K}$ and by its samples $\{w(\frac{k}{2K+1})\}_{k=-K}^{K}$, as shown in Vetterli et al. [2014]. The latter are coefficients of w with respect to the orthonormal basis

$$\Psi_K = \left\{ \frac{1}{\sqrt{2(2K+1)}} d\left(\cdot - \frac{2k}{2K+1}\right) \right\}_{k=0}^{2K}, \tag{6.5}$$

where d denotes the Dirichlet kernel of order K and period 2, defined as

$$d(t) = \sum_{k=-K}^{K} e^{i\pi k t}.$$

Moreover, the DFT $\{\widehat{W}_k\}_{k=-K}^{K}$ of the sample sequence $\{w(\frac{k}{2K+1})\}_{k=-K}^{K}$ relates to the Fourier coefficients of w through

$$\widehat{W}_k = (2K+1)W_k, \quad k = -K, \ldots, K,$$

with proof provided in Vetterli et al. [2014]. This yields the commutative diagram

$$\begin{array}{ccc} \mathcal{P}_K & \xrightarrow{\mathcal{F}} & \mathbb{C}^{2K+1}, \\ \downarrow{\scriptstyle T^{\dagger}_{\Psi_K}} & & \uparrow{\scriptstyle \mathrm{Id}} \\ \mathbb{C}^{2K+1} & \xrightarrow{(2K+1)\cdot \mathrm{F}} & \mathbb{C}^{2K+1} \end{array}$$

where $T^{\dagger}_{\Psi_K} : \mathcal{P}_K \to \mathbb{C}^{2K+1}$ represents the analysis operator associated with basis (6.5). Similarly, we construct the commutative diagram

$$\begin{array}{ccc} \mathbb{C}^{2K'+1} & \xrightarrow{\mathcal{F}^{-1}} & \mathcal{P}_{K'} \\ \downarrow{\scriptstyle \mathrm{Id}} & & \uparrow{\scriptstyle T_{\Psi_{K'}}} \\ \mathbb{C}^{2K'+1} & \xrightarrow{\frac{1}{(2K'+1)}\cdot \mathrm{F}^{-1}} & \mathbb{C}^{2K'+1} \end{array}$$

where $T_{\Psi_{K'}} : \mathbb{C}^{2K'+1} \to \mathcal{P}_{K'}$ represents the synthesis operator associated with basis (6.5) with $K = K'$.

Here, $R = \{R_k\}_{k=-K'}^{K'}$, with $K' \leq K$, denotes the Fourier coefficients of a 2-periodic function, and the mapping $R \odot \mathcal{F} \colon \mathcal{P}_K \to \mathbb{C}^{2K'+1}$ is defined as

$$(R \odot \mathcal{F}w)(k) = R_k W_k, \quad k = -K', \ldots, K'.$$

By definition, $R \odot \mathcal{F}$ provides a continuous-discrete equivalence operation. Overall, we obtain the commutative diagram

$$\begin{array}{ccccccc} \mathcal{P}_K & \xrightarrow{\mathcal{F}} & \mathbb{C}^{2K+1} & \xrightarrow{R\odot} & \mathbb{C}^{2K'+1} & \xrightarrow{\mathcal{F}^{-1}} & \mathcal{P}_{K'} \\ \downarrow{\scriptstyle T^{\dagger}_{\Psi_K}} & & \uparrow{\scriptstyle \mathrm{Id}} & & \updownarrow{\scriptstyle \mathrm{Id}} & & \uparrow{\scriptstyle T_{\Psi_{K'}}} \\ \mathbb{C}^{2K+1} & \xrightarrow{(2K+1)\cdot \mathrm{F}} & \mathbb{C}^{2K+1} & \xrightarrow{R\odot} & \mathbb{C}^{2K'+1} & \xrightarrow{\frac{1}{(2K'+1)}\cdot \mathrm{F}^{-1}} & \mathbb{C}^{2K'+1} \end{array}$$

demonstrating that the discretization of Fourier layer 6.4, the blue path in the commutative diagram above, is defined through Equation (6.2). Consequently, Fourier layer 6.4, considered as an operator from $\mathcal{P}_K$ to $\mathcal{P}_{K'}$, satisfies the requirements of a Representation equivalent Operator (cf. Definition 6.1). However, pointwise activation functions applied to bandlimited inputs will not necessarily

preserve bandwidth. In fact, with common activation functions like ReLU, $\sigma(f) \notin \mathcal{P}_K$ for any $K \in \mathbb{N}$. Therefore, the FNO layer

$$\sigma(\mathcal{K}v) = \sigma(\mathcal{F}^{-1}(R \odot \mathcal{F})(v))$$

may not respect continuous-discrete equivalence and can introduce aliasing errors, a fact previously identified in Fanaskov and Oseledets [2023]. Hence, FNOs *may not qualify as ReNOs in the sense of Definition 6.1*.

6.2 Empirical Analysis

We analyze the impact of activation functions using empirical analysis. Specifically, we consider a function f belonging to the space $\mathcal{P}_K$, where $K = 20$, assuming f to be both periodic and bandlimited. The uppermost plot in Figure 3 visualizes the values of f, alongside its transformations via the ReLU (ReLU(f)), GELU (GeLU(f)), Sigmoid (sigmoid(f)), and Tanh (tanh(f)) activation functions. Given that f's Fourier coefficients vanish beyond the Nyquist frequency K, sampling f at $2K + 1$ points on a grid is sufficient for its complete characterization. However, the lower plot demonstrates that this condition no longer holds for ReLU(f), GeLU(f), sigmoid(f), and tanh(f). Consequently, the discrete representation on the grid fails to uniquely represent the continuous functions, leading to aliasing artifacts.

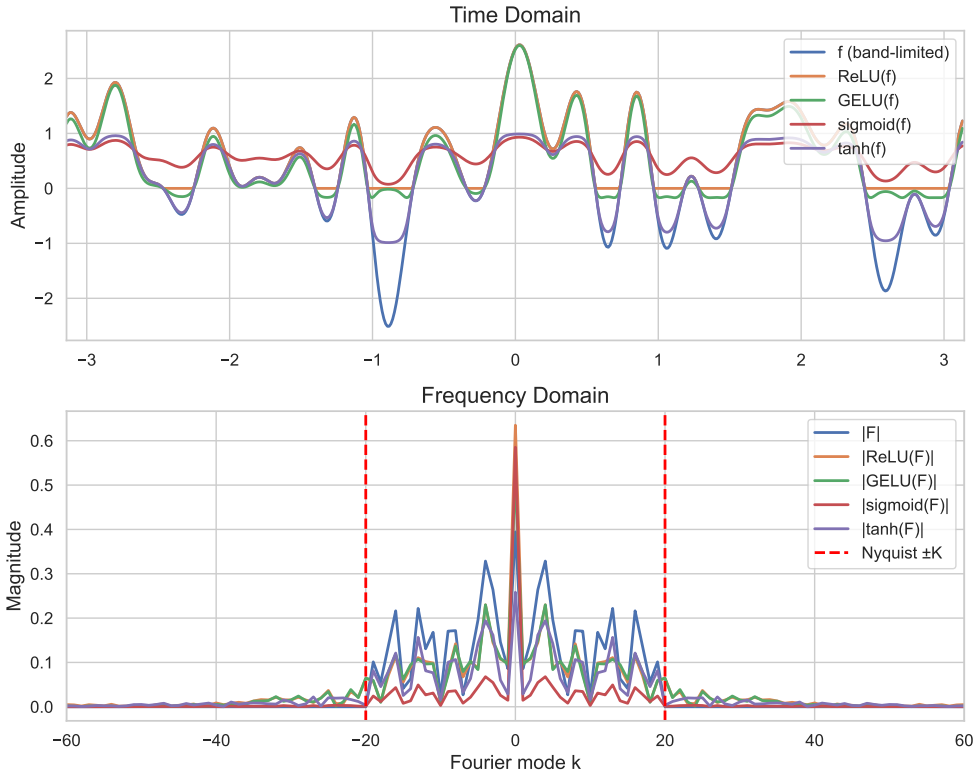


Figure 3: The application of the activation function broadens the bandwidth of the input function beyond the Nyquist frequency, resulting in aliasing errors.

7 Spectral Neural Operator

To eliminate this aliasing issue within the FNO layers, the Spectral Neural Operator (SNO) [Fanaskov and Oseledets, 2023] works directly in coefficient space. Given an input function $a(x)$, we first compute its truncated spectral coefficients

$$d = (d_k)_{|k| \le K}, \quad a(x) \approx \sum_{|k| \le K} d_k \, \phi_k(x).$$

We then seek a finite-dimensional map

$$d \xrightarrow{\mathcal{N}_\theta} b, \qquad b = (b_\ell)_{|\ell| \le L},$$

so that the output function $u(x) = \sum_{|\ell| \le L} b_\ell \, \phi_\ell(x)$ approximates the true solution $\mathcal{F}^*(a)$. In practice we stack the coefficient vector d into a tall matrix $U \in \mathbb{C}^{(2K+1) \times d_v}$ when multiple feature channels (multiple functions) d_v are used, and implements $\mathcal{N}_\theta$ in three neural-network stages:

$$U \xrightarrow{N_1} V \xrightarrow{N_2} Y \xrightarrow{N_3} W,$$

where N_1 and N_3 are pointwise (channel-wise) linear layers with nonlinear activations, altering only the number of feature channels, that is:

$$\begin{aligned} U^{(n+1)} &= \sigma\left(U^{(n)} A + b\right) \in \mathbb{C}^{k \times m} \\ U^{(n)} &\in \mathbb{C}^{k \times l},\ A \in \mathbb{C}^{l \times m},\ b \in \mathbb{C}^{1 \times m}, \end{aligned} \tag{7.1}$$

where σ is a pointwise activation.

Meanwhile, N_2 introduces global mixing by a learned low-rank operator B (corresponding to low-rank integral operator) acting along the coefficient dimension. It is stacked from the following layers:

$$\begin{aligned} U^{(n+1)} &= \sigma\left(B U^{(n)} A + b\right) \in \mathbb{C}^{r \times m} \\ B &\in \mathbb{C}^{r \times k},\ U^{(n)} \in \mathbb{C}^{k \times l},\ A \in \mathbb{C}^{l \times m},\ b \in \mathbb{C}^{r \times m}. \end{aligned} \tag{7.2}$$

Finally, the resulting matrix $W \in \mathbb{C}^{(2L+1) \times d_u}$ provides the output coefficients for each solution channel, and application of the inverse FFT or DCT reconstructs $u(x)$ at arbitrary points.

By learning entirely in the truncated coefficient domain, SNOs avoid the aliasing errors that plague pointwise nonlinearities in Fourier layers, as shown theoretically in Appendix B.5. Moreover, because spectral coefficients are explicit, we obtain immediate access to spectral decay rates, Sobolev-norm estimates, and derivative bounds. The per-layer complexity is $O(N \log N + N \, d_v^2)$. SNOs also enjoy *mesh invariance*: once the coefficient map $\mathcal{N}_\theta$ is trained, evaluation on any grid or set of points in D requires only coefficient reconstruction via the basis functions $\{\phi_k\}$, without retraining.

8 Numerical Experiments

In this section, we empirically evaluate the performance of the Spectral Neural Operator (SNO) against the widely-used Fourier Neural Operator (FNO) baseline. We conduct experiments on three benchmark problems governed by canonical partial differential equations: the viscous Burgers' equation [Burgers, 1948], the Korteweg-de Vries (KdV) equation [Korteweg and De Vries, 1895], and the Kuramoto-Sivashinsky (KS) equation [Kuramoto, 1978, Ashinsky, 1988]. We compare FNO with two variants of SNO: one employing a Fourier basis [baron de Fourier, 1822] (SNO-Fourier) and another using a Chebyshev basis [Mason and Handscomb, 2002] (SNO-Chebyshev).

8.1 Governing Equations

Due to the computational resource constraint, we only use 1D data for both neural operators here. We focus on learning the solution operator for three 1D time-dependent PDEs, each exhibiting distinct physical behaviors. The task is to learn the mapping from an initial condition $u(x, 0)$ to the solution at a future time T, i.e., $\mathcal{G} : u(x, 0) \mapsto u(x, T)$. All problems are considered on a periodic domain $x \in [0, 2\pi]$.

1. **Viscous Burgers' Equation:** This equation is a fundamental model for convection-diffusion phenomena, combining a nonlinear convection term and a linear diffusion term. It is given by:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2} \tag{8.1}$$

where ν is the viscosity coefficient. For our experiments, we set $\nu = 0.01$. This equation models the formation of shock waves balanced by viscous dissipation.

2. **Korteweg-de Vries (KdV) Equation:** The KdV equation models weakly non-linear, long-wave propagation in dispersive media, famously describing shallow water waves. It is known for its stable, localized soliton solutions. The equation is:
$$\frac{\partial u}{\partial t} + 6u\frac{\partial u}{\partial x} + \frac{\partial^3 u}{\partial x^3} = 0 \tag{8.2}$$
The third-order derivative term introduces dispersion, which counteracts the wave-steepening effect of the nonlinear term.
3. **Kuramoto-Sivashinsky (KS) Equation:** The KS equation is a model for instabilities and spatio-temporal chaos in various physical systems, such as flame fronts and thin film flows. It is defined as:
$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + \frac{\partial^2 u}{\partial x^2} + \frac{\partial^4 u}{\partial x^4} = 0 \tag{8.3}$$
The equation features a second-order anti-diffusion term ($+\partial^2 u/\partial x^2$) that introduces instability, which is regularized by the fourth-order hyper-diffusion term ($+\partial^4 u/\partial x^4$).

8.2 Data Generation

We summarize the key aspects of the data generation pipeline below.

Numerical Solver To ensure accuracy and stability, we employ a second-order Implicit-Explicit Runge-Kutta (IMEX-RK2) scheme [Ascher et al., 1997]. This method is well-suited for the target PDEs as it treats the stiff linear terms (e.g., diffusion, dispersion) implicitly in Fourier space while handling the non-linear advection term explicitly. The spatial domain $[0, 2\pi]$ is discretized with $N_x = 256$ grid points. The solution operator is learned for a fixed future time of $T = 0.1$, with a fine time step of $\Delta t = 10^{-5}$ used in the solver to guarantee a high-fidelity ground truth.

Initial Conditions For each PDE, we generate a diverse set of random initial conditions to ensure the learned operator generalizes well.

- For the **Burgers'** equation, initial conditions are a mix of random Gaussian pulses and superpositions of low-frequency sinusoids.
- For the **KdV** equation, initial conditions are composed of single and multiple soliton-like profiles (using sech^2 functions) with varying amplitudes and positions.
- For the **KS** equation, initial conditions are generated as a sum of several sinusoids with random phases and amplitudes, plus a small amount of white noise to effectively trigger its chaotic dynamics.

For each equation, we generate a dataset containing $2,000$ samples for training and $1,000$ samples for testing.

8.3 Implementation Details

All models are implemented in PyTorch [Paszke, 2019] and trained on a single A100 GPU. The goal is to ensure a fair comparison by keeping the number of trainable parameters and the training procedure consistent across all models.

Model Hyperparameters We carefully selected hyperparameters for FNO and SNO to have a comparable number of trainable parameters (approximately 209k for both).

- **FNO**: The architecture consists of a lifting layer, 3 Fourier layers, and a projection layer. The key hyperparameters are a feature width of `width=64` and `modes=16` Fourier modes kept in each layer.
- **SNO (Fourier & Chebyshev)**: The SNO architecture consists of a lifting network, 3 spectral mixing layers, and a projection network. The hyperparameters are set to `n_coeffs=64` (number of input spectral coefficients), `lifting_features=64`, and `integral_neurons=256`. For SNO-Chebyshev, inputs are transformed to the spectral domain using the Discrete Cosine Transform (DCT) [Ahmed et al., 2006], while SNO-Fourier uses the Real Fast Fourier Transform (RFFT) [Sorensen et al., 2003].

Training and Evaluation The training setup is identical for all experiments. We use the AdamW optimizer [Loshchilov and Hutter, 2017] with a learning rate of $\eta = 0.001$ and a weight decay of 10^{-5}. A cosine annealing learning rate scheduler [Loshchilov and Hutter, 2016] is employed over the entire training duration. The models are trained by minimizing the relative L2 error, defined as $\mathcal{L}(u, \widehat{u}) = \frac{\|u-\widehat{u}\|_{L^2}}{\|u\|_{L^2}}$, where u is the ground truth and $\widehat{u}$ is the model prediction. All models are trained for a total of $20,000$ epochs with a batch size of $2,000$.

8.4 Results

We report the final test performance of all models across the three PDE benchmarks. The evaluation metric is the average relative L2 error on the test set. The results, along with the number of trainable parameters for each model, are summarized in Table 1. The result shows that FNO generally performs better on 1D settings.

Table 1: Comparison of test set relative L2 error for FNO, SNO-Fourier, and SNO-Chebyshev across three different PDE benchmarks. The number of trainable parameters for each model is kept approximately equal for a fair comparison.

Model	Parameters	Burgers	KdV	KS
FNO	≈ 209.2k	4.47×10^{-4}	3.33×10^{-3}	7.55×10^{-4}
SNO-Fourier	≈ 209.1k	1.30×10^{-2}	7.49×10^{-3}	2.08×10^{-3}
SNO-Chebyshev	≈ 209.1k	8.33×10^{-3}	6.64×10^{-3}	4.60×10^{-3}

9 Conclusion

This work presented a comparative study of Fourier Neural Operators (FNOs) and Spectral Neural Operators (SNOs). We illustrated the architecture of FNO and SNO, and conducted theoretical analysis and numerical experiments for both neural operators. This work provides guidance for practitioners on selecting the appropriate neural operator architecture for different application contexts.

Acknowledgment

I would like to express my sincere gratitude to Professor Lei Zhang. His course on numerical solutions was immensely beneficial, not only equipping me with a wealth of knowledge but also fostering a deep interest in this field of research. I am also very thankful for his thorough and insightful guidance on this course project, from advising on the suitability of the topic to recommending essential reading materials. I am truly grateful for his mentorship and support.

References

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.

Anima Anandkumar, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Nikola Kovachki, Zongyi Li, Burigede Liu, and Andrew Stuart. Neural operator: Graph kernel network for partial differential equations. In *ICLR 2020 workshop on integration of deep neural models and differential equations*, 2020.

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.

Zongyi Li, Nikola Borislavov Kovachki, Kamyar Azizzadenesheli, Burigede liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=c8P9NQVtmnO`.

Vladimir Sergeevich Fanaskov and Ivan V Oseledets. Spectral neural operators. In *Doklady Mathematics*, volume 108, pages S226–S232. Springer, 2023.

Francesca Bartolucci, Emmanuel de Bézenac, Bogdan Raonic, Roberto Molinaro, Siddhartha Mishra, and Rima Alaifari. Are neural operators really neural operators? frame theory meets operator learning. *arXiv preprint arXiv:2305.19913*, 2023.

Christian Grossmann. *Numerical treatment of partial differential equations*. Springer, 2007.

Philippe G Ciarlet. *The finite element method for elliptic problems*. SIAM, 2002.

Susanne C Brenner. *The mathematical theory of finite element methods*. Springer, 2008.

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

Dmitrii Kochkov, Jamie A Smith, Ayya Alieva, Qing Wang, Michael P Brenner, and Stephan Hoyer. Machine learning–accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences*, 118(21):e2101784118, 2021.

Xiaoxiao Guo, Wei Li, and Francesco Iorio. Convolutional neural networks for steady flow approximation. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 481–490, 2016.

Jonas Adler and Ozan Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12):124007, 2017.

Saakaar Bhatnagar, Yaser Afshar, Shaowu Pan, Karthik Duraisamy, and Shailendra Kaushik. Prediction of aerodynamic flow fields using convolutional neural networks. *Computational Mechanics*, 64:525–545, 2019.

Leah Bar and Nir Sochen. Unsupervised deep learning algorithm for pde-based forward and inverse problems. *arXiv preprint arXiv:1904.05417*, 2019.

Shaowu Pan and Karthik Duraisamy. Physics-informed probabilistic learning of linear embeddings of nonlinear dynamics with guaranteed stability. *SIAM Journal on Applied Dynamical Systems*, 19 (1):480–509, 2020.

James W Cooley and John W Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, 19(90):297–301, 1965.

Levi E. Lingsch, Mike Yan Michelis, Emmanuel de Bezenac, Sirani M. Perera, Robert K. Katzschmann, and Siddhartha Mishra. Beyond regular grids: Fourier-based neural operators on arbitrary domains. In *Forty-first International Conference on Machine Learning*, 2024. URL `https://openreview.net/forum?id=aVqqoFAavs`.

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.

Martin Vetterli, Jelena Kovačević, and Vivek K Goyal. *Foundations of signal processing*. Cambridge University Press, 2014.

Johannes Martinus Burgers. A mathematical model illustrating the theory of turbulence. *Advances in applied mechanics*, 1:171–199, 1948.

Diederik Johannes Korteweg and Gustav De Vries. Xli. on the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary waves. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 39(240):422–443, 1895.

Yoshiki Kuramoto. Diffusion-induced chaos in reaction systems. *Progress of Theoretical Physics Supplement*, 64:346–367, 1978.

Gi Siv Ashinsky. Nonlinear analysis of hydrodynamic instability in laminar flames—i. derivation of basic equations. In *Dynamics of Curved Fronts*, pages 459–488. Elsevier, 1988.

Jean Baptiste Joseph baron de Fourier. *Théorie analytique de la chaleur*. Firmin Didot, 1822.

John C Mason and David C Handscomb. *Chebyshev polynomials*. Chapman and Hall/CRC, 2002.

Uri M Ascher, Steven J Ruuth, and Raymond J Spiteri. Implicit-explicit runge-kutta methods for time-dependent partial differential equations. *Applied Numerical Mathematics*, 25(2-3):151–167, 1997.

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

Nasir Ahmed, T_ Natarajan, and Kamisetty R Rao. Discrete cosine transform. *IEEE transactions on Computers*, 100(1):90–93, 2006.

H V Sorensen, D Jones, Michael Heideman, and C Burrus. Real-valued fast fourier transform algorithms. *IEEE Transactions on acoustics, speech, and signal processing*, 35(6):849–863, 2003.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Appendix Contents

A Notation

In this section, we introduce frequently used notation in the main text and recall some essential facts about Fourier analysis.

In the main text, our focus is on functions defined on the periodic torus $\mathbb{T}^d$, identified as $\mathbb{T}^d = [0, 2\pi]^d$. Following standard practice, we denote by $L^2(\mathbb{T}^d)$ the space of square-integrable functions. For any such function $v \in L^2(\mathbb{T}^d)$, we define the *Fourier Transform* as,

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

For any $k \in \mathbb{Z}^d$, the k-th Fourier coefficient of v is denoted by $\widehat{v}_k = \mathcal{F}(v)(k)$.

Given a set of Fourier coefficients $\{\widehat{v}_k\}_{k\in\mathbb{Z}^d}$, the *inverse Fourier Transform* is defined as,

$$\mathcal{F}^{-1}(\widehat{v})(x) := \sum_{k\in\mathbb{Z}^d} \widehat{v}_k e^{i\langle k,x\rangle}, \quad \forall\, x \in \mathbb{T}^d. \tag{A.2}$$

Using the Fourier transform (A.1) and for $s \geq 0$, we denote by $H^s(\mathbb{T}^d)$ the *Sobolev space* of functions $v \in L^2(\mathbb{T}^d)$, with Fourier coefficients $\{\widehat{v}_k\}_{k\in\mathbb{Z}^d}$, having a finite H^s-norm:

$$\|v\|_{H^s}^2 := \frac{(2\pi)^d}{2} \sum_{k\in\mathbb{Z}^d} (1 + |k|^{2s})|\widehat{v}_k|^2 < \infty. \tag{A.3}$$

Note that with this definition, we have from Parseval's identity, that $\|v\|_{H^0} = \|v\|_{L^2}$, so that $H^0(\mathbb{T}^d) = L^2(\mathbb{T}^d)$.

We also introduce the corresponding *homogeneous Sobolev spaces* $\dot{H}^s(\mathbb{T}^d)$ (and $\dot{L}^2(\mathbb{T}^d) := \dot{H}^0(\mathbb{T}^d)$), consisting of functions $v(x) \in H^s(\mathbb{T}^d)$ with zero mean $⨍_{\mathbb{T}^d} v(x)\, dx = \widehat{v}_0 = 0$, and with norm

$$\|v\|_{\dot{H}^s} := \left((2\pi)^d \sum_{k\in\mathbb{Z}^d\setminus\{0\}} |k|^{2s} |\widehat{v}_k|^2 \right)^{1/2}. \tag{A.4}$$

Given $N \in \mathbb{N}$, throughout this work, we will denote by $L^2_N(\mathbb{T}^d)$ the space of trigonometric polynomials $v_N : \mathbb{T}^d \to \mathbb{R}$, of the form

$$v_N(x) = \sum_{|k|_\infty \leq N} c_k e^{i\langle x,k\rangle}, \tag{A.5}$$

where the summation is over all $k = (k_1, \ldots, k_d) \in \mathbb{Z}^d$ such that

$$|k|_\infty := \max_{i=1,\ldots,d} |k_i| \leq N.$$

The space $L^2_N(\mathbb{T}^d)$ is viewed as a normed vector space with norm $\|\cdot\|_{L^2}$. Similarly, for $s \geq 0$, we denote by $H^s_N(\mathbb{T}^d)$ the normed vector space of trigonometric polynomials v_N of degree $\leq N$, with norm $\|\cdot\|_{H^s}$.

We note that in order to ensure that $v_N(x) \in \mathbb{R}$ is real-valued for all $x \in \mathbb{T}^d$, the coefficients $c_k \in \mathbb{C}$ must satisfy the relations $c_{-k} = \overline{c_k}$ for all $|k|_\infty \leq N$, and where $\overline{c_k}$ denotes the complex conjugate of c_k.

We denote by

$$P_N : L^2(\mathbb{T}^d) \to L^2_N(\mathbb{T}^d), \quad v \mapsto P_N v, \tag{A.6}$$

the *L^2-orthogonal projection onto $L^2_N(\mathbb{T}^d)$*; or more explicitly,

$$P_N\left(\sum_{k\in\mathbb{Z}^d} c_k e^{i\langle k,x\rangle}\right) = \sum_{|k|_\infty \leq N} c_k e^{i\langle k,x\rangle}, \quad \forall\, (c_k)_{k\in\mathbb{Z}^d} \in \ell^2(\mathbb{Z}^d).$$

In fact, the mapping P_N defines a projection $H^s(\mathbb{T}^d) \to H^s_N(\mathbb{T}^d)$ for any $s \geq 0$. We have the following spectral approximation estimate: Let $s > 0$ be given. There exists a constant $C = C(s,d) > 0$, such that for any $v \in H^s(\mathbb{T}^d)$, we have

$$\|v - P_N v\|_{H^\varsigma} \leq C N^{-(s-\varsigma)} \|v\|_{H^s}, \quad \text{for any } \varsigma \in [0,s]. \tag{A.7}$$

We also define a natural projection

$$\dot{P}_N : L^2(\mathbb{T}^d) \to \dot{L}^2_N(\mathbb{T}^d), \tag{A.8}$$

by removing the mean, i.e. $\dot{P}_N v = P_N v - \fint_{\mathbb{T}^d} v(x)\,dx$, or equivalently:

$$\dot{P}_N\left(\sum_{k\in\mathbb{Z}^d} c_k e^{i\langle k,x\rangle}\right) = \sum_{0<|k|_\infty \leq N} c_k e^{i\langle k,x\rangle}, \quad \forall\, (c_k)_{k\in\mathbb{Z}^d} \in \ell^2(\mathbb{Z}^d).$$

Recall the set of Fourier wave numbers (5.3) and we define the **discrete Fourier transform** $\mathcal{F}_N : \mathbb{R}^{\mathcal{J}_N} \to \mathbb{C}^{\mathcal{K}_N}$ by

$$\mathcal{F}_N(v)(k) := \frac{1}{(2N+1)^d} \sum_{j\in\mathcal{J}_N} v_j e^{-2\pi i\langle j,k\rangle/N}, \tag{A.9}$$

with inverse $\mathcal{F}_N^{-1} : \mathbb{C}^{\mathcal{K}_N} \to \mathbb{R}^{\mathcal{J}_N}$,

$$\mathcal{F}_N^{-1}(\widehat{v})(j) := \sum_{k\in\mathcal{K}_N} \widehat{v}_k e^{2\pi i\langle j,k\rangle/N}. \tag{A.10}$$

B Detailed Proofs and Technical Results for Section 5

B.1 Proof of Lemma 5.3

The demonstration of Lemma 5.3 requires the following auxiliary technical result:

***Lemma* B.1.** Let $s' \geq 0$ and $N \in \mathbb{N}$ be given parameters. Consider a compact set $K \subset H^{s'}$ and assume that the activation function satisfies $\sigma \in C^m$ for some integer $m > s'$. Then for any prescribed tolerance $\epsilon > 0$, there exists a single-layer FNO $\mathcal{L} : H^{s'} \to H^{s'}$ such that

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

Proof of Lemma B.1. We begin by noting that the Fourier projection operator $P_N : H^{s'} \to H^{s'}$ is continuous, which implies that the image set $P_N K \subset H^{s'}$ is compact. Moreover, P_N maps into a

finite-dimensional subspace of $H^{s'}$. Due to the equivalence of norms on finite-dimensional spaces, there exists a constant $C_0 = C_0(N, K) > 0$ such that

$$\sup_{v\in K} \|P_N v\|_{L^\infty} \le C_0, \quad \sup_{v\in K} \|P_N v\|_{H^m} \le C_0. \tag{B.1}$$

Let $x_0 \in \mathbb{R}$ be chosen such that $\sigma'(x_0) \neq 0$. For any parameter $h > 0$, we define the function

$$\psi_h(x) := \frac{\sigma(x_0 + hx) - \sigma(x_0 - hx)}{2h\sigma'(x_0)}. \tag{B.2}$$

One can readily verify that $\psi_h \in C^m$, and there exists a constant $C_1 = C_1(\sigma, C_0) > 0$ such that

$$\|\psi_h\|_{C^m([-C_0,C_0])} \le C_1, \quad \forall\, h \in (0, 1]. \tag{B.3}$$

Furthermore, applying Taylor expansion yields

$$|\psi_h(x) - x| \le Ch, \quad \forall\, x \in [-C_0, C_0],\ \forall\, h \in (0, 1]. \tag{B.4}$$

By the composition rule for Sobolev functions, we have $\psi_h \circ P_N a \in H^m$ whenever $P_N a \in H^m$, and there exists a constant $C_2 = C_2(C_1, C_0) > 0$ such that

$$\|\psi_h(P_N v)\|_{H^m} \le C_2, \quad \forall\, v \in K. \tag{B.5}$$

We observe that the mapping $v \mapsto \mathcal{L}_h(v) := \psi_h(P_N v)$ can be implemented as a single-layer FNO, and by (B.5), we obtain

$$\|\mathcal{L}_h(v)\|_{H^m} \le C_2, \quad \forall\, v \in K. \tag{B.6}$$

Using the interpolation inequality between Sobolev spaces, we have

$$\|\mathcal{L}_h(v) - P_N v\|_{H^{s'}} \le \|\mathcal{L}_h(v) - P_N v\|_{L^2}^{\theta} \|\mathcal{L}_h(v) - P_N v\|_{H^m}^{1-\theta},$$

where $\theta = 1 - s'/m > 0$. The second factor can be bounded independently of $h > 0$ using (B.6) and (B.1). By (B.4) and (B.1), we obtain

$$\|\mathcal{L}_h(v) - P_N v\|_{L^2} = \|\psi_h(P_N v) - P_N v\|_{L^2} \le Ch,$$

for some constant $C > 0$ independent of both h and $v \in K$. We conclude that

$$\|\mathcal{L}_h(v) - P_N v\|_{H^{s'}} \le Ch^{\theta} \to 0,$$

as $h \to 0$, for some constant $C > 0$ independent of h. This establishes the desired result. □

We now proceed with the main proof:

Proof of Lemma 5.3. Let $\mathcal{G} : H^s \to H^{s'}$ be a continuous operator, and let $K \subset H^s$ be a compact set. We assume that FNOs provide universal approximation for operators of the form $H^s \to L^2$, and we aim to demonstrate that for any tolerance $\epsilon > 0$, there exists an FNO approximation of $\mathcal{G} : H^s \to H^{s'}$ achieving accuracy ϵ.

We first observe that due to the compactness of $\mathcal{G}(K) \subset H^{s'}$, there exists $N \in \mathbb{N}$ such that

$$\sup_{a\in K} \|\mathcal{G}(a) - P_N \mathcal{G}(a)\|_{H^{s'}} \le \epsilon/2. \tag{B.7}$$

Let $\delta > 0$ be a parameter whose specific value will be determined at the conclusion of this proof. By our assumption regarding the universal approximation property for operators $H^s \to L^2$, there exists an FNO $\widetilde{\mathcal{N}} : H^s \to L^2$, continuous as an operator $H^s \to L^2$, such that

$$\sup_{a\in K} \|P_N \mathcal{G}(a) - \widetilde{\mathcal{N}}(a)\|_{L^2} \le \delta. \tag{B.8}$$

One challenge in our construction is that there is no guarantee that $\widetilde{\mathcal{N}}$ defines a mapping $H^s \to H^{s'}$; indeed, for $s' > s$, this is generally not the case. We address this issue by composing with an additional FNO layer $\widetilde{\mathcal{L}} : L^2 \to H^{s'}$.

By Lemma B.1, there exists a single-layer FNO $v \mapsto \widetilde{\mathcal{L}}(v)$ satisfying the identity

$$\widetilde{\mathcal{L}}(v) = \widetilde{\mathcal{L}}(P_N v), \tag{B.9}$$

for all v, and defining a continuous operator $H^{s'} \to H^{s'}$, such that

$$\sup_{v \in K'} \|P_N v - \widetilde{\mathcal{L}}(v)\|_{H^{s'}} \leq \delta, \tag{B.10}$$

where $K' := P_N \widetilde{\mathcal{N}}(K) \subset H^{s'}$ is a compact subset[3] of $H^{s'}$.

Next, we define a new FNO through composition $\mathcal{N} := \widetilde{\mathcal{L}} \circ \widetilde{\mathcal{N}} : H^s \to H^{s'}$. The operator $\mathcal{N}$ is continuous from H^s to $H^{s'}$, since it can be expressed as the composition

$$H^s \xrightarrow{\widetilde{\mathcal{N}}} L^2 \xrightarrow{P_N} H^{s'} \xrightarrow{\widetilde{\mathcal{L}}} H^{s'},$$

of continuous operators.

We observe that for any $a \in K$, the following bound holds:

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

where we have utilized the inequality $\|P_N v\|_{H^{s'}} \leq C N^{s'} \|P_N v\|_{L^2}$ for a constant $C = C(\mathbb{T}^d, s') > 0$ independent of N, and the fact that $\mathcal{N}(a) = \widetilde{\mathcal{L}}(\widetilde{\mathcal{N}}(a)) = \widetilde{\mathcal{L}}(P_N \widetilde{\mathcal{N}}(a))$ (see (B.9)).

Using (B.8), we can estimate

$$C N^{s'} \|P_N \mathcal{G}(a) - \widetilde{\mathcal{N}}(a)\|_{L^2} \leq C N^{s'} \delta.$$

The bound (B.10) implies that $\|P_N \widetilde{\mathcal{N}}(a) - \widetilde{\mathcal{L}}(P_N \widetilde{\mathcal{N}}(a))\|_{H^{s'}} = \|P_N v - \widetilde{\mathcal{L}}(v)\|_{H^{s'}} \leq \delta$ by (B.10), with $v := P_N \widetilde{\mathcal{N}}(a) \in K'$. Thus we obtain

$$\|P_N \mathcal{G}(a) - \mathcal{N}(a)\|_{H^{s'}} \leq (C N^{s'} + 1)\delta, \tag{B.11}$$

where $C = C(\mathbb{T}^d, s') > 0$ is independent of δ. Since $\delta > 0$ is arbitrary, we can ensure that $(C N^{s'} + 1)\delta \leq \epsilon/2$.

From this estimate, combined with the bound (B.7), we conclude that there exists an FNO $\mathcal{N} : H^s \to H^{s'}$ such that

$$\begin{aligned}\sup_{a \in K} \|\mathcal{G}(a) - \mathcal{N}(a)\|_{H^{s'}} &\leq \sup_{a \in K} \|\mathcal{G}(a) - P_N \mathcal{G}(a)\|_{H^{s'}} \\ &\quad + \sup_{a \in K} \|P_N \mathcal{G}(a) - \mathcal{N}(a)\|_{H^{s'}} \\ &\leq \epsilon.\end{aligned}$$

This completes our proof. □

B.2 Proof of Lemma 5.4

Proof. **Step 1:** In this initial step, for any given tolerance $\epsilon > 0$, we construct an FNO

$$\mathcal{N}_1 : L^2(\mathbb{T}^d; \mathbb{R}) \to L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}),$$

such that

$$\begin{cases} \|\mathcal{N}_1(v)_{1,k} - P_N v(x)\, \cos(\langle k, x \rangle)\|_{L^\infty} < \epsilon, \\ \|\mathcal{N}_1(v)_{2,k} - P_N v(x)\, \sin(\langle k, x \rangle)\|_{L^\infty} < \epsilon, \end{cases} \quad \forall\, k \in \mathcal{K}_N. \tag{B.12}$$

To construct such an operator $\mathcal{N}_1$, we first define a lifting mapping

$$\mathcal{R}_1 : L^2(\mathbb{T}^d; \mathbb{R}) \to L^2(\mathbb{T}^d; \mathbb{R}^{4\mathcal{K}_N}), \quad v(x) \mapsto \widehat{w}(x),$$

[3]We note that $\widetilde{\mathcal{N}}(K) \subset L^2$ is compact as the continuous image of K, and that $P_N : L^2 \to H^{s'}$ defines a continuous mapping for fixed $N \in \mathbb{N}$.

where $\widehat{w}(x) := \{(v(x), 0, v(x), 0)\}_{k\in\mathcal{K}_N} \in \mathbb{R}^{4\mathcal{K}_N}$ for any $x \in \mathbb{T}^d$. In what follows, we identify $\mathbb{R}^{4\mathcal{K}_N} \simeq \mathbb{R}^{d_v}$ with $d_v = 4|\mathcal{K}_N|$.

We then define the inner components of the first FNO layer (specifically the matrix W, multiplier P, and bias $b(x)$ in (4.2)) such that $P(k) \equiv 1_{|k|\le N}\mathbf{1}_{d_v\times d_v}$ either vanishes (for $|k| > N$) or equals the identity matrix (for $|k| \le N$), $W \equiv 0$ is the zero matrix, and the bias function is $b(x) := \{(0, b_{2,k}(x), 0, b_{4,k}(x))\}_{k\in\mathcal{K}_N}$ with $b_{2,k}(x) = \cos(\langle k, x\rangle)$ and $b_{4,k}(x) = \sin(\langle k, x\rangle)$, yielding

$$\begin{aligned}\widehat{\mathcal{L}}_1(\widehat{v})(x) &:= W\widehat{w}(x) + b(x) + \mathcal{F}^{-1}(P\mathcal{F}\widehat{w})(x)\\ &= \big\{\big(\, P_N\widehat{w}_{1,k}(x),\, \cos(\langle k, x\rangle),\, P_N\widehat{w}_{3,k}(x),\, \sin(\langle k, x\rangle)\,\big)\big\}_{k\in\mathcal{K}_N}.\end{aligned}$$

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

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

where $C = C(N) \propto N^{d/2}$ is a constant depending on N.

By the universal approximation theorem for standard neural networks, there exists a neural network

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

with activation function σ such that $\max_{(a,b,c,d)} |\widehat{\mathcal{N}}(a, b, c, d) - (ab, cd)| < \epsilon$, where the maximum is taken over all $a, c \in [-CB, CB]$ and $b, d \in [-1, 1]$.

Then the pointwise mapping

$$\mathcal{N}_1 = \widehat{\mathcal{N}} \circ \widehat{\mathcal{L}}_1 \circ \mathcal{R}_1 : \begin{cases} L^2(\mathbb{T}^d; \mathbb{R}) \to L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}),\\ v(x) \mapsto \widetilde{w}(x) := \widehat{\mathcal{L}}_1(\mathcal{R}_1(v))(x) \mapsto \widehat{\mathcal{N}}(\widetilde{w}(x)),\end{cases}$$

satisfies (B.12), and $\mathcal{N}_1$ can be represented as an FNO (see Remark 4.1).

Step 2: From the previous step, we have $\|\mathcal{N}_1(v)_{1,k} - P_Nv(x)\, \cos(\langle k, x\rangle)\|_{L^\infty} < \epsilon$ and $\|\mathcal{N}_1(v)_{2,k} - P_Nv(x)\, \sin(\langle k, x\rangle)\|_{L^\infty} < \epsilon$ for all $v \in L^2(\mathbb{T}^d)$ with $\|v\|_{L^2} \le B$.

We observe that since

$$P_Nv(x) = \sum_{|k|_\infty\le N} \widehat{v}_k e^{i\langle k,x\rangle},$$

and since $v(x)$ is real-valued, the Fourier coefficients $\widehat{v}_k$ satisfy $\widehat{v}_{-k} = \overline{\widehat{v}_k}$, where $\overline{\widehat{v}_k}$ denotes the complex conjugate of $\widehat{v}_k$, and

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

In particular, this implies that the zeroth Fourier modes of $P_Nv(x)\, \cos(\langle k, x\rangle)$ and $P_Nv(x)\, \sin(\langle k, x\rangle)$, respectively, are given by

$$\mathcal{F}\Big[P_Nv\cos(\langle k, \cdot\rangle)\Big](0) = \mathrm{Re}(\widehat{v}_k), \quad \mathcal{F}\Big[P_Nv\sin(\langle k, \cdot\rangle)\Big](0) = -\mathrm{Im}(\widehat{v}_k).$$

Consequently, using the Fourier multiplier

$$\delta_0(k') = \begin{cases}1, & (k' = 0)\\ 0, & (k' \ne 0),\end{cases} \qquad \forall\, k' \in \mathbb{Z}^d,$$

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

$$\begin{aligned}\Big\|\mathcal{F}^{-1}\Big(\delta_0(k')\mathcal{F}(w_{1,k})(k')\Big)(x) - \mathrm{Re}(\widehat{v}_k)\Big\|_{L^\infty} < \epsilon,\\ \Big\|\mathcal{F}^{-1}\Big(\delta_0(k')\mathcal{F}(w_{2,k})(k')\Big)(x) + \mathrm{Im}(\widehat{v}_k)\Big\|_{L^\infty} < \epsilon,\end{aligned}$$

for all $v \in L^2(\mathbb{T}^d)$ with $\|v\|_{L^2} \leq B$.

We use this observation to define an appropriate FNO layer with local matrix $W = 0$, bias $b(x) = 0$, and a Fourier multiplier matrix $P : \mathbb{Z}^d \to \mathbb{C}^{d_v \times d_v}$, $k' \mapsto P(k')$ (where $d_v = 2|\mathcal{K}_N|$ and $\mathbb{C}^{d_v} \simeq \mathbb{C}^{2\mathcal{K}_N}$), with entries

$$[P(k')]_{(\ell,k),(\widetilde{\ell},\widetilde{k})} := \delta_0(k') \left\{\delta_{\ell=1}(\ell) - \delta_{\ell=2}(\ell)\right\} \mathbf{1}_{d_v \times d_v}.$$

With this definition of the Fourier multiplier P, we define $\widehat{\mathcal{L}}_2 : L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}) \to L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N})$ by

$$\widehat{\mathcal{L}}_2(w) := Ww(x) + b(x) + \mathcal{F}^{-1}\Big(P\mathcal{F}w\Big)(x).$$

Then, by construction, for any $v \in L^2(\mathbb{T}^d)$ with $\|v\|_{L^2} \leq B$, the output $w := \widehat{\mathcal{L}}_2(\mathcal{N}_1(v))$ with $w(x) = \{(w_{1,k}(x), w_{2,k}(x))\}_{k \in \mathcal{K}_N}$ satisfies

$$\begin{cases} \|w_{1,k}(x) - \mathrm{Re}(\widehat{v}_k)\|_{L^\infty} < \epsilon, \\ \|w_{2,k}(x) - \mathrm{Im}(\widehat{v}_k)\|_{L^\infty} < \epsilon. \end{cases}$$

We also note that $\widehat{\mathcal{L}}_2$ outputs only constant functions by construction.

This is nearly the desired result, except that the composition $\widehat{\mathcal{L}}_2 \circ \mathcal{N}_1$ does not define an FNO since the linear layer $\widehat{\mathcal{L}}_2$ lacks the nonlinearity σ. This can be remedied by composition with an appropriate standard σ-neural network $\widetilde{\mathcal{N}}$ that approximates the identity function.

Indeed, the last estimate implies that $|w_{\ell,k}| \leq |\widehat{v}_k| + \epsilon \leq B + \epsilon$ for all ℓ and k. By the standard universal approximation theorem, there exists a neural network $\widetilde{\mathcal{N}} : \mathbb{R}^{2\mathcal{K}_N} \to \mathbb{R}^{2\mathcal{K}_N}$ such that

$$\|\widetilde{\mathcal{N}}(w) - w\|_{\ell^\infty} < \epsilon, \quad \forall w \in \mathbb{R}^{2\mathcal{K}_N},\ \|w\|_{\ell^\infty} \leq B + \epsilon.$$

Then the composition $\mathcal{N}_2 := \widetilde{\mathcal{N}} \circ \widehat{\mathcal{L}}_2 : L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}) \to L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N})$, given by $w(x) \mapsto \widetilde{\mathcal{N}}\left(\widehat{\mathcal{L}}_2(w)(x)\right)$, *does* define an FNO.

Step 3: We finally observe that since both $\mathcal{N}_1$ and $\mathcal{N}_2$ can be represented as FNOs, their composition

$$\mathcal{N} := \mathcal{N}_2 \circ \mathcal{N}_1 : L^2(\mathbb{T}^d) \to L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N})$$

can also be represented as an FNO with $\mathrm{depth}(\mathcal{N}) = \mathrm{depth}(\mathcal{N}_1) + \mathrm{depth}(\mathcal{N}_2)$, $\mathrm{width}(\mathcal{N}) = \max_{j=1,2} \mathrm{width}(\mathcal{N}_j)$, and $\mathrm{lift}(\mathcal{N}) = \max_{j=1,2} \mathrm{lift}(\mathcal{N}_j)$.

Furthermore, the mapping $v \mapsto \mathcal{N}(v)$ produces constant functions, and for any $v \in L^2(\mathbb{T}^d)$ with $\|v\|_{L^2} \leq B$, we have

$$\begin{cases} \|\mathrm{Re}(\widehat{v}_k) - \mathcal{N}(v)_{1,k}\|_{L^\infty} < 2\epsilon, \\ \|\mathrm{Im}(\widehat{v}_k) - \mathcal{N}(v)_{2,k}\|_{L^\infty} < 2\epsilon. \end{cases}$$

Since $\epsilon > 0$ was arbitrary, the claim follows. □

B.3 Proof of Lemma 5.5

Proof. **Step 1:** We begin by constructing an FNO $\mathcal{N}_1 : L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}) \to L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N})$ such that

$$\begin{cases} \|\mathcal{N}_1(w)_{1,k} - P_N w_{1,k}(x) \cos(\langle k, x\rangle)\|_{L^\infty} < \left(2|\mathcal{K}_N||\mathbb{T}^d|^{1/2}\right)^{-1} \epsilon, \\ \|\mathcal{N}_1(w)_{2,k} - P_N w_{2,k}(x) \sin(\langle k, x\rangle)\|_{L^\infty} < \left(2|\mathcal{K}_N||\mathbb{T}^d|^{1/2}\right)^{-1} \epsilon, \end{cases} \tag{B.13}$$

for all $w(x) = (w_{1,k}(x), w_{2,k}(x))$ such that $\|w_{1,k}\|_{L^\infty}$, $\|w_{2,k}\|_{L^\infty} \leq B$ for all $k \in \mathcal{K}_N$. Here $|\mathbb{T}^d| = (2\pi)^d$ denotes the Lebesgue measure of $\mathbb{T}^d$ and $|\mathcal{K}_N| = (2N+1)^d$ represents the cardinality of $\mathcal{K}_N$.

To construct such an operator $\mathcal{N}_1$, we first define a lifting $\mathcal{R} : L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}) \to L^2(\mathbb{T}^d; \mathbb{R}^{4\mathcal{K}_N})$ given by $\{(w_{1,k}(x), w_{2,k}(x))\}_{k \in \mathcal{K}_N} \mapsto \{(w_{1,k}(x), 0, w_{2,k}(x), 0)\}_{k \in \mathcal{K}_N}$, followed by a linear layer

$\mathcal{L} : L^2(\mathbb{T}^d; \mathbb{R}^{4\mathcal{K}_N}) \to L^2(\mathbb{T}^d; \mathbb{R}^{4\mathcal{K}_N})$ that introduces only a bias term (setting $W \equiv 0$, $b(x) = \{(0, \cos(\langle k, x\rangle), 0, \sin(\langle k, x\rangle))\}_{k\in\mathcal{K}_N}$, $P \equiv 0$), yielding

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

for all $w \in L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N})$.

There exists a standard neural network $\widehat{\mathcal{N}} : [-B, B] \times [-1, 1] \times [-B, B] \times [-1, 1] \to \mathbb{R}^2$ such that $\max_{a,b,c,d} |\widehat{\mathcal{N}}(a, b, c, d) - (ab, cd)| < (2|\mathcal{K}_N||\mathbb{T}^d|^{1/2})^{-1}\epsilon$, where the maximum is taken over $a, c \in [-B, B]$ and $b, d \in [-1, 1]$. Since by assumption we have $\|w_{1,k}\|_{L^\infty}, \|w_{2,k}\|_{L^\infty} \leq B$ for the inputs of interest, we conclude that $\mathcal{N}_1(w) := \widehat{\mathcal{N}} \circ \mathcal{L} \circ \mathcal{R}(w)$ satisfies (B.13).

Step 2: We define a projection operator $\mathcal{Q}$ (similar to the definition of neural operators (4.3)) that maps $w(x) \in L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N})$ to a scalar-valued function:

$$\mathcal{Q} : L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}) \to L^2(\mathbb{T}^d), \quad w(x) \mapsto \sum_{k\in\mathcal{K}_N} (w_{1,k}(x) - w_{2,k}(x)) .$$

Then $\mathcal{N} := \mathcal{Q} \circ \mathcal{N}_1 : L^2(\mathbb{T}^d; \mathbb{R}^{2\mathcal{K}_N}) \to L^2(\mathbb{T}^d)$ is an FNO, and for any $v \in L^2_N(\mathbb{T}^d)$ and $w = \{(\mathrm{Re}(\widehat{v}_k), \mathrm{Im}(\widehat{v}_k))\}_{k\in\mathcal{K}_N}$ defined as in the statement of this lemma, we have

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

□

B.4 Proof of Remark 6.2

We maintain the notation established in Section 3.2. Assuming the aliasing error $\varepsilon(U, u, \Psi, \Phi)$ vanishes, we have the following relationship:

$$U = T_\Phi \circ u(\Psi, \Phi) \circ T_\Psi^\dagger. \tag{B.14}$$

From equation (B.14), it directly follows that

$$T_\Phi^\dagger \circ U \circ T_\Psi = T_\Phi^\dagger \circ T_\Phi \circ u(\Psi, \Phi) \circ T_\Psi^\dagger \circ T_\Psi = u(\Psi, \Phi),$$

where the final equality holds because $T_\Psi^\dagger \circ T_\Psi$ represents the orthogonal projection onto the orthogonal complement of the kernel of T_Ψ, denoted $(\mathrm{Ker}(T_\Psi))^\perp$, which is equivalent to the range of $T_\Psi^\dagger$, i.e., $\mathrm{Ran}(T_\Psi^\dagger)$. By our initial assumption, $u(\Psi, \Phi)$ maps the range of $T_\Psi^\dagger$ into the range of $T_\Phi^\dagger$. This completes the proof of Remark 6.2.

B.5 SNOs are aliasing-free

We begin by establishing the mathematical foundation of the architecture of Spectral Neural Operators (SNOs). For a positive integer $K > 0$, we define

$$\mathcal{P}_K = \left\{ g(x) = \sum_{k=-K}^{K} c_k e^{i\pi kx} : \{c_k\}_{k=-K}^K \in \mathbb{C}^{2K+1} \right\},$$

which represents the function space of signals that are 2-periodic and have their frequency content limited to πK. The collection $\Psi_K = \{e^{i\pi k\cdot}\}_{k=-K}^{K}$ forms an orthonormal basis for the space $\mathcal{P}_K$. We can define the associated synthesis operator $T_{\Psi_K}: \mathbb{C}^{2K+1} \to \mathcal{P}_K$ and its adjoint analysis operator $T_{\Psi_K}^*: \mathcal{P}_K \to \mathbb{C}^{2K+1}$ as follows:

$$T_{\Psi_K}(\{c_k\}_{k=-K}^{K}) = \sum_{k=-K}^{K} c_k e^{i\pi k\cdot}, \quad T_{\Psi_K}^* f = \{\langle f, e^{i\pi k\cdot}\rangle\}_{k=-K}^{K}.$$

The spectral neural operator architecture is constructed as a composition of three mappings: $T_{\Psi_{K'}} \circ \mathcal{N} \circ T_{\Psi_K}^*$, where the middle component $\mathcal{N}: \mathbb{C}^{2K+1} \to \mathbb{C}^{2K'+1}$ represents a standard feedforward neural network equipped with activation function σ. This network is structured as:

$$\mathcal{N}(x) = W^{(L+1)}\sigma(W^{(L)}\cdots\sigma(W^{(2)}\sigma(W^{(1)}x - b^{(1)}) - b^{(2)})\cdots - b^{(L)}) - b^{(L+1)} \tag{B.15}$$

where $W^{(\ell)}$ denotes the weight matrices and $b^{(\ell)}$ represents the bias vectors for layers $\ell = 1, \ldots, L+1$.

The complete architectural framework can be visualized through the following commutative diagram:

$$\begin{array}{ccccccc}
\mathcal{P}_K & \xrightarrow{T_{\Psi_K}^*} & \mathbb{C}^{2K+1} & \xrightarrow{\mathcal{N}} & \mathbb{C}^{2K'+1} & \xrightarrow{T_{\Psi_{K'}}} & \mathcal{P}_{K'} \\
{\scriptstyle T_{\Psi_K}}\uparrow\downarrow{\scriptstyle T_{\Psi_K}^*} & & \uparrow\downarrow \mathrm{Id} & & \uparrow\downarrow \mathrm{Id} & & {\scriptstyle T_{\Psi_{K'}}}\uparrow\downarrow{\scriptstyle T_{\Psi_{K'}}^*} \\
\mathbb{C}^{2K+1} & \xrightarrow{\mathrm{Id}} & \mathbb{C}^{2K+1} & \xrightarrow{\mathcal{N}} & \mathbb{C}^{2K'+1} & \xrightarrow{\mathrm{Id}} & \mathbb{C}^{2K'+1}
\end{array}$$

When implemented in discrete form, spectral neural operators reduce to conventional feedforward neural networks that operate by transforming one set of Fourier coefficients into another. To establish that SNOs qualify as ReNOs within the context of function spaces $\mathcal{P}_K$ and $\mathcal{P}'_K$, it becomes necessary to ensure that equation (6.3) holds for more general frame sequence selections. Furthermore, the SNO framework admits natural extensions to accommodate arbitrary frames within finite-dimensional inner-product spaces.