CoNO: Complex Neural Operator for Learning the Dynamics of Continuous Physical Systems

Anonymous Authors[1]

Abstract

Neural operators extend data-driven models to map between infinite-dimensional functional spaces. Nevertheless, while these operators effectively operate in either the time or frequency domain, their performance may be limited when applied to non-stationary spatial or temporal signals. To address this challenge, we introduce a Complex Neural Operator (CoNO) that parameterizes the integral kernel using Fractional Fourier Transform (FrFT), better representing nonstationary signals in a complex domain. We perform an extensive empirical evaluation of CoNO on seven challenging partial differential equations (PDEs), including regular grids, structured meshes, and point clouds. Empirically, CoNO consistently attains state-of-the-art performance, showcasing an average relative gain of 10.9%. Further, CoNO exhibits superior performance, outperforming all other models in additional tasks such as zero-shot super-resolution and robustness to noise. Moreover, CoNO exhibits the ability to learn from small amounts of data—giving the same performance as the next best model with just 60% of the training data. Finally, CoNO exhibits increased stability in long-horizon prediction. Altogether, CoNO presents a robust and superior model for modeling continuous dynamical systems, providing a fillip to scientific machine learning.

1. Introduction

Continuum systems span various scientific and engineering fields, such as solid mechanics, robotics, biological systems, climatic modeling, and fluid dynamics, among others (Debnath & Debnath, 2005). These systems are mathematically represented using Partial Differential Equations (PDEs) that

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

are numerically solved to obtain the system's time evolution. The solution of PDEs necessitates the identification of an optimal solution operator, which maps from functional spaces encompassing initial conditions and coefficients. Achieving this mapping entails discretization procedures for the data capture. Traditionally, numerical methods, such as finite element and spectral methods, have been employed to approximate the solution operator for PDEs. However, these approaches often incur high computational costs and exhibit limited adaptability to arbitrary resolutions and geometries (Sewell, 2012). Such high computational costs of these numerical methods inhibit the real-time prediction crucial in computer graphics, weather forecasting, and robotics.

Neural networks have been proposed to solve PDEs through a data-driven approach. Neural operators represent an extension of neural networks, facilitating the mapping between infinite-dimensional functional spaces and serving as a universal approximation of the operator. Notably, these operators operate without prior knowledge of the underlying PDE, relying solely on data-driven training, leading to faster inference times than the traditional methods (Li et al., 2020; Kovachki et al., 2021).

In the realm of operator learning, the Fourier Neural Operator (FNO) (Li et al., 2020) has gained widespread recognition for its ability to navigate the infinite-dimensional functional space via kernel integral operations in the Fourier domain. Renowned for its versatility, FNO has found successful applications across diverse domains, including weather forecasting (Kurth et al., 2023), biomedical surrogate modeling (Guan et al., 2021), and expediting sampling processes in diffusion models (Zheng et al., 2023). Nevertheless, recent investigations have brought to light some specific challenges associated with FNO, including aliasing errors (Fanaskov & Oseledets, 2022), a departure from continuous-discrete equivalence (Bartolucci et al., 2023), susceptibility to vanishing gradient problems with an increasing number of layers (Tran et al., 2021), and suboptimal performance on time-dependent PDEs (Li et al., 2023), exemplified by the turbulent Navier-Stokes equations (chaotic flow). Notably, FNO encounters difficulties in making predictions over extended time horizons, constrained by the limitations of the Fourier Transform (FT), which is tailored for stationary
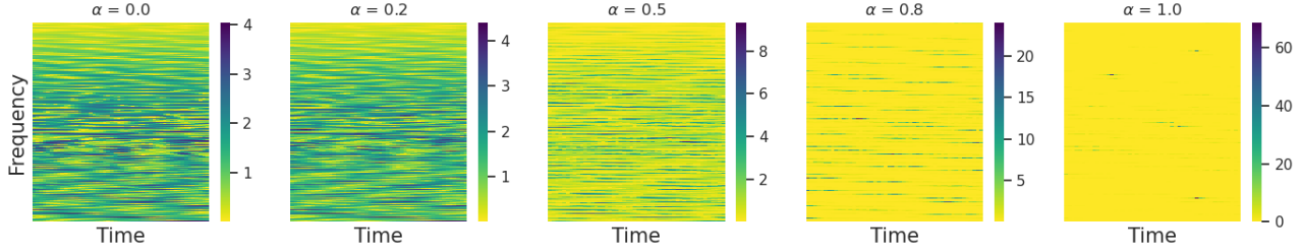
Figure 1. FrFT heatmaps illustrating the temporal-frequency characteristics of the 2D Navier-Stokes equation for varying values of α. Each subplot represents the magnitude of the transformed frequency content over time, obtained by applying the FrFT and then flattened 2D frequency map of the Navier-Stokes equation. Different subplots correspond to fractional orders α, highlighting the diverse spectral behaviors captured by the FrFT across both the temporal and frequency domains. Note that $\alpha = 0$ represents the time domain while $\alpha = 1$ represents the frequency domain.

signals and needs more time-frequency characteristics—as an aspect that has received much less attention. For instance, the change of frequency with time in the Navier-Stokes equation is non-stationary and nonlinear (see Figure 1 and Appendix A), leading to the concentration of spectrum around low frequency in the Fourier Transform. Note that the Navier-Stokes equation holds significant relevance across diverse practical domains, including but not limited to aerodynamics, weather forecasting, and oceanography. Addressing these challenges through a data-driven methodology underscores the pressing need for alternative operator formulation.

A generalization of the FT for non-stationary signals, where spectra evolve with time, by employing fractional FT (FrFT) can potentially improve the prediction of long-horizon dynamics, particularly in handling highly nonlinear and rapidly changing time-frequency characteristics. The FrFT generalizes FT by rotating the signal between the time and frequency domains, transitioning from real to complex domain, and incorporating phase information (Ozaktas & Kutay, 2001), resulting in a complex representation of the signal. However, the complex-valued representations have remained unexplored in the operator learning paradigm, although a large literature exists on complex-valued neural networks (CVNNs) (Nitta, 2002). CVNNs offer easier optimization (Nitta, 2002), faster convergence during training (Arjovsky et al., 2016; Danihelka et al., 2016; Wisdom et al., 2016), better generalization (Hirose & Yoshida, 2012), data efficiency, and resilience to noise (Danihelka et al., 2016; Wisdom et al., 2016). Thus, combining the operator paradigm with CVNNs can potentially result in an architecture that exploits both best, leading to a model that provides improved long-horizon prediction.

Our Contributions. Motivated by the above observations, we introduce a novel learning-based operator called the Complex Neural Operator (CONO), with the following desirable properties and comprehensive comparison with current SOTA operators are presented in Table 1:

1. **A complex operator architecture:** CONO represents the first instance that performs operator learning employing a CVNN that parameterizes the integral kernel through FrFT, enabling richer information through phase details learnable by CVNNs.

2. **Superior Performance:** CONO consistently exhibit superior performance compared to state-of-the-art operators, always ranking among the top two on all the datasets. Particularly, we observe that on datasets characterized by highly complex and irregular geometries such as Plasticity, CONO exhibits a gain of 31.6%.

3. **Data Efficiency:** CONO demonstrates superior performance even with limited samples, data instances, and training epochs. Specifically, CONO provides comparable performance with SOTA methods even when trained on 60% of the dataset.

4. **Robustness to noise:** CONO provides more robustness when noise is injected into the training data compared to the existing methods and performs better than SOTA methods even under 0.1% data noise.

Table 1. A comprehensive comparative analysis of features of SOTA operators with CONO. "*" denotes not applicable.

Features	FNO	LSM	CONO (Ours)
Integral Kernel	Frequency	Spatial	Spatial-Frequency
Elementary function	Sine	*	Linear Frequency Modulation
Representation	Real	Real	Complex
Pertinent Signal	Stationary	*	Time Varying Signal
α Parameter	Fixed (90°)	Fixed (0°)	Not Fixed (Learnable)
Applicability	Individual	Individual	Unified

2. Related Work

Neural Operators (NO): Neural operators have shown promise in solving PDEs in a data-driven fashion (Kovachki et al., 2021). Lu et al. (2021) introduced DeepOnet, theoretically establishing its universal approximation capabilities. The DeepOnet architecture comprises a branch network and a trunk network, with the former dedicated to learning the input function operator and the latter tasked with learning the function space onto which it is projected. Another famous architecture, the FNO, proposed by Li et al. (2020),

utilizes a frequency domain method. FNO incorporates Fourier kernel-based integral transformations through fast Fourier transform and projection blocks. An enhanced version, F-FNO (Tran et al., 2021), improves upon the original FNO by integrating distinct spectral mixing and residual connections. Subsequently, various kernel integral neural operators based on transformations have emerged. For example, Fanaskov & Oseledets (2022) introduced spectral methods, such as Chebyshev and Fourier series, to mitigate aliasing errors and enhance clarity in FNO outputs. Li et al. (2022) incorporated specialized learnable transforms to facilitate operator learning on irregular domains, achieved by transforming them into uniform latent meshes.

Kovachki et al. (2021) demonstrated that the well-known attention mechanism can be seen as a particular case of neural operator learning the integral kernel specifically applied to address irregular meshes for solving PDEs, a characteristic managed by Geo-FNO (Li et al., 2022) differently. Cao (2021) employed two self-attention mechanism-based operators without utilizing a softmax layer, accompanied by a theoretical interpretation. Recently, Hao et al. (2023) introduced the GNOT operator, featuring a linear cross-attention block designed to enhance the encoding of irregular geometries. However, transformer-based operators are susceptible to issues arising from limited data samples, displaying a tendency to overfit easily on training data without exhibiting robust generalization. In addressing the challenges posed by multiscale PDEs, Liu et al. (2022) proposed a hierarchical attention-based operator.

Fractional Fourier Transform (FrFT): The FrFT represents a generalization of the classical Fourier Transform (FT), providing robust capabilities for spectral analysis. It achieves this by facilitating the transformation of input signals into an intermediate domain between the time and frequency domains, thereby establishing a time-frequency representation (Ozaktas & Kutay, 2001). FrFT is particularly effective in processing non-stationary signals, commonly called "chirp signals" or signals exhibiting frequency derivatives over time. Figure 3 (Appendix) illustrates the efficacy of employing the FrFT for noise filtering within the signal spectrum through the rotation of the fractional order axis. In contrast to FrFT, alternative signal processing techniques such as wavelet or Gabor transformation don't provide a joint signal energy distribution across both time and frequency domains. Additionally, these alternatives often grapple with challenges related to high computational complexity, the selection of wavelet functions, and sensitivity to signal noise. FrFT, with its distinct advantages, has found applications in various domains, ranging from solving differential equations (McBride & Kerr, 1987), wireless communication (Li et al., 2018), biomedical signal processing (Gómez-Echavarría et al., 2020), image encryption and image compression (Naveen Kumar et al., 2019) etc.

Complex Valued Neural Networks (CVNNs): A complex-valued neural network incorporates complex-valued parameters and variables within its architecture, enabling the representation of both magnitude and phase information in the neural networks (Lee et al., 2022; Chiheb et al., 2017). The utilization of CVNNs encompasses diverse advantages, extending from biological to signal processing applications. Danihelka et al. (2016) has demonstrated that complex numbers enhance efficiency and stability in information retrieval processes. Additionally, Arjovsky et al. (2016) have introduced complex Recurrent Neural Networks (RNNs), highlighting that unitary matrices offer a more intricate representation, thereby mitigating issues associated with vanishing and exploding gradient problems. In image processing, phase information is a critical descriptor, offering detailed insights about image shape, orientation, and edges. Oppenheim & Lim (1981) showed the information encapsulated in the phase of an image proves sufficient for the recovery of a substantial proportion of the encoded magnitude information. Yu et al. (2024) demonstrated the benefits of employing FrFT over conventional convolution across various tasks in computer vision, including segmentation, object detection, classification, super-resolution, etc.

CVNNs have been a research focus for a long time (Georgiou & Koutsougeras, 1992; Kim & Adalı, 2003; Hirose, 2003; Nitta, 2004). Recently, Geuchen et al. (2023) established the universal approximation capabilities of CVNNs for deep, narrow architectures, significantly contributing to understanding their expressive power. Prior works (Reichert & Serre, 2013; Tygert et al., 2016; Arjovsky et al., 2016; Danihelka et al., 2016; Chatterjee et al., 2022) have made noteworthy strides in both experimental and theoretical aspects of CVNNs. In the domain of computer vision, the utilization of scattering transformation-based complex neural networks has demonstrated considerable promise, showcasing their ability to achieve performance on par with real-valued counterparts while employing significantly fewer parameters (Ko et al., 2022; Worrall et al., 2017; Rawat et al., 2021). In NLP, complex embeddings have been incorporated for semantic and phonetic processing of natural languages (Trouillon & Nickel, 2017; Demir & Ngomo, 2021). In contrast, Yang et al. (2020) and Dong et al. (2021) showcased the advantages of employing CVNNs transformers for the NLP community. Despite these notable applications across various domains, exploring the applicability of CVNNs within the SciML community is still limited.

3. Preliminaries

Problem Setting: We have followed and adopted the notations in Li et al. (2020). Let us denote a bounded open set as $D \subset \mathbb{R}^d$, with $A = A(D; \mathbb{R}^{d_a})$ and $U = U(D; \mathbb{R}^{d_u})$ as separable Banach spaces of functions, representing elements
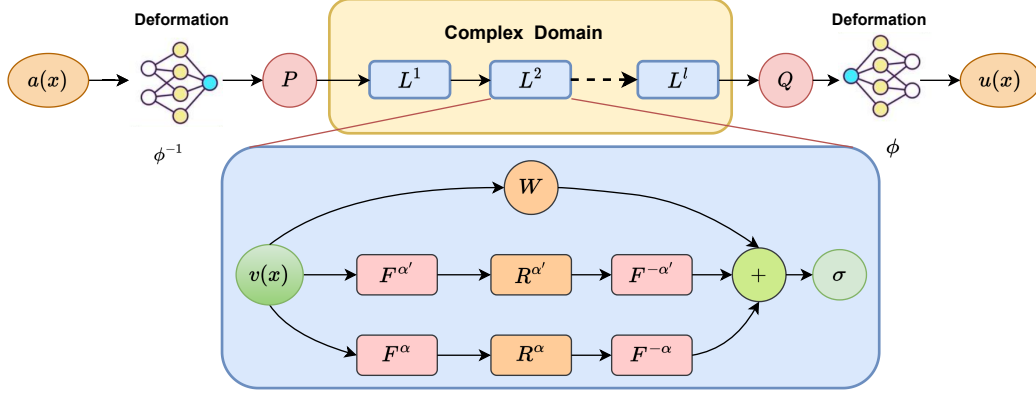
Figure 2. **CoNO architecture overview. (Top)** (1) The input function $a(x)$ undergoes a deformation ϕ^{-1} to convert an irregular mesh into a uniform mesh for operator learning. (2) The deformed input is then lifted to a higher dimension in the channel space using a NN. (3) Apply iterative CoNO Layer in complex domain consisting of fractional integral kernel. (4) Then, the output is projected back on the lower dimension in channel space using NN. (5) Solution $u(x)$ is obtained by passing through the deformation ϕ. **(Bottom)** Zoomed version of FrFT Integral Kernel defined in Equation 6 with learnable parameters R^α, $R^{\alpha'}$ and W and fractional order α and α'.

in $\mathbb{R}^{d_a}$ and $\mathbb{R}^{d_u}$, respectively. Consider $G^\dagger : A \to U$ a nonlinear surrogate mapping arising from the solution operator for a parametric PDE. It is assumed that there is access to i.i.d. observations $(a_j, u_j)_{j=1}^N$, where $a_j \sim \mu$, drawn from the underlying probability measure μ supported on A, and $u_j = G^\dagger(a_j)$.

The objective of operator learning is to construct an approximation for $G^\dagger$ via a parametric mapping $G : A \times \Theta \to U$, or equivalently, $G_\theta : A \to U, \theta \in \Theta$, within a finite-dimensional parameter space Θ. The aim is to select $\theta^\dagger \in \Theta$ such that $G(\cdot, \theta^\dagger) = G_\theta^\dagger \approx G^\dagger$. This framework facilitates learning in infinite dimensional spaces as the solution to the optimization problem in Equation 1 constructed using a with a loss function $L : U \times U \to \mathbb{R}$.

$$\min_{\theta \in \Theta} \mathbb{E}_{a \sim \mu} \left[L(G(a, \theta), G^\dagger(a)) \right], \tag{1}$$

The optimization problem is solved in operator learning frameworks using a data-driven empirical approximation of the loss function akin to the regular supervised learning approach using train-test observations. Usually, G_θ is parameterized using deep neural networks.

Fractional Fourier Transform (FrFT): Inspired by the kernel formulation for solving linear PDEs using Green's function, we construct the model G_θ employing an iterative approach to map an input function a to an output function u within the CoNO framework. In CoNO, the kernel integral is formulated using the FrFT with a learnable order. The fractional transformation of order a $(a \in \mathbb{R})$ is a parameter of the Fractional Fourier Transform (FrFT) that corresponds to the a-th power of the Fourier Transform (FT) $(\mathcal{F}^a)$.

Definition 3.1. (Ozaktas & Kutay, 2001) The fractional Fourier transform with angle α of a signal $f(x)$ is defined
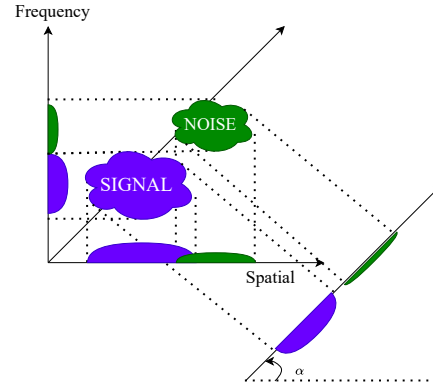

Figure 3. The figure illustrates the inherent optimality of FrFT in signal filtering for non-stationary image signals, showcasing its ability to separate effects along the fractional order axis and examining its relationship with spatial-frequency plane rotation.

as:

$$\mathcal{F}^\alpha\{f\}(x) = \int_{-\infty}^{\infty} f(y) K_\alpha(x, y)\, dt, \tag{2}$$

where,

$$K_\alpha(x, y) = \begin{cases} \frac{c(\alpha)}{\sqrt{2\pi}} \exp\{ja(\alpha)[(x^2 + y^2) - 2b(\alpha)xy]\} & \text{if } \alpha \neq 0, \frac{\pi}{2}, \pi, \\ \delta(x - y) & \text{if } \alpha = 0, \\ \delta(x + y) & \text{if } \alpha = \pi, \\ \frac{1}{\sqrt{2\pi}} e^{-jxy} & \text{if } \alpha = \frac{\pi}{2}. \end{cases}$$

with $a(\alpha) = \frac{\cot \alpha}{2}$, $b(\alpha) = \sec \alpha$, and $c(\alpha) = \sqrt{1 - j \cot \alpha}$.

Complex Valued Neural Networks (CVNNs): In the CoNO framework, the integration of kernels in the operator G_θ is performed within the complex-valued domain using CVNNs. A CVNN is modeled as real and imaginary parts or magnitude and phases as follows (Trouillon & Nickel, 2017):

$$z = x + jy = |z|e^{j\angle z} \tag{3}$$

where $j = \sqrt{-1}$ is imaginary unit, x and y are the real and imaginary parts of z, and $|z|$ and $\angle z$ are the magnitude and phase of z.

Definition 3.2 (Complex Valued Activation). Let $z \in \mathbb{C}$ be a complex number with real part $\text{Re}(z)$ and imaginary part $\text{Im}(z)$. The Complex GeLU (CGeLU) activation function is defined as follows:

$$\text{CGeLU}(z) = \text{GeLU}(\text{Re}(z)) + j \cdot \text{GeLU}(\text{Im}(z)), \quad (4)$$

where $\text{GeLU}(\cdot)$ is the Gaussian Error Linear Unit activation function (Hendrycks & Gimpel, 2016). The CGeLU activation function satisfies the Cauchy-Riemann equations when the real and imaginary parts of z are strictly positive or negative.

Complex Valued Backpropagation: Complex-valued backpropagation involves extending traditional backpropagation algorithms to handle complex numbers, utilizing mathematical tools like Wirtinger calculus (Amin et al., 2011). These changes enable training neural networks with complex-valued weights and activations, allowing for the modeling of intricate relationships within complex data domains (Barrachina et al., 2023; Chiheb et al., 2017) (App. B).

4. Complex Neural Operator (CoNO)

This subsection presents our operator framework's Complex Neural Operator (CoNO). We will provide a thorough and detailed explanation of this proposed method. Figure 2 depicts the entire architectural details.

In CoNO, G_θ is defined as an iterative sequence of functions. The operator $G_\theta : A \to U$, where $A = A(D; \mathbb{R}^{d_a})$ and $U = U(D; \mathbb{R}^{d_u})$ are separable Banach spaces of functions, representing elements in $\mathbb{R}^{d_a}$ and $\mathbb{R}^{d_u}$, respectively. Our goal is to construct the operator in a structure-preserving manner where we band-limit a function over a given spectrum (Vetterli et al.), thus preserving complex continuous-discrete equivalence such that Shannon-Whittaker-Kotel'nikov theorem is obeyed for all continuous operations (Unser, 2000).

CoNO Operator, G_θ : The operator is defined as

$$G_\theta = \phi \circ Q \circ L^l \circ \ldots \circ \sigma(L^1) \circ P \circ \phi^{-1}, \quad (5)$$

where $\circ$ denotes composition. On irregular geometries, ϕ represents the function modeled by a neural network that maps the irregular domain into a regular latent mesh for operator learning. The operators $P : \mathbb{R}^{d_a} \to \mathbb{R}^{d_1}$ and $Q : \mathbb{R}^{d_l} \to \mathbb{R}^{d_u}$ correspond to the lifting and projection operations, encoding lower-dimensional spaces into higher-dimensional spaces or vice versa which helps in converting the non-linear dynamics into linear dynamics inspired by

Koopman Operator (Bevanda et al., 2021). The operator consists of l layers of non-linear integral operators $\sigma(L^l)$ (Equation 6), where σ is an activation function which is applied from layer L^i, where i belongs to 1 to $l - 1$ to introduce non-linearity in operators, akin to standard neural networks, to learn highly nonlinear operators , and θ denotes all the learnable parameters of the operator.

Non Linear Operator Layer, L : In CoNO, the complex kernel operator is defined as follows:

$$v^{l+1}(x) = \sigma\left(Wv^l(x) + b^l + \mathcal{K}(a, \alpha)v^l(x)\right) \quad \forall x \in D \tag{6}$$

Where: v^l is the representation at layer l and $v^l \in \mathbb{C}^{d_l}$, $\mathcal{K} : V^l \times \Theta_K \to \mathcal{L}(V^{l+1}(D; \mathbb{C}^{d_l}), V^{l+1}(D; \mathbb{C}^{d_{l+1}}))$ maps to bounded linear operators on $V^{l+1}(D; \mathbb{R}^{d_{l+1}})$, σ is a non-linear complex activation function, W is a linear transformation, $W : \mathbb{C}^{d_l} \to \mathbb{C}^{d_{l+1}}$, $\mathcal{K}(a, \phi)$ is an integral kernel parameterized by CVNN (Equation 7), a is a complex input function, and α belongs to the parameter space of G_θ and b is the bias which we have omitted in Figure 2 for simplicity.

Integral Operator, $\mathcal{K}(a, \alpha)$: In CoNO, the integral operator is defined as, $\mathcal{K}(a, \alpha) : V^l(D) \to V^l(D)$ by

$$\mathcal{K}(a, \alpha)v^l(x) = \sum_{\alpha, \alpha'=1} \mathcal{F}^{-\alpha}(R^\alpha \cdot (\mathcal{F}^\alpha v^l))(x) \quad \forall x \in D \tag{7}$$

where $\mathcal{F}^\alpha$ denotes the FrFT with order α, $\mathcal{F}^{-\alpha}$ denotes the inverse FrFT, R^α is the learnable function which we learn from data, a is a complex input function, α is the learnable fractional order which we learn with data as well and α belongs to the parameter space of G_θ. With the definition above of CoNO, the operator G_θ can map between infinite-dimensional spaces and acquire the ability to learn non-linear operators as shown in 4.1 multiplication in fractional domain equivalence to spatial domain convolution.

Theorem 4.1 (**Convolution Theorem for Fractional Fourier Transform (FrFT)**). *(Zayed, 1998)*

Let us consider square integrable functions f and g, let $h(x) = (f * g)(x)$ (where $*$ denotes the convolution defined in Equation 14) and let F^α, G^α, H^α denote the FrFT of f, g and h respectively as defined in Equation 10. Then

$$H^\alpha(u) = F^\alpha(u)G^\alpha(u)e^{-ia(\alpha)u^2}. \tag{8}$$

Proof: See Appendix A.

Mitigation of Aliasing: In Neural Operator learning, the utilization of non-linear operations, such as non-linear pointwise activations, can introduce high-frequency components into the output signal. The manifestation of aliasing induced by nonlinearity can result in distortion of symmetry inherent in the physical signal, leading to undesirable effects. Additionally, the pursuit of translational invariance, a key

Table 2. The main result with sixteen baselines on all benchmarks datasets: Mean Relative ℓ_2 Error (Equation 9) is reported as the evaluation metric, where a smaller ℓ_2 Error indicates superior performance. "INCREMENT %" refers to the relative error reduction concerning the second-best model on each benchmark. Specifically focusing on the 2D Navier–Stokes benchmark, a detailed comparison is conducted with KNO (Xiong et al., 2023), and TF-Net (Wang et al., 2020), as they are designed for auto-regressive (time-dependent) tasks. Instances marked with '*' indicate that the baseline cannot handle the benchmark. In the color legend, blue represents the best performance, green indicates the second-best performance, and orange signifies the third-best performance among the baselines.

MODEL	Elasticity-P	Elasticity-G	Plasticity	Navier-Stokes	Darcy	Airfoil	Pipe
U-NET (2015)	0.0235	0.0531	0.0051	0.1982	0.0080	0.0079	0.0065
RESNET (2016)	0.0262	0.0843	0.0233	0.2753	0.0587	0.0391	0.0120
TF-NET (2020)	\	\	\	0.1801	\	\	\
SWIN (2021)	0.0283	0.0819	0.0170	0.2248	0.0397	0.0270	0.0109
DEEPONET (2021)	0.0965	0.0900	0.0135	0.2972	0.0588	0.0385	0.0097
FNO (2020)	0.0229	0.0508	0.0074	0.1556	0.0108	0.0138	0.0067
U-FNO (2022)	0.0239	0.0480	0.0039	0.2231	0.0183	0.0269	0.0056
WMT (2021)	0.0359	0.0520	0.0076	0.1541	0.0082	0.0075	0.0077
GALERKIN (2021)	0.0240	0.1681	0.0120	0.2684	0.0170	0.0118	0.0098
SNO (2022)	0.0390	0.0987	0.0070	0.2568	0.0495	0.0893	0.0294
U-NO (2022)	0.0258	0.0469	0.0034	0.1713	0.0113	0.0078	0.0100
HT-NET (2022)	0.0372	0.0472	0.0333	0.1847	0.0079	0.0065	0.0059
F-FNO (2021)	0.0263	0.0475	0.0047	0.2322	0.0077	0.0078	0.0070
KNO (2023)	\	\	\	0.2023	\	\	\
GNOT (2023)	0.0315	0.0494	*	0.1670	0.0105	0.0081	*
LSM (2023)	0.0218	**0.0408**	0.0025	0.1535	0.0065	0.0062	**0.0050**
CoNO (Ours)	**0.0210**	0.0436	**0.0019**	**0.1287**	**0.0051**	**0.0057**	0.0054
INCREMENT %	**3.8%**	-6.8%	**31.6%**	**19.3%**	**27.5%**	**8.7%**	-8.0%

characteristic in neural operators, becomes susceptible to degradation due to aliasing. We propose a two-step process to address the challenges of aliasing errors within the continuous equivariance paradigm. Firstly, before applying any activation function, we employ an upsampling operation on the input function, exceeding its frequency bandwidth. Subsequently, a non-linear operation is applied to the upsampled signal, followed by a sinc-based filter and downsampling. Incorporating the sinc-based low-pass filter effectively attenuates higher frequency components in the output signal; refer to Algorithm 1 for more mathematical details.

5. Numerical Experiments

This section provides a thorough empirical investigation of CoNO in contrast to multiple vision models and neural operator baselines. We systematically conduct extensive experiments on a diverse set of challenging benchmarks spanning various domains to demonstrate the efficacy of our proposed method.

5.1. Experiments details

Benchmarks: We assess the performance of our model on Darcy and Navier Stokes (Li et al., 2020) benchmarks to gauge its proficiency on regular grids. Subsequently, we extend our experimentation to benchmarks featuring irregular geometries, such as Airfoil, Plasticity, and Pipe (Li et al., 2022), modeled using structured meshes and Elasticity (Li et al., 2022), represented in point clouds. Refer to Appendix C for more details about benchmarks and tasks.

Baselines: We systematically assess CoNO by comparing it against sixteen established models across seven benchmarks, which include baselines from vision models (U-Net (Ronneberger et al., 2015), ResNet (He et al., 2016), SwinTransformer (Liu et al., 2021)) and thirteen baselines specifically designed for PDEs (DeepONet (Lu et al., 2021), TF-Net (Wang et al., 2020), FNO (Li et al., 2020), U-FNO (Wen et al., 2022), WMT (Gupta et al., 2021), GalerkinTransformer (Cao, 2021), SNO (Fanaskov & Oseledets, 2022), U-NO (Rahman et al., 2022), HT-Net (Liu et al., 2022), F-FNO (Tran et al., 2021), KNO (Xiong et al., 2023), GNOT (Hao et al., 2023), LSM (Wu et al., 2023)). Notably, for the Elasticity-P benchmark in the point cloud, we incorporate the specialized transformation proposed by geo-FNO (Li et al., 2022) at both the start and end of these models. This transformation facilitates the conversion of irregular input domains into or back from a uniform mesh.

Evaluation Metric: Mean relative ℓ_2 error is used throughout the experiments.

$$L = \frac{1}{N} \sum_{i=1}^{N} \frac{\|G_\theta(a_i) - G^\dagger(a_i)\|_2}{\|G^\dagger(a_i)\|_2} \quad (9)$$

the regular mean-squared error (MSE) is enhanced with a normalizer $\|G^\dagger(a_i)\|_2$ to take account for discrepancies in absolute resolution scale across different benchmarks as described in (Li et al., 2020).

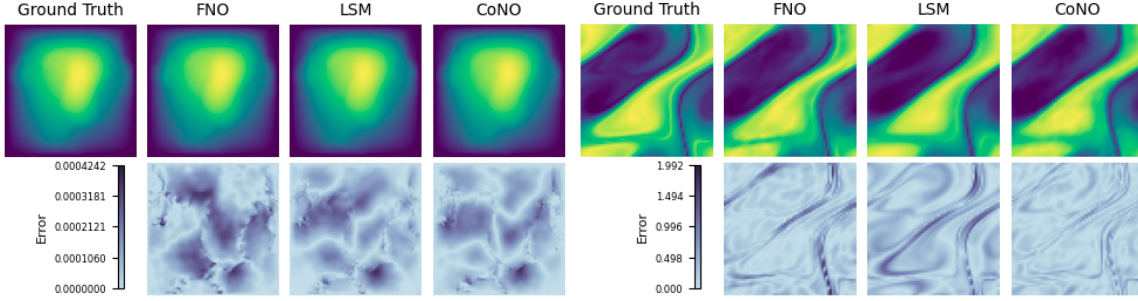**Implementation Details:** We have used Mean relative ℓ_2

Figure 4. Depiction of results for different methods on fluid datasets, Darcy Flow (**Left**) and Navier Stokes (**Right**). We plotted the heatmap of the absolute difference value between ground truth and prediction to compare the predicted output. See the Appendix D for more solid physics and fluid physics benchmarks showcases.

error (Equation 9) as the training and evaluation metric. We train all the models for 500 epochs using the Adam optimizer (Kingma & Ba, 2014). Comprehensive details are provided in the Appendix. All the experiments are conducted on a Linux machine running Ubuntu 20.04.3 LTS on an Intel(R) Core(TM) i9-10900X processor and a single NVIDIA RTX A6000 GPU with 48 GB RAM.

Empirical Results: As illustrated in Table 2, CONO consistently demonstrates superior performance on all the datasets in comparison to all baseline models, ranking among the top two. This performance superiority is evident across benchmark datasets characterized by diverse geometries and dimensions, exhibiting an average improvement of 10.9%. With the second-best performance observed in the Pipe and Elasticity-G benchmarks, our findings suggest that architectures resembling UNET demonstrate superior efficacy in capturing the underlying solution compared to the CONO model. When applied to time-dependent PDEs, CONO surpasses all previously established baselines, achieving an average improvement of 25.5%. This result underscores the efficacy of incorporating the change in frequency derivative captured by the FrFT in complex domains, thereby showcasing the promise of our approach in handling temporal dynamics in PDEs.

Table 3. Comprehensive ablation study on CONO: investigating the impact of individual component removal on Navier-Stokes and Darcy Flow benchmark (w/o denotes the performance without that component).

DESIGN	Navier-Stokes	Darcy
w/o Bias	0.1560	0.0080
w/o FrFT	0.1530	0.0086
w/o Complex NN	0.1490	0.0072
w/o Alias Free Activation	0.1290	0.0052
CONO	**0.1287**	**0.0050**

5.2. Ablation Studies

To assess the efficacy of each component in the CONO operator, we conducted a comprehensive ablation study by

systematically excluding individual components. The results presented in Table 3 indicate that all components are crucial for the effectiveness of the CONO operator, as evidenced by a notable change in the l_2 error with the addition or deletion of an element. Specifically, removing the FrFT block results in a substantial degradation in performance, underscoring its effectiveness in capturing non-stationary signals. Similarly, the absence of the CVNN leads to comparable adverse effects. Notably, our analysis reveals that including bias is instrumental in introducing high-frequency components, further emphasizing its importance. Interestingly, CONO performance degraded only slightly when the alias-free activation function is removed. It raises an intriguing question regarding its necessity for optimizing the operator's efficiency. See App.

Visual Demonstrations: For a concise representation of intuitive performance, Figure 4 presents a comparative analysis between FNO, LSM, and CONO. Notably, CONO exhibits remarkable proficiency in addressing time-dependent PDEs, including Navier-Stokes and Plasticity, as depicted in Figure 9. Moreover, CONO outperforms LSM and FNO significantly in the case of Darcy by 27% and having fewer artifacts present in prediction. Additionally, CONO excels in capturing singularities around corners, as illustrated in the elasticity dataset in the Appendix D, emphasizing its robust and superior performance.

Performance across various Resolutions: In Figure 5 (Left), the operator CONO consistently exhibits superior performance compared to other operators on darcy flow PDEs at various resolutions. Notably, CONO demonstrates stability across different resolutions, adhering to the principle of discrete-continuous equivalence. It contrasts HT-Net, which experiences degradation in very high dimensions. Furthermore, FNO and CONO represent the exclusive class of operators capable of zero-shot super-resolution without requiring explicit training.

Out of Distribution Generalization: We conducted experiments on the Navier-Stokes dataset in this investigation, training our model with a viscosity coefficient of 10^{-5}. Sub-
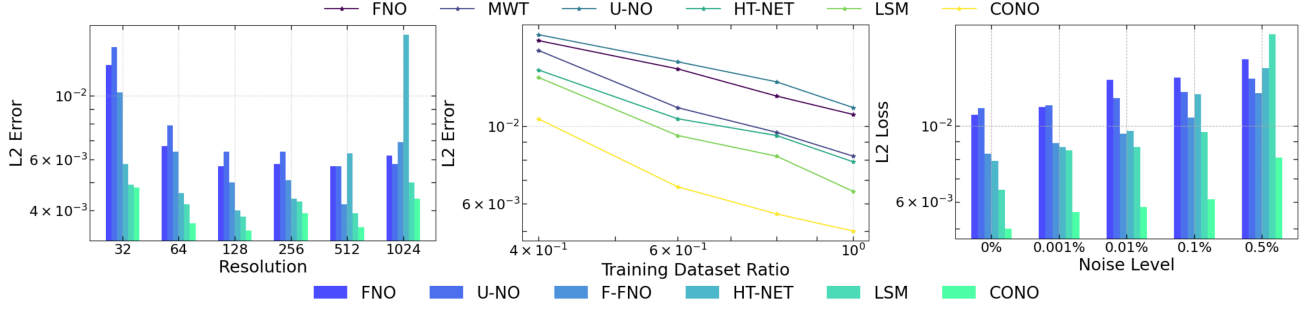
Figure 5. (**Left**) Models performance under different resolutions on Darcy. (**Middle**) Models performance under different training datasets the ratio on Darcy. (**Right**) Models performance under the presence of noise on Darcy. The lower $l2$ loss indicates better performance.

sequently, we assessed the out-of-distribution generalization capabilities by evaluating the trained model on a viscosity coefficient of 10^{-4}, as depicted in the Appendix. Our findings consistently reveal that CONO demonstrates a significantly superior generalization performance, exhibiting an increment of 64.3% compared to FNO. It also highlights the significance of capturing latent variable information or the UNET architecture, as achieved by LSM, which outperforms all other operators even CONO.

Data Efficiency: As demonstrated in Figure 5 (Middle), CONO exhibits comparable performance to the second-best operator LSM when trained on 60% of the data. Furthermore, across various training dataset ratios, CONO consistently outperforms all other operators, underscoring its superior data efficiency compared to SOTA operators.

Robustness to Noise: In this study, we performed experiments introducing different noise levels into the training using Gaussian noise. The noise addition process follows: For each input sample denoted as $x(n)$ within the dataset D, we modified it by adding Gaussian noise with parameters $\gamma N(0, \sigma_D^2)$. Here, σ_D^2 represents the variance of the entire dataset, and γ indicates the specified noise intensity level. Our investigation yielded notable results as in Figure 5 (Right), particularly when evaluating the performance of CONO in the presence of noise within the training dataset; specifically, the noisy training with 0.1% yielded a better result than the LSM operator without noisy training, confirming the robustness of CONO to the noisy dataset.
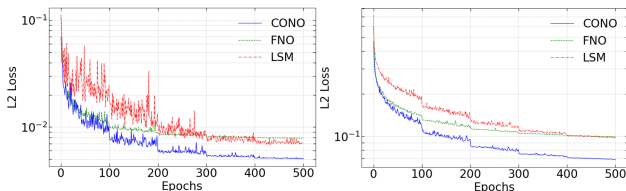

Figure 6. Learning Curve for Darcy flow (**Left**) and Navier Stokes (**Right**) where the x-axis denotes epochs and the y-axis $l2$ error.

Training Stability: The performance of CONO is more stable during training, as visually observed in Figure 6. The model exhibits reduced oscillations and consistently per-

forms better than FNO and LSM. Remarkably, CONO attains an equivalent performance compared with the best LSM performance after training within the initial 200 epochs, demonstrating its efficient faster and better convergence while training.

Long Time Prediction on Navier Stokes: We evaluated the long-term behavior of the proposed operator CONO by training it on the Navier-Stokes equation and assessing its performance using the Rolling Error metric. Compared to operators LSM and FNO, CONO consistently perform better in capturing the system's spectrum across an extended time horizon, demonstrating better stability. Furthermore, an additional experiment utilizing the Navier Stokes model with a viscosity coefficient of 10^{-4} reinforced our findings, showing that CONO excels in extrapolating beyond the prediction horizon compared to LSM and FNO as shown in the Appendix section F.4.

6. Conclusion and Future Work

Here, we introduce a novel operator learning paradigm called the CONO, which capitalizes on CVNNs and the FrFT as the integral operator. We demonstrate the effectiveness of leveraging the expressive power of CVNNs within the operator learning framework to construct resilient, data-efficient, and superior NO capable of improving the learning of function-to-function mappings. CONO surpasses existing operators in terms of performance, zero-shot superresolution, out-of-distribution generalization, and noise robustness. This advancement positions CONO as a promising approach for developing efficient operators dedicated to the real-time inference of PDEs and promises new tools for the SciML community.

Limitation and Future Work: Although CONO shows improved performance empirically, it is still not clear what exact features of CONO enable superior performance. Understanding the nature of the loss landscape and the learning mechanisms that provide superior performance is especially crucial. Further, making CONO computationally efficient is crucial to accelerate the inference. Additional limitations and future works are discussed in detail in the Appendix G.

7. Broader Impact

This paper introduces an innovative suite of tools designed to revolutionize the field of scientific machine learning (SciML). By seamlessly integrating Complex Neural Networks (CVNNs) and the Fractional Fourier Transform (FrFT) into the Neural Operator framework, we present a novel approach that tackles the intricate challenges associated with partial differential equations (PDEs). It is particularly noteworthy in Fractional PDEs, which lack explicit differential forms and are prevalent in natural phenomena (Ahmad et al., 2020; Li & Chen, 2018). Our research holds significant implications for diverse applications across biology, physics, civil engineering, and more, addressing a crucial scientific problem. It sets the stage for transformative advancements at the intersection of machine learning and interdisciplinary problem-solving.

References

Ahmad, I., Ahmad, H., Thounthong, P., Chu, Y.-M., and Cesarano, C. Solution of multi-term time-fractional pde models arising in mathematical biology and physics by local meshless method. *Symmetry*, 12(7):1195, 2020.

Amin, M. F., Amin, M. I., Al-Nuaimi, A. Y. H., and Murase, K. Wirtinger calculus based gradient descent and levenberg-marquardt learning algorithms in complex-valued neural networks. In *International Conference on Neural Information Processing*, pp. 550–559. Springer, 2011.

Arjovsky, M., Shah, A., and Bengio, Y. Unitary evolution recurrent neural networks. In *International conference on machine learning*, pp. 1120–1128. PMLR, 2016.

Barrachina, J. A., Ren, C., Vieillard, G., Morisseau, C., and Ovarlez, J.-P. Theory and implementation of complex-valued neural networks. *arXiv preprint arXiv:2302.08286*, 2023.

Bartolucci, F., de Bézenac, E., Raonić, B., Molinaro, R., Mishra, S., and Alaifari, R. Are neural operators really neural operators? frame theory meets operator learning. *arXiv preprint arXiv:2305.19913*, 2023.

Bevanda, P., Sosnowski, S., and Hirche, S. Koopman operator dynamical models: Learning, analysis and control. *Annual Reviews in Control*, 52:197–212, 2021.

Bihani, V., Pratiush, U., Mannan, S., Du, T., Chen, Z., Miret, S., Micoulaut, M., Smedskjaer, M. M., Ranu, S., and Krishnan, N. Egraffbench: Evaluation of equivariant graph neural network force fields for atomistic simulations. *arXiv preprint arXiv:2310.02428*, 2023.

Cao, S. Choose a transformer: Fourier or galerkin. *Advances in neural information processing systems*, 34: 24924–24940, 2021.

Chatterjee, S., Tummala, P., Speck, O., and Nürnberger, A. Complex network for complex problems: A comparative study of cnn and complex-valued cnn. In *2022 IEEE 5th International Conference on Image Processing Applications and Systems (IPAS)*, pp. 1–5. IEEE, 2022.

Chiheb, T., Bilaniuk, O., Serdyuk, D., et al. Deep complex networks. In *International Conference on Learning Representations*, 2017.

Danihelka, I., Wayne, G., Uria, B., Kalchbrenner, N., and Graves, A. Associative long short-term memory. In *International conference on machine learning*, pp. 1986–1994. PMLR, 2016.

Debnath, L. and Debnath, L. *Nonlinear partial differential equations for scientists and engineers*. Springer, 2005.

Demir, C. and Ngomo, A.-C. N. Convolutional complex knowledge graph embeddings. In *The Semantic Web: 18th International Conference, ESWC 2021, Virtual Event, June 6–10, 2021, Proceedings 18*, pp. 409–424. Springer, 2021.

Dong, Y., Peng, Y., Yang, M., Lu, S., and Shi, Q. Signal transformer: Complex-valued attention and meta-learning for signal recognition. *arXiv preprint arXiv:2106.04392*, 2021.

Fanaskov, V. and Oseledets, I. Spectral neural operators. *arXiv preprint arXiv:2205.10573*, 2022.

Georgiou, G. M. and Koutsougeras, C. Complex domain backpropagation. *IEEE transactions on Circuits and systems II: analog and digital signal processing*, 39(5): 330–334, 1992.

Geuchen, P., Jahn, T., and Matt, H. Universal approximation with complex-valued deep narrow neural networks. *arXiv preprint arXiv:2305.16910*, 2023.

Gómez-Echavarría, A., Ugarte, J. P., and Tobón, C. The fractional fourier transform as a biomedical signal and image processing tool: A review. *Biocybernetics and Biomedical Engineering*, 40(3):1081–1093, 2020.

Guan, S., Hsu, K.-T., and Chitnis, P. V. Fourier neural operator networks: A fast and general solver for the photoacoustic wave equation. *arXiv preprint arXiv:2108.09374*, 2021.

Gupta, G., Xiao, X., and Bogdan, P. Multiwavelet-based operator learning for differential equations. *Advances in neural information processing systems*, 34:24048–24062, 2021.

Hao, Z., Wang, Z., Su, H., Ying, C., Dong, Y., Liu, S., Cheng, Z., Song, J., and Zhu, J. Gnot: A general neural operator transformer for operator learning. In *International Conference on Machine Learning*, pp. 12556–12569. PMLR, 2023.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hendrycks, D. and Gimpel, K. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Hirose, A. *Complex-valued neural networks: theories and applications*, volume 5. World Scientific, 2003.

Hirose, A. and Yoshida, S. Generalization characteristics of complex-valued feedforward neural networks in relation to signal coherence. *IEEE Transactions on Neural Networks and learning systems*, 23(4):541–551, 2012.

Kim, T. and Adalı, T. Approximation by fully complex multilayer perceptrons. *Neural computation*, 15(7):1641–1666, 2003.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Ko, M., Panchal, U. K., Andrade-Loarca, H., and Mendez-Vazquez, A. Coshnet: A hybird complex valued neural network using shearlets. *arXiv preprint arXiv:2208.06882*, 2022.

Kovachki, N., Li, Z., Liu, B., Azizzadenesheli, K., Bhattacharya, K., Stuart, A., and Anandkumar, A. Neural operator: Learning maps between function spaces. *arXiv preprint arXiv:2108.08481*, 2021.

Kurth, T., Subramanian, S., Harrington, P., Pathak, J., Mardani, M., Hall, D., Miele, A., Kashinath, K., and Anandkumar, A. Fourcastnet: Accelerating global high-resolution weather forecasting using adaptive fourier neural operators. In *Proceedings of the Platform for Advanced Scientific Computing Conference*, pp. 1–11, 2023.

Lee, C., Hasegawa, H., and Gao, S. Complex-valued neural networks: A comprehensive survey. *IEEE/CAA Journal of Automatica Sinica*, 9(8):1406–1426, 2022.

Li, C. and Chen, A. Numerical methods for fractional partial differential equations. *International Journal of Computer Mathematics*, 95(6-7):1048–1099, 2018.

Li, Y., Song, Z., and Sha, X. The multi-weighted type fractional fourier transform scheme and its application over wireless communications. *EURASIP Journal on wireless communications and networking*, 2018(1):1–10, 2018.

Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

Li, Z., Huang, D. Z., Liu, B., and Anandkumar, A. Fourier neural operator with learned deformations for pdes on general geometries. *arXiv preprint arXiv:2207.05209*, 2022.

Li, Z., Peng, W., Yuan, Z., and Wang, J. Long-term predictions of turbulence by implicit u-net enhanced fourier neural operator. *Physics of Fluids*, 35(7), 2023.

Liu, X., Xu, B., and Zhang, L. Ht-net: Hierarchical transformer based operator learning model for multiscale pdes. *arXiv preprint arXiv:2210.10890*, 2022.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

Lu, L., Jin, P., Pang, G., Zhang, Z., and Karniadakis, G. E. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature machine intelligence*, 3(3):218–229, 2021.

McBride, A. and Kerr, F. On namias's fractional fourier transforms. *IMA Journal of applied mathematics*, 39(2):159–175, 1987.

Naveen Kumar, R., Jagadale, B., and Bhat, J. A lossless image compression algorithm using wavelets and fractional fourier transform. *SN Applied Sciences*, 1:1–8, 2019.

Nitta, T. On the critical points of the complex-valued neural network. In *Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP'02.*, volume 3, pp. 1099–1103. IEEE, 2002.

Nitta, T. Orthogonality of decision boundaries in complex-valued neural networks. *Neural computation*, 16(1):73–97, 2004.

Oppenheim, A. V. and Lim, J. S. The importance of phase in signals. *Proceedings of the IEEE*, 69(5):529–541, 1981.

Ozaktas, H. M. and Kutay, M. A. The fractional fourier transform. In *2001 European Control Conference (ECC)*, pp. 1477–1483. IEEE, 2001.

Rahman, M. A., Ross, Z. E., and Azizzadenesheli, K. U-no: U-shaped neural operators. *arXiv preprint arXiv:2204.11127*, 2022.

Rawat, S., Rana, K., and Kumar, V. A novel complex-valued convolutional neural network for medical image denoising. *Biomedical Signal Processing and Control*, 69:102859, 2021.

Reichert, D. P. and Serre, T. Neuronal synchrony in complex-valued deep networks. *arXiv preprint arXiv:1312.6115*, 2013.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pp. 234–241. Springer, 2015.

Sewell, G. *Analysis of a finite element method: PDE/PROTRAN*. Springer Science & Business Media, 2012.

Tran, A., Mathews, A., Xie, L., and Ong, C. S. Factorized fourier neural operators. *arXiv preprint arXiv:2111.13802*, 2021.

Trouillon, T. and Nickel, M. Complex and holographic embeddings of knowledge graphs: a comparison. *arXiv preprint arXiv:1707.01475*, 2017.

Tygert, M., Bruna, J., Chintala, S., LeCun, Y., Piantino, S., and Szlam, A. A mathematical motivation for complex-valued convolutional networks. *Neural computation*, 28 (5):815–825, 2016.

Unser, M. Sampling-50 years after shannon. *Proceedings of the IEEE*, 88(4):569–587, 2000.

Vetterli, M., Kovacevic, J., and Goyal, V. Foundations of signal processing, cambridge university press, cambridge, 2014.

Wang, R., Kashinath, K., Mustafa, M., Albert, A., and Yu, R. Towards physics-informed deep learning for turbulent flow prediction. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1457–1466, 2020.

Wen, G., Li, Z., Azizzadenesheli, K., Anandkumar, A., and Benson, S. M. U-fno—an enhanced fourier neural operator-based deep-learning model for multiphase flow. *Advances in Water Resources*, 163:104180, 2022.

Wisdom, S., Powers, T., Hershey, J., Le Roux, J., and Atlas, L. Full-capacity unitary recurrent neural networks. *Advances in neural information processing systems*, 29, 2016.

Worrall, D. E., Garbin, S. J., Turmukhambetov, D., and Brostow, G. J. Harmonic networks: Deep translation and rotation equivariance. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5028–5037, 2017.

Wu, H., Hu, T., Luo, H., Wang, J., and Long, M. Solving high-dimensional pdes with latent spectral models. *arXiv preprint arXiv:2301.12664*, 2023.

Xiong, W., Huang, X., Zhang, Z., Deng, R., Sun, P., and Tian, Y. Koopman neural operator as a mesh-free solver of non-linear partial differential equations. *arXiv preprint arXiv:2301.10022*, 2023.

Yang, M., Ma, M. Q., Li, D., Tsai, Y.-H. H., and Salakhutdinov, R. Complex transformer: A framework for modeling complex-valued sequence. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4232–4236. IEEE, 2020.

Yu, H., Huang, J., Li, L., Zhao, F., et al. Deep fractional fourier transform. *Advances in Neural Information Processing Systems*, 36, 2024.

Zayed, A. I. A convolution and product theorem for the fractional fourier transform. *IEEE Signal processing letters*, 5(4):101–103, 1998.

Zheng, H., Nie, W., Vahdat, A., Azizzadenesheli, K., and Anandkumar, A. Fast sampling of diffusion models via operator learning. In *International Conference on Machine Learning*, pp. 42390–42402. PMLR, 2023.

A. Fractional Fourier Transform

Definition A.1 (Fractional Fourier Transform). The fractional Fourier transform with angle α of a signal $f(x)$ is defined as

$$\mathcal{F}^\alpha\{f\}(x) = \int_{-\infty}^{\infty} f(y) K_\alpha(x, y) \, dt, \tag{10}$$

where

$$K_\alpha(x, y) = \begin{cases} \frac{c(\alpha)}{\sqrt{2\pi}} \exp\{ja(\alpha)[(x^2 + y^2) - 2b(\alpha)xy]\} & \text{if } \alpha \neq 0, \frac{\pi}{2}, \pi, \\ \delta(x - y) & \text{if } \alpha = 0, \\ \delta(x + y) & \text{if } \alpha = \pi, \\ \frac{1}{\sqrt{2\pi}} e^{-jxy} & \text{if } \alpha = \frac{\pi}{2}. \end{cases}$$

where,

$a(\alpha) = \frac{\cot \alpha}{2}$, $b(\alpha) = \sec \alpha$, and $c(\alpha) = \sqrt{1 - j \cot \alpha}$.

Some Properties of FrFT

1. The Fourier transform is a special case of the FrFT when $\alpha = \frac{\pi}{2}$.

2. The FrFT is a linear and bounded operator, i.e.,

$$\mathcal{F}^\alpha\{\sum_k c_k x_k(t)\} = \sum_k c_k \mathcal{F}^\alpha\{x_k(t)\} \tag{11}$$

.

3. FrFT satisfies the Parseval theorem:

$$\int f(t) g^*(t) dt = \int \mathcal{F}^\alpha\{f\}(u) \mathcal{G}^{\alpha*}\{g\}(u) du \tag{12}$$

4. İnverse FrFT of signal is obtained by applying $\mathcal{F}^{-\alpha}$ to the transformed signal.

5. Relation with Wigner Distribution: The Wigner distribution of given signal $f(x)$ is defined as follows (where * denotes conjugation):

$$W_f(x, \nu) = \int_{-\infty}^{\infty} f\left(x + \frac{x'}{2}\right) f^*\left(x - \frac{x'}{2}\right) e^{-i2\pi\nu x'} \, dx' \tag{13}$$

. $W_f(x, \nu)$ denotes the spectrum distribution over time and frequency domain and connects FrFT with spatial-frequency rotation as in Figure 3.

Definition A.2. Let us consider square integrable functions f and g,

Let, $h(x) = (f * g)(x)$ (where * denotes the convolution), i.e.,

$$h(x) = (f * g)(x) = \int_{-\infty}^{\infty} f(t) g(x - t) \, dt. \tag{14}$$

Definition A.3. (Zayed, 1998) For any function $f(x)$, let us define the function $\widetilde{f}(x)$ by $\widetilde{f}(x) = f(x) e^{ja(\alpha)x^2}$. For any two functions f and g, we define the convolution operation $*$ by

$$h(x) = (f * g)(x) = c(\alpha) \frac{e^{-ja(\alpha)x^2}}{\sqrt{2\pi}} (\widetilde{f} * \widetilde{g})(x),$$

where $*$ is the convolution operation as defined in Equation 14.

Theorem A.4 (**Convolution Theorem for Fractional Fourier Transform (FrFT)**). *(Zayed, 1998)*

Let us consider square integrable functions f and g, let $h(x) = (f * g)(x)$ (where $*$ denotes the convolution defined in Equation 14) and let F^α, G^α, H^α denote the FrFT of f, g and h respectively as defined in Equation 10. Then

$$H^\alpha(u) = F^\alpha(u)G^\alpha(u)e^{-ia(\alpha)u^2}. \tag{15}$$

Proof: By FrFT definition from Equation 10, we know that

$$
\begin{aligned}
H_\alpha(u) &= \frac{c(\alpha)}{\sqrt{2\pi}} \int_{-\infty}^{\infty} h(t)e^{j[a(t^2+u^2)-2abut]}dt \\
&= \frac{(c(\alpha))^2}{2\pi} \int_{-\infty}^{\infty} e^{j[a(t^2+u^2)-2abut]}e^{-jat^2}dt \int_{-\infty}^{\infty} f(x)e^{jax^2}g(t-x)e^{ja(t-x)^2}dx \\
&= \frac{(c(\alpha))^2}{2\pi} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x)g(t-x)\exp\left\{j[a(t^2+u^2)-2abut+2ax^2-2atx]\right\}dxdt.
\end{aligned}
$$

By substituting, $t - x = v$ in above equation, we obtain

$$
\begin{aligned}
H^\alpha(u) &= \frac{(c(\alpha))^2}{2\pi} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x)g(v)\exp[ja(x^2+u^2+v^2)-2jabu(x+v)]dxdv \\
&= \frac{(c(\alpha))^2}{2\pi}e^{-jau^2} \int_{-\infty}^{\infty} f(x)e^{j[a(x^2+u^2)-2jabux]}dx \int_{-\infty}^{\infty} g(v)e^{j[a(v^2+u^2)-2jabuv]}dv \\
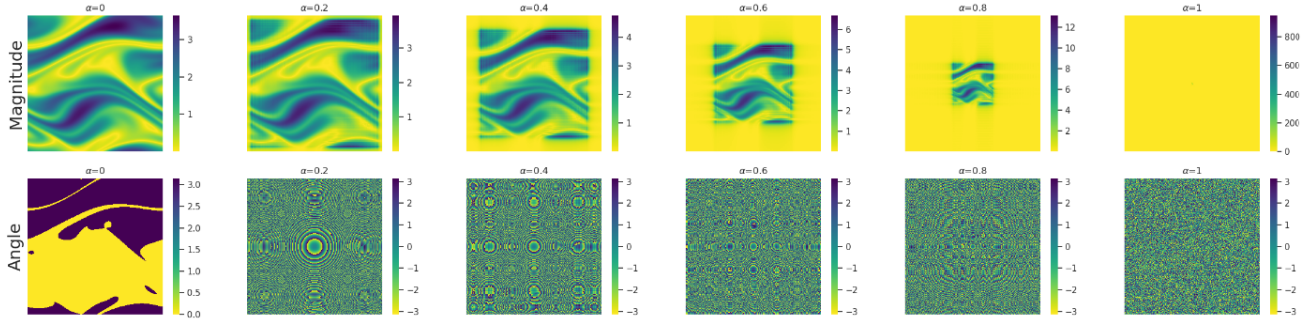&= F^\alpha(u)G^\alpha(u)e^{-jau^2}.
\end{aligned}
$$

Figure 7. The figure illustrates the FrFT of the Navier-Stokes Partial Differential Equation (PDE) across varying α values. The (**Top**) displays heatmaps representing the magnitude of the transformed data, while the (**Bottom**) showcases corresponding heatmaps depicting the phase angle. Each column corresponds to a distinct α value, highlighting the impact of fractional Fourier transform parameters on the learned representation of the Navier-Stokes PDE. It is observed that in Fourier Transform, most spectrum is concentrated at low frequency for $\alpha = 1$.

Figure 3 is an elucidative representation of the intrinsic optimality associated with the Fractional Fourier Transform (FrFT) in signal filtering for non-stationary image signals. It notably underscores FrFT's distinctive capability to effectively separate various signal effects along the fractional order axis. Furthermore, the figure provides insightful scrutiny into the intricate relationship between FrFT and the rotation of the spatial-frequency plane. This visualization contributes to a nuanced understanding of FrFT's prowess in handling non-stationary image signals. It offers valuable insights into its utility and potential applications in the broader signal-processing domain. Figure 7 shows that as the parameter α increases, the spectral distribution becomes increasingly concentrated around the origin. This concentration results in a loss of spatial information while emphasizing frequency information. Conversely, lower values of α reveal the preservation of information in the phase

domain, which, unfortunately, becomes increasingly noisy when transformed into the frequency domain. This behavior highlights the delicate balance between spatial and frequency information, emphasizing the parameter α's critical role in shaping the signal's characteristics.

B. Complex valued backpropagation

Complex-valued backpropagation was implemented using the Wirtinger calculus (Amin et al., 2011), which generalizes the notion of complex derivatives and trains complex-valued neural networks (generalized complex chain rule for real-valued loss function) (Barrachina et al., 2023; Chiheb et al., 2017). If L is a real-valued loss function and z is a complex variable such that $z = x + iy$ where $x, y \in \mathbb{R}$:

$$\nabla_z L = \frac{\partial L}{\partial z} = \frac{\partial L}{\partial x} + i\frac{\partial L}{\partial y} = \frac{\partial L}{\partial (\text{Re}(z))} + i\frac{\partial L}{\partial (\text{Im}(z))} = (\nabla_{Re(z)} L + i(\nabla_{Im(z)} L) \tag{16}$$

We have used Pytorch to build blocks for CoNO based on the following GitHub repositories:

1. `https://github.com/ChihebTrabelsi/deep_complex_networks`

2. `https://complex-valued-neural-networks.readthedocs.io`

C. Details for Benchmark

C.1. Details for benchmarks tasks

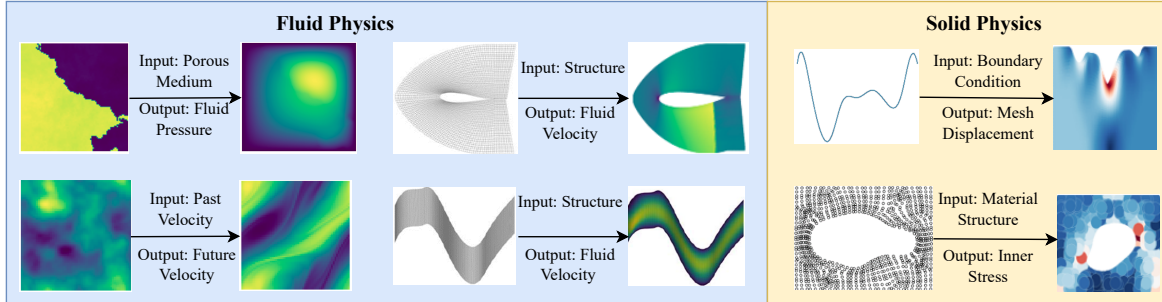The following sections comprehensively elucidate details for benchmark tasks.

Figure 8. The diagram presents an overview of the Operator learning task applied to distinct Partial Differential Equations (PDEs) classified as Fluid Physics and Solid Physics. (**Top**) showcases three specific PDEs: Darcy Flow, Airfoil, and Plasticity. (**Bottom**), an additional set of three PDEs is featured: Navier Stokes, Pipe, and Elasticity.

Table 4. Detailed benchmark tasks provide a thorough and systematic exploration of each task's specific characteristics used in experimentation.

Dataset	Geometry	Task	Input	Output
ELASTICITY-P	Point Cloud	Estimate Stress	Material Structure	Inner Stress
ELASTICITY-G	Regular Grid	Estimate Stress	Material Structure	Inner Stress
PLASTICITY	Structured Mesh	Model Deformation	Boundary Condition	Mesh Displacement
NAVIER-STOKES	Regular Grid	Predict Future	Past Velocity	Future Velocity
AIRFOIL	Structured Mesh	Estimate Velocity	Structure	Fluid Velocity
PIPE	Structured Mesh	Estimate Velocity	Structure	Fluid Velocity
DARCY	Regular Grid	Estimate Pressure	Porous Medium	Fluid Pressure

C.2. Description of Datasets

Table 4 and Figure 8 comprehensively present the benchmark details. The categorization of generation details is presented according to the governing partial differential equations (PDEs) as follows:

Elasticity-P and Elasticity-G Dataset (Li et al., 2022): The dataset is designed to evaluate internal stress within an incompressible material characterized by an arbitrary void at its center, subject to external tension. The material's structural configuration constitutes the input, and the resulting internal stress is the output. Notably, two distinct approaches are employed in modeling the material's geometry: Elasticity-P utilizes a point cloud comprising 972 points. Elasticity-G represents the data on a structured grid with dimensions 41×41, obtained through interpolation from the Elasticity-P dataset.

Plasticity Dataset (Li et al., 2022): This dataset addresses the plastic forging scenario, wherein a die with an arbitrary shape impacts a plastic material from above. The input to the benchmark is characterized by the die's shape, encoded in a structured mesh. The benchmark aims to predict the deformation of each mesh point over the subsequent 20 timesteps. The structured mesh employed has a resolution of 101×31.

Navier Stokes Dataset (Li et al., 2020): 2D Navier-Stokes equation mathematically describes the flow of a viscous, incompressible fluid in vorticity form on the unit torus as follows:

$$\partial_t w(x,t) + u(x,t) \cdot \nabla w(x,t) = \nu \Delta w(x,t) + f(x), \quad x \in (0,1)^2, \, t \in (0,T] \tag{17}$$

$$\nabla \cdot u(x,t) = 0, \quad x \in (0,1)^2, \, t \in [0,T] \tag{18}$$

$$w(x,0) = w_0(x), \quad x \in (0,1)^2 \tag{19}$$

where, u represents the velocity field, $w = \nabla \times u$ is the vorticity, w_0 is the initial vorticity, ν is the viscosity coefficient, and f is the forcing function. In this dataset, the viscosity (ν) is fixed at 10^{-5}, and the 2D field has a resolution of 64×64. Each sample within the dataset comprises 20 consecutive frames. The objective is to predict the subsequent ten frames based on the preceding ten.

Pipe Dataset (Li et al., 2022): This dataset focuses on the incompressible flow through a pipe. The governing equations are Equation 21:

$$\nabla \cdot \mathbf{U} = 0, \tag{20}$$

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{U} \cdot \nabla \mathbf{U} = \mathbf{f}^{-1} \frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{U}. \tag{21}$$

The dataset is constructed on a geometrically structured mesh with a 129×129 resolution. We employ the mesh structure as input data for experimental purposes, with the output being the horizontal fluid velocity within the pipe.

Airfoil Dataset (Li et al., 2022): The dataset pertains to transonic flow over an airfoil. Due to the negligible viscosity of air, the viscous term $\nu \nabla^2 U$ is omitted from the Navier-Stokes equation. Consequently, the governing equations for this scenario are expressed as follows:

$$\frac{\partial \rho f}{\partial t} + \nabla \cdot (\rho f U) = 0 \tag{22}$$

$$\frac{\partial (\rho f U)}{\partial t} + \nabla \cdot (\rho f U U + pI) = 0 \tag{23}$$

$$\frac{\partial E}{\partial t} + \nabla \cdot ((E + p)U) = 0, \tag{24}$$

where ρf represents fluid density, and E denotes total energy. The data is on a structured mesh with dimensions 200×50, and the mesh point coordinates are utilized as inputs. The corresponding output is the Mach number at each mesh point.

Darcy Flow Dataset (Li et al., 2020): It represents the flow through porous media. 2D Darcy flow over a unit square is given by

$$\nabla \cdot (a(x)\nabla u(x)) = f(x), \quad x \in (0,1)^2, \tag{25}$$

$$u(x) = 0, \quad x \in \partial(0,1)^2. \tag{26}$$

where $a(x)$ is the viscosity, $f(x)$ is the forcing term, and $u(x)$ is the solution. This dataset employs a constant value of forcing term $F(x) = \beta$. Further, Equation 25 is modified in the form of a temporal evolution as

$$\partial_t u(x,t) - \nabla \cdot (a(x)\nabla u(x,t)) = f(x), \quad x \in (0,1)^2, \tag{27}$$

In this dataset, the input is represented by the parameter a, and the corresponding output is the solution u. The dataset comprises samples organized on a regular grid with a resolution of 85×85.

D. Prediction Visualization

As illustrated in Figure 9, the performance exhibited by CONO in predictive tasks surpasses that of other benchmark datasets, notably outperforming the state-of-the-art operator LSM. This superiority is particularly evident in time-dependent and independent partial differential equations (PDEs) scenarios. CONO showcases enhanced predictive accuracy and significantly reduces artifacts. These compelling findings underscore the efficacy of CONO as a robust solution for a diverse range of PDE applications, marking a significant advancement in scientific machine learning.
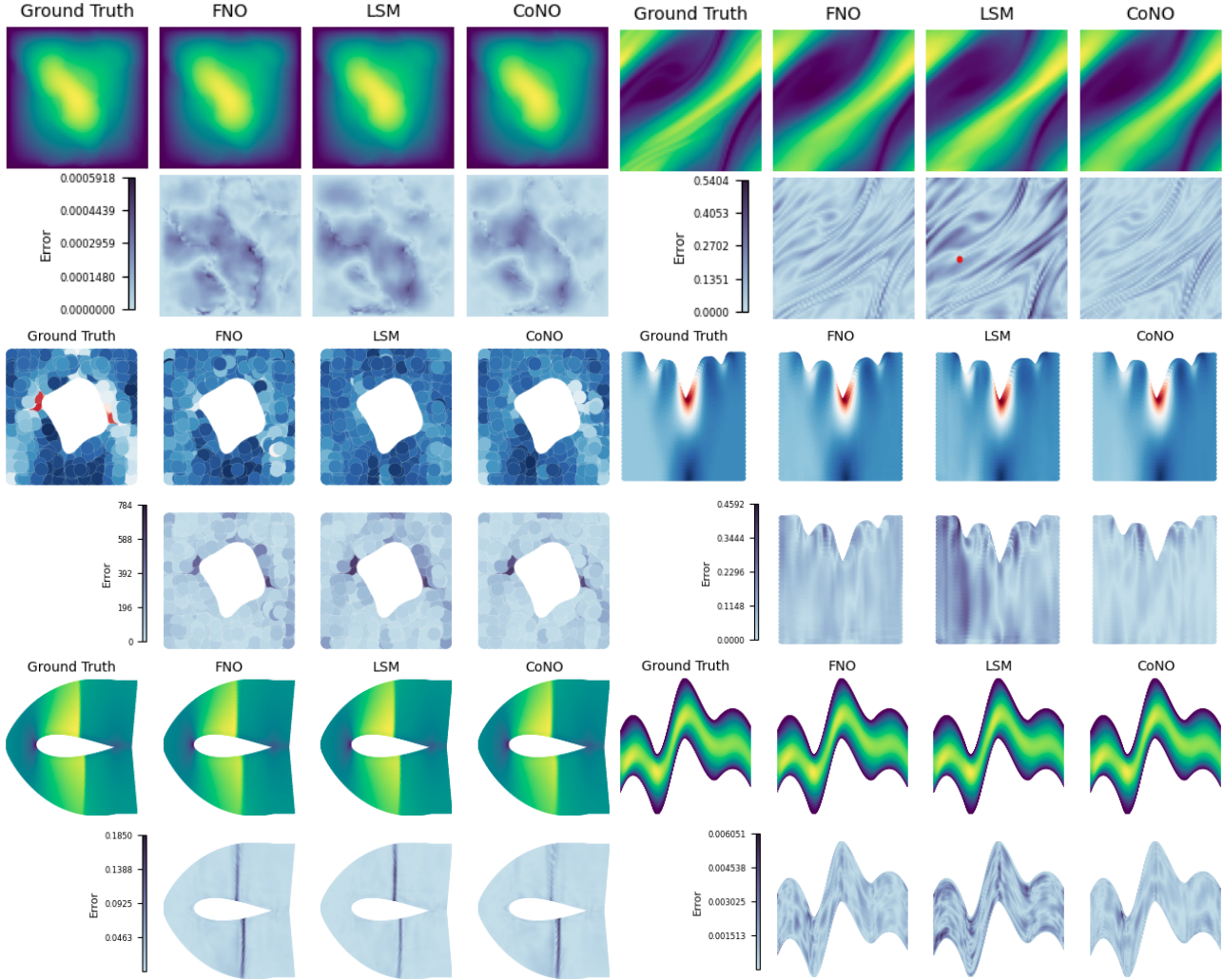
Figure 9. **(Top)** Showcase of Darcy Flow **(Left)** and Navier Stokes **(Right)**. **(Middle)** Showcase of Airfoil **(Right)** and Pipe **(Left)**. **(Bottom)** Showcase of Plasticity **(Left)** and Elasticity **(Right)**. For Comparison of predicted output, we have plotted the heatmap of the absolute value of the difference between Ground Truth and Prediction.

E. Implementation Details

The following section provides a comprehensive overview of the training and testing samples, including details about the shapes of input and output tensors.

16

E.1. Training Details

Table 5 presents a comprehensive overview of the experimental setup, including details regarding the training and testing split and the shapes of the input and output tensors. This information is crucial for understanding the specific configurations employed in our experimentation process. In training CoNO, we need to initialize the fractional orders, following which it is learned in the same way as other matrices which are part of our network using optimizers such as Adam. Notably, fractional orders can vary across axes, and there is no requirement for uniformity initialization of fractional orders across different axes.

Table 5. Training details for benchmark datasets. The input-output resolutions are presented in the shape of (temporal, spatial, variate). The symbol "/" indicates dimensions excluded.

Dataset	Training Samples	Testing Samples	Input Tensor	Output Tensor
ELASTICITY-P	1000	200	$(/, 972, 2)$	$(/, 972, 1)$
ELASTICITY-G	1000	200	$(/, 41 \times 41, 1)$	$(/, 41 \times 41, 1)$
PLASTICITY	900	80	$(/, 101 \times 31, 2)$	$(20, 101 \times 31, 4)$
NAVIER-STOKES	1000	200	$(10, 64 \times 64, 1)$	$(10, 64 \times 64, 1)$
AIRFOIL	1000	100	$(/, 200 \times 50, 2)$	$(/, 200 \times 50, 1)$
PIPE	1000	200	$(/, 129 \times 129, 2)$	$(/, 129 \times 129, 1)$
DARCY	1000	200	$(/, 85 \times 85, 1)$	$(/, 85 \times 85, 1)$

E.2. Complex Neural Networks (CVNNs)

In light of the inherent real-valued nature of all datasets, we devised a neural network tailored for real-valued operations within the complex domain. The network adeptly assimilates the intricacies of complex data components through the utilization of Complex-Valued Neural Networks (CVNNs). The transformative process of extracting complex representations from the initial real-valued data is seamlessly integrated via a residual block, as illustrated in Figure 10. This innovative architectural framework ensures that the neural network efficiently navigates and processes the inherent complexities of the underlying real-valued datasets within the complex domain.
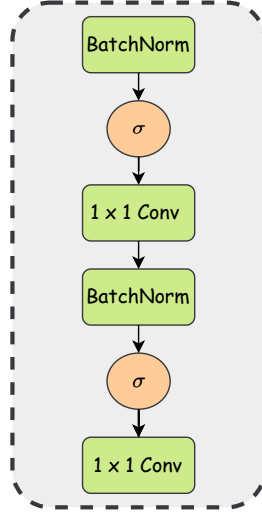

Figure 10. Complex Residual Network used in CoNO for learning the complex part for CVNNs.

Alias-free activation is a simple yet effective approach to mitigate aliasing artifacts introduced during non-linear operations in neural networks as described in Algorithm 1. By incorporating a strategic combination of upsampling, non-linear activation functions, and downsampling with a sinc-based low-pass filter, Alias-Free Activation seeks to preserve the fidelity of signal information, particularly in the context of neural operator learning. This method enhances the accuracy of neural network predictions by minimizing distortions caused by aliasing, which is especially critical in applications dealing with complex signal-processing tasks.

Algorithm 1 Alias-Free Activation

Input: Signal $x \in \mathbb{R}^{n \times m}$
Output: Activated signal $y \in \mathbb{R}^{n \times m}$
Step 1: Upsampling
    Compute the Discrete Fourier Transform (DFT): X of the input signal x.
    Upsample the frequency components by a factor of 2: $X_{\text{up}}[i,j] = X[i,j]$ for $|i| \leq n/2, |j| \leq m/2$ and 0 otherwise.
Step 2: Non-Linear Operation
    Apply the desired non-linear activation function σ element-wise to the upsampled signal: $Z = \sigma(X_{\text{up}})$.
Step 3: Downsampling
    Convolve the activated signal Z with a sinc-based low-pass filter H: $Y_{\text{conv}} = Z * H$.
    Downsample the resulting signal by a factor of 2: $y = Y_{\text{conv}} \downarrow 2$.

E.3. Difference between LSM and CoNO

LSM model uses UNET as the backbone of the neural operator with a latent spectral attention block (transformer block) and a hierarchical projection network as a skip connection across resolution to learn the underlying latent evolution of the PDEs in linear complexity. CoNO uses a series of nonlinear operators to model the underlying nonlinear operator by carrying out a nonlocal integral kernel, which is inspired by the green's function using a fractional integral kernel that extends FNO, which is analogous to the standard neural networks for learning non-linear operators. CoNO has an average gain (averaged over all the datasets) of 10.9% over the next best model and is always among the top 2. It is the SOTA model for all the datasets except two, particularly Pipe and Elasticity-G, where LSM outperforms CoNO. Indeed, it is an open question to be addressed in future work as to why LSM outperforms CoNO on only two specific datasets. This could be potentially attributed to capturing latent variables across different resolutions, as demonstrated by the LSM model. The LSM introduces attention mechanisms to skip connections in UNET, offering a compelling framework for addressing this challenge. Our approach, leveraging nonlinear operations like those in CoNO, achieves comparable performance while yielding a more data-efficient and noise-resilient model compared to LSM.

F. Experiments

The following section provides an exhaustive examination of the results obtained from multiple tasks associated with the operator. These tasks encompass a broad spectrum, including different resolutions, resilience to noise, performance in out-of-generalization tasks, data efficiency, and training stability. The comprehensive analysis presented in this section aims to offer a detailed insight into the performance and capabilities of the operator across a range of critical aspects, contributing to a nuanced understanding of its practical utility and effectiveness in diverse scenarios. All baseline models are instantiated and implemented utilizing the official source code. Note that to ensure a fair comparison across various operators and benchmark datasets, we meticulously executed the GNOT architecture on the Navier-Stokes problem using the Adam optimizer, which stands as a benchmark in the field. While the original implementation employed the AdamW optimizer, Adam provided a rigorous evaluation and facilitated a transparent comparison of different neural operators.

F.1. Performance Under Different Resolution

To assess the efficacy of our operator at diverse resolutions, we conducted experiments on Darcy and Navier-Stokes problems across varying spatial resolutions. Notably, CONO consistently outperformed other operators across these resolutions, demonstrating its superior effectiveness. The comparative results, as presented in Table 6 and Table 7, unequivocally highlight the robust performance and versatility of CONO in addressing challenges associated with different spatial resolutions in the context of both Darcy and Navier-Stokes scenarios.

F.2. Robustness to Noise

To evaluate the robustness of our operator in the presence of noisy training input, we systematically conducted experiments across varying input noise levels. Our objective was to comprehensively understand the impact of noise on the performance of the proposed operator. Remarkably, our findings revealed that CONO, even when subjected to 0.1% data noise, consistently outperformed LSM—the best-performing operator trained without exposure to noisy data. This result is a compelling confirmation of the enhanced robustness exhibited by our proposed operator under challenging conditions involving noisy

Table 6. Neural Operators performance comparison on Darcy Flow under Different Resolutions.

Resolution	UNET	FNO	MWT	UNO	FFNO	HTNET	LSM	CoNO
32x32	0.0059	0.0128	0.0083	0.0148	0.0103	0.0058	0.0049	**0.0048**
64x64	0.0052	0.0067	0.0078	0.0079	0.0064	0.0046	0.0042	**0.0036**
128x128	0.0054	0.0057	0.0064	0.0064	0.0050	0.0040	0.0038	**0.0034**
256x256	0.0251	0.0058	0.0057	0.0064	0.0051	0.0044	0.0043	**0.0039**
512x512	0.0496	0.0057	0.0066	0.0057	0.0042	0.0063	0.0039	**0.0035**
1024x1024	0.0754	0.0062	0.0077	0.0058	0.0069	0.0163	0.0050	**0.0044**

Table 7. Neural Operators performance comparison on the Navier-Stokes Benchmark Under Different Resolutions. "/" indicates poor $l2$ error performance.

Resolution	UNET	FNO	MWT	UNO	FFNO	HTNET	LSM	CoNO
64×64	0.1982	0.1556	0.1541	0.1713	0.2322	0.1847	0.1535	**0.1287**
128×128	/	0.1028	0.1099	0.1068	0.1506	0.1088	0.0961	**0.0817**

training inputs, as highlighted in Table 8.

We observe relative error increases of 22%, 47%, and 28% for CoNO, LSM, and FNO, respectively. Here, CoNO outperforms all the models. Furthermore, under 0.5% data noise conditions, our findings indicate relative error increases of 62%, 186%, and 45% for CoNO, LSM, and FNO, respectively. Here, it performs significantly better than LSM but poorer than FNO. However, it should be noted that the absolute performance of CoNO with respect to FNO is significantly better. Thus, it can be argued that CoNO is robust against noise at least as much as FNO, if not better and clearly better LSM, which is the SOTA model.

Table 8. Neural Operators performance comparison on Darcy Flow under different noise Levels in the training dataset. "/" indicates poor ℓ_2 error performance.

Noise %	UNET	FNO	MWT	UNO	FFNO	HTNET	LSM	CoNO
0 %	0.0080	0.0108	0.0082	0.0113	0.0083	0.0079	0.0065	**0.0050**
0.001 %	0.0094	0.0114	0.0089	0.0115	0.0089	0.0087	0.0085	**0.0056**
0.01 %	0.0105	0.0137	0.0097	0.0121	0.0095	0.0097	0.0087	**0.0058**
0.1 %	0.0113	0.0139	0.0125	0.0126	0.0106	0.0124	0.0096	**0.0061**
0.5 %	/	0.0157	0.0135	0.0138	0.0125	0.0148	0.0186	**0.0081**

F.3. Out of Distribution Generalization

In this study, we conducted extensive experiments utilizing the Navier-Stokes dataset, training our model with a viscosity coefficient set to 10^{-5}. Subsequently, we rigorously assessed the out-of-distribution generalization capabilities by subjecting the trained model to a viscosity coefficient of 10^{-4}, as depicted in Table 9. Our empirical observations consistently demonstrate that the CoNO model exhibits a notably superior generalization performance, showcasing an impressive increment of 64.3% compared to the FNO model's performance. Furthermore, our findings underscore the critical importance of capturing latent variable information, a task effectively accomplished by the UNET architecture, particularly exemplified by the Latent Spectral Operator. Significantly, LSM outperforms all other operators, including CoNO, emphasizing its role in enhancing generalization capabilities in fluid dynamics modeling.

Table 9. Neural Operators performance comparison on the Navier-Stokes Benchmark Under Out of distribution performance. Model is trained on NS viscosity coefficient $1e^{-5}$ and tested on NS viscosity coefficient $1e^{-4}$. "/" indicates poor ℓ_2 error performance.

NS Viscosity Coefficient	UNET	FNO	MWT	UNO	FFNO	HTNET	LSM	CoNO
$1e^{-5}$	0.1982	0.1556	0.1541	0.1713	0.2322	0.1847	0.1535	**0.1287**
$1e^{-4}$	/	0.6621	0.5864	0.5436	0.5606	0.4888	**0.1887**	0.2321

F.4. Long Term Prediction

To assess the long-term performance of the model on time-dependent PDEs, we employ the Rollout Error ($RE(t)$) (Bihani et al., 2023), which is bounded between 0 and 1.

$$RE(t) = \frac{1}{N} \sum_{i=1}^{N} \frac{\|G_\theta(a_i(t)) - G^\dagger(a_i(t))\|_2}{\|G_\theta(a_i)(t)\|_2 + \|G^\dagger(a_i(t))\|_2} \tag{28}$$

To assess the long-term behavior of the operator, we conducted training on the Navier-Stokes equation, subsequently evaluating its performance in terms of Rolling Error (Equation 28) to observe potential divergence in long-term predictions. The proposed operator (CoNO) outperforms alternative operators LSM and FNO consistently throughput prediction trajectory, demonstrating consistent and superior performance in capturing the system's spectrum across the extended time horizon, as depicted in Figure 11. In further substantiating our evidence regarding the enhanced stability of CoNO in long-horizon predictions, we conducted an additional experiment utilizing the Navier Stokes model with a viscosity coefficient of 10^{-4}. Here, we performed forecasts for the subsequent ten steps based solely on the preceding ten observations and extrapolated the results over ten more steps. The outcomes of this experiment are presented in the tables below. Our findings indicate that although not explicitly trained for predicting the subsequent 20 timestamps, CoNO consistently outperforms LSM and FNO in extrapolating beyond the prediction horizon. This observation underscores the heightened stability and robustness of CoNO in long-horizon prediction tasks as shown in Table 10.

Table 10. A comparative analysis of the performance of Neural Operators on the Navier-Stokes equations, with a viscosity coefficient of 10^{-4}, involves training models to predict the subsequent 10 timestamps based on the preceding 10 timestamps. Subsequently, the extrapolated results are utilized to forecast the subsequent 10 timestamps.

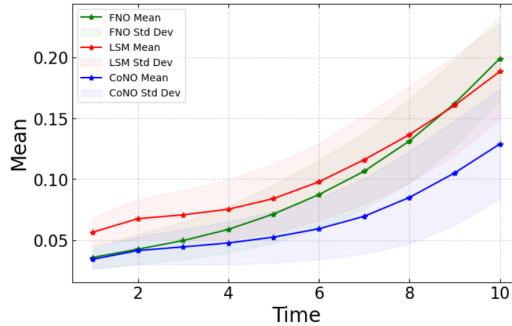| Neural Operator	$T = 5$	$T = 10$	$T = 15$	$T = 20$
FNO	0.028	0.050	0.17	0.34
LSM	0.065	0.13	0.24	0.38
CoNO	**0.020**	**0.045**	**0.14**	**0.31**

Figure 11. Rolling Error versus time plot for Navier Stokes. The lower the $RE(t)$, the better operator learns the underlying solution.

F.5. Effect of Number of Layers

We conducted experiments to analyze the relationship between the number of layers and performance, in comparison to FNO, on the Darcy flow dataset. Our findings, presented in the table below, reveal that performance initially improves

with increased layers in FNO. However, a significant drop in performance occurs due to the vanishing gradient problem. Importantly, our method consistently avoids the vanishing gradient problem even with an increase in the number of layers as shown in Table 11.

Table 11. Neural Operators performance comparison on Darcy Benchmark under different numbers of layers.

Number of Layers	2	4	6	8
FNO	0.0114	0.0108	0.0087	0.0098
CoNO	**0.0066**	**0.0052**	**0.0053**	**0.0052**

F.6. Data Efficiency

As shown in Table 12, the performance of CoNO is superior, showcasing its competitive capabilities comparable to the second-best operator LSM when trained on 60% of the available data. Remarkably, across diverse training dataset ratios, CoNO consistently surpasses all other operators, emphasizing its remarkable data efficiency compared to state-of-the-art (SOTA) operators. This observation underscores the efficacy and robustness of CoNO across varying training scenarios, positioning it as a noteworthy solution in the realm of operator-based learning. With a data ratio of 0.6, we observed relative error increases of 34%, 44%, and 35% for CoNO, LSM, and FNO, respectively. Thus, it is evident that our approach consistently outperforms the second-best operator, LSM, across various ratio settings. Further, the absolute performance of CoNO is significantly better than FNO.

Table 12. Neural Operators performance comparison on Darcy Benchmark under different training dataset ratios. "/" indicates poor ℓ_2 error performance.

Ratio	UNET	FNO	MWT	UNO	FFNO	HTNET	LSM	CoNO
0.2	/	0.2678	0.2854	0.2734	0.2573	0.2564	0.2465	**0.2234**
0.4	/	0.0176	0.0165	0.0183	0.0153	0.0145	0.0138	**0.0105**
0.6	0.1234	0.0146	0.0113	0.0153	0.0142	0.0105	0.0094	**0.0067**
0.8	0.0107	0.0122	0.0096	0.0134	0.0095	0.0094	0.0082	**0.0056**
1.0	0.0080	0.0108	0.0082	0.0113	0.0077	0.0079	0.0065	**0.0050**

F.7. Inference Time Comparison

The section examines inference time complexity across diverse neural operators applied to the Darcy Flow benchmark dataset within the Fluid Physics field. It aims to elucidate these operators' computational efficiency and efficacy in modeling fluid flow phenomena, as shown in Table 13.

Table 13. The evaluation of inference time complexity across various neural operators on the Darcy Flow benchmark dataset in the domain of Fluid Physics.

Neural Operator	UNET	UFNO	FNO	WMT	LSM	CoNO
Inference (s)	0.035	0.042	0.135	0.045	0.020	0.055

F.8. Bias vs FrFT

Intrinsically, the omission of bias and FrFT components leads to a notable decrease in performance. Bias operates in the spatial/temporal domain and captures the high-frequency signals as it is obtained from a point-wise convolution. In contrast, FrFT operates within the frequency and spatial/temporal domain spectrum, encapsulating orthogonal information. Figure 7 in Appendix A shows the scaling effect inherent to FrFT. Consequently, eliminating either component leads to a substantial decline in performance. However, it is noteworthy that removing the FrFT component results in a more pronounced decrease in performance compared to eliminating the bias component, as illustrated by the ablations in our manuscript. The function

of bias within neural operators is to discern and incorporate high-frequency components, whereas FrFT plays a pivotal role in integrating temporal/spatial and frequency components into neural operators. Consequently, removing either the FrFT component results in a notable decline in performance, as evidenced by our ablation studies.

G. Limitations and Future Work

To deepen our comprehension of CoNO, a thorough exploration of its underlying mathematical and algorithmic principles becomes imperative. Our objective is to unveil the intricate learning mechanisms inherent in the latent space and establish a robust theoretical foundation for complex operators. In the pursuit of advancing the field, our research introduces several noteworthy challenges that demand careful investigation. These challenges encompass refining the initialization procedures tailored for fractional orders, designing streamlined architectures tailored for complex neural operators, exploring the development of equivariant complex operators, and elucidating the pivotal role played by the Fractional Fourier Transform (FrFT) in gaining insights into the continuous dynamics of complex systems. Additionally, our work prompts inquiries into establishing foundational models for Partial Differential Equations (PDEs). These avenues represent compelling directions for future studies, offering opportunities further to unravel the intricacies of CoNO and contribute to the broader landscape of SciML.