Journal of Computational Physics 513 (2024) 113168

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Out-of-distributional risk bounds for neural operators with applications to the Helmholtz equation

Jose Antonio Lara Benitez [a,*], Takashi Furuya [b,*], Florian Faucher [c], Anastasis Kratsios [d], Xavier Tricoche [e], Maarten V. de Hoop [a]

[a] Rice University, United States of America
[b] Shimane University, Japan
[c] Team Makutu, Inria Bordeaux, University of Pau and Pays de l'Adour, TotalEnergies, France
[d] McMaster University and the Vector Institute, Canada
[e] Purdue University, United States of America

ARTICLE INFO

Keywords:
Neural operator
Transformer-inspired
Forward operator
Generalization error bounds
Out-of-distributional risk bounds

ABSTRACT

Despite their remarkable success in approximating a wide range of operators defined by PDEs, existing *neural operators* (NOs) do not necessarily perform well for all physics problems. We focus here on high-frequency waves to highlight possible shortcomings. To resolve these, we propose a subfamily of NOs enabling an enhanced empirical approximation of the nonlinear operator mapping wave speed to solution, or boundary values for the Helmholtz equation on a bounded domain. The latter operator is commonly referred to as the "*forward*" operator in the study of inverse problems, and we propose a hypernetwork version of the subfamily of NOs as a surrogate model. Our methodology draws inspiration from transformers and techniques such as stochastic depth. Experiments reveal certain surprises in the generalization and the relevance of introducing stochastic depth. Our NOs show superior performance as compared with standard NOs, not only for testing within the training distribution but also for out-of-distribution scenarios. To delve into this observation, we obtain a novel *out-of-distribution* risk bound tailored to Gaussian measures on Banach spaces, relating stochastic depth with the bound. We conclude by offering an in-depth analysis of the Rademacher complexity associated with our modified models and prove an upper bound tied to their stochastic depth that existing NOs do not satisfy.

1. Introduction

Data-driven approximation of operators is gaining momentum due to its potential to approximate operators over expensive numerical solvers at a fraction of the computational cost, particularly in the context of parametric partial differential equations (PDEs). This approach proves particularly advantageous in scenarios where constitutive laws are approximated, or only data are available. Once the model is fully-trained, the solution is, up to an approximation error, obtained by evaluating the neural network with restrictions on the input, i.e., the test data are drawn from the same or a sufficiently similar distribution as the training data. Numerous architectures have been proposed in recent years, such as DeepONets [89,90], PCA-Net [11,51], PINNs [99,61], and neural operators (NOs) [83,66,82]. Among them, Fourier neural operators (FNOs) have gained widespread popularity. Indeed,

* Corresponding authors.
E-mail addresses: antonio.lara@rice.edu (J.A. Lara Benitez), takashi.furuya0101@gmail.com (T. Furuya).

https://doi.org/10.1016/j.jcp.2024.113168
Received 15 July 2023; Received in revised form 5 April 2024; Accepted 1 June 2024

Available online 6 June 2024

they can efficiently compute the costly integral operator, they enjoy "*discretization invariance*",[1] and are universal approximators, under the assumption of regularity and separability of spaces. FNOs or iterations thereof, [17,112,107,18], have rapidly become the network of choice, finding applications in various domains [98,46,116,113,78,45,70].

FNOs have shown promising results in certain 2D PDE problems, e.g., *incompressible Navier-Stokes equation*, [65, Section 3.2], and even some non-linear inverse problems, [115,95]. However, application to realistically complex large-scale problems remains an issue, despite some recent progress [45]. NOs are the natural generalization of multilayer perceptron (MLPs) to functional spaces, and they share their limitations. For example, You et al. [118] have shown that deep FNOs perform poorly on some nonlinear operators for PDEs, despite being theoretically universal [65]. These findings underscore the need for architectures that possess more desirable properties in implementation. Moreover, the increasing interest in enhancing or replacing traditional numerical methods has prompted a focus on understanding the generalization capabilities and training dynamics rather than solely relying on the approximation power of networks, e.g. [91,72,71,73,28].

In this paper we focus on *the out-of-sample, or generalization*, performance of neural operators trained from finitely many noisy inputs. We consider neural operators of the form

$$v_{\ell+1} = \sigma \circ (W_\ell + \boldsymbol{\mathcal{K}}_\ell + b_\ell) \circ v_\ell, \tag{1}$$

or our proposed network

$$v_{\ell+1} = \left(\mathbf{I_d} + \mathbf{X}_\ell\, f_\ell \circ \mathrm{N}\right) \circ \left(\mathbf{I_d} + \mathbf{X}_\ell\, \sigma \circ (\boldsymbol{\mathcal{K}}_\ell + b_\ell) \circ \mathrm{N}\right) \circ v_\ell. \tag{2}$$

Here, $\boldsymbol{\mathcal{K}} v(x) = \int k(x,y) v(y) dy$ represents integral operators, where the kernel function $k_{i,j}$ is uniformly bounded for each point x and y. This boundedness property allows us to establish theorems that hold regardless of the choice of basis expansions for $\boldsymbol{\mathcal{K}}$. $\mathbf{X}_\ell$ are Bernoulli random variables, $\mathbf{X}_\ell \sim \mathrm{Ber}(\mathrm{p}_\ell)$, acting as "switches", controlling the propagation of information within the network, and adding extra randomness in the training. The process of adding $\mathbf{X}_\ell \sim \mathrm{Ber}(\mathrm{p}_\ell)$, such that p_ℓ decreases with depth is known as *stochastic depth* [55]. Finally, f_ℓ is a simple multilayer perceptron (MLP), σ the activation, N a normalizer, and $\mathbf{I_d}$ the identity operator, which we introduce formally in Section 2.

The theory of generalization for neural operators is still in its early stages, with ongoing advancements in the field, as Kim and Kang [62], controlling the estimation error using uniform laws of large numbers. However, such methods have primarily focused on finite-dimensional parameters, as they rely on established theorems within the statistical learning community. Nevertheless, the underlying theory can be extended to encompass a broader range of kernel functions, beyond those approximated by Fourier basis. In our work, we have extended the theory to a wider class of kernel functions and have avoided relying on the constraints of finite dimensionality. This allows for the consideration of alternative bases for expressing the integral operator, such as wavelet basis, spherical harmonics, and others.

Additionally, the theory of generalization in neural networks encompasses their ability to handle perturbations in the underlying distribution, including out-of-distribution scenarios. While empirical and theoretical results for operator learning are relatively scarce and challenging to obtain, there are notable exceptions, such as the work by de Hoop et al. [26, Sec 4.1.2] that focuses on learning linear operators from data. In our work, we make a further contribution to this area by investigating the robustness of the proposed network (2), to changes in the input distribution. Empirically, we observe that the network exhibits robustness to such changes. Theoretically, we leverage properties from the theory of general Gaussian measures on Banach spaces and the duality of the Wasserstein 1 distance to establish an upper bound on the network's robustness to a change of measure. It is important to note that the random variables in (2) play a significant role in controlling the bound, particularly as the depth of the networks increases. These findings shed light on the generalization capabilities of the networks and provide insights into their behavior beyond the training distribution. However, it is worth mentioning that our bounds rely on estimates of the Lipschitz constant, and those are not tight. Strictly speaking, further analysis is needed to fully understand the growth and provide a complete explanation of the observed out-of-distribution behavior.

Our proposed architecture modifications in (2) borrow ideas from transformers, in particular to the encoder part, whose layers can be described as

$$v_{\ell+1} = \left(\mathbf{I_d} + \mathbf{X}_\ell\, f_\ell \circ \mathrm{N}\right) \circ \left(\mathbf{I_d} + \mathbf{X}_\ell \circ \mathrm{Attn}\right) \circ v_\ell, \quad \mathrm{Attn}(v_\ell) = \mathrm{softmax}\left(\mathrm{Const.}\, Q(v_\ell) K(v_\ell)^\top\right) V(v_\ell), \tag{3}$$

for $v_\ell \in \mathbb{R}^{n \times d}$, and $Q(v_\ell) = v_\ell W^Q$, $K(v_\ell) = v_\ell W^K$, $V(v_\ell) = v_\ell W^V$ for $W^Q, W^K, W^V \in \mathbb{R}^{d \times d}$. As we delve into the subject, we will discover how this approach grants us significant control over the complexity class within the family, while effectively bounding the out-of-distribution risk through stochastic depth for Gaussian measures. Additionally, it empowers us to leverage a proven network layout, which has consistently demonstrated promising empirical results across various domains. In recent years, there has been a shift towards the adoption of transformer-based architectures [109,29,31,85,2] throughout machine learning. These architectures, which include widely publicized models such as BERT and ChatGPT [29,84], have shown remarkable success in various tasks, outperforming previous state-of-the-art models. Kovachki et al. [66, Section 3.3] has identified a connection between transformers and neural operators, where self-attention can be viewed as a Monte Carlo approximation of a *nonlinear integral operator*, showing that the underlying principles of these seemingly different architectures are linked. Cao [19], Kissas et al. [63], Li et al. [81] explored transformers for parametric PDEs. Despite the promising results for small-scale problems, using transformers to approximate

[1] In the sense of zero-shot super-resolution, that is, training in a coarse grid and testing in a finer grid.

operators is hindered by the inherent scalability issue of self-attention.[2] Attention operations have a cost of $\mathcal{O}(n^2)$, making them prohibitively expensive for realistic 3D inputs. Incorporating workarounds like shifted windows in visual transformers, as seen in [85], can be beneficial for certain applications. However, the absence of a solid theoretical foundation in these approaches makes it challenging to analyze the convergence of the architecture, particularly in scientific computing scenarios. In contrast, the *convolutional integral operator* in FNOs can be efficiently estimated by the Fast Fourier Transform (FFT) with a computational cost of $\mathcal{O}(n \log n)$. Furthermore, the adaptive Fourier neural operator (AFNO) [47] presents a promising approach to address the scalability limitations of transformers. Nonetheless, current applications have been primarily limited to vision, and further research is needed to explore these architectures in scientific computing.

Extensive empirical evidence has shown that design choices in transformers can yield significant improvements in the capacity of network families, training stability, generalization performance to in-distribution-data, and sometimes out-of-distribution, [50]. This has resulted in a growing trend in various fields of machine learning to adopt "*transformized*" architectures [105,77,119,93,100,86]. The work of Yu et al. [119], abstracts the self-attention of the transformer leading to a so-called *metaformer* architecture. Here, we take advantage of the abstracted layout of this approach to overcome limitations associated with traditional self-attention in terms of input's dimension. Furthermore, this opens up possibilities for designing transformer-based models that can effectively tackle problems arising in scientific computing on an ad-hoc basis.

Our contributions

(a) We introduce modifications to neural operators to adopt a transformer-like architecture, drawing inspiration from works such as [86,77,119]. The resulting network (Section 2) is referred to as sFNO $+ \varepsilon$I and sNO $+ \varepsilon$I, respectively, for experiments and theory, where the ε indicates that "minor" changes are incorporated, and the s stands for sequential, as we preserve the arrangement: non-local (integral operator layer as "token mixer"), and local (MLP layer as "feature mixing") in transformers (contrasting with traditional NOs).

(b) We construct a benchmark for the time-harmonic wave equation according to [37]. We observe that modifying FNOs towards sFNO $+ \varepsilon$I leads to a smaller test loss in the parametric form of the Helmholtz equation $c \mapsto \mathrm{p}$ (Section 4.4) for data *in-distribution*.

(c) We provide an exhaustive empirical study of the robustness of the trained networks for perturbation in the data distribution. We show that the proposed architecture is able to generalize to *out-of distribution* input, while earlier networks are unable to. Remarkably, the proposed network is able to obtain *reasonable* wave propagation from an *anisotropic* covariance operator, change in the input's range and roughness coefficient, despite being only trained on smooth Gaussian random fields with Whittle–Matérn *isotropic* covariance, and fixed wave speed range (Section 5).

(d) We propose a hypernetwork version of the architecture, as a surrogate model to effectively approximate the forward operator of the Helmholtz equation (Section 6). That is, $(f, c) \to \mathrm{p}^f|_\Sigma$, where $\mathrm{p}^f|_\Sigma$ is the restriction of the wavefield at receivers location for a given source, f.

(e) We give theoretical guarantees supporting the out-of-distribution performance of the sNO $+ \varepsilon$I and sNO $+ \varepsilon$Iv2 (2) models in the case where the inputs are sampled from a centered Gaussian measure μ_X on various Banach spaces (Section 7). We find that the out-of-sample generalization of both neural operator models is described by the metric entropy of the unit Cameron-Martin space associated with μ_X. The analysis extends the transport-theoretic tools for deriving risk-bounds introduced in [53] and merges it with small-ball estimates for Gaussian processes on Banach spaces, e.g. [80].

(f) We offer a novel analysis of the *Rademacher complexity* of NOs and *our proposed architecture* (2) (Section 8). For NO, our analysis is general in the sense that it applies independently of the discretization and of the choice of basis in the integral operator,[3] contrasting with [62]. In addition, our work not only extends the previous results to functional space but also provides a better bound on the Rademacher complexity with order $\mathcal{O}\left(1/n^{\frac{1}{\hat{d}+1}}\right)$ (n is the cardinality of the training dataset, and $\hat{d}$ is the doubling dimension of $D \times D$, where D is the spatial domain), whilst $\mathcal{O}(1)$ in [62]. For the *Rademacher complexity* of (2) our analysis is tied to *stochastic depth*. We show that stochastic depth controls the expected Rademacher complexity, irrespective of the number of layers. For instance, if $\mathbf{X}_\ell \sim \mathrm{Ber}(p_\ell)$, and $p_\ell = \mathcal{O}(\ell^{-(1+\varepsilon)})$, where ℓ denotes the layer's number, and $\varepsilon > 0$, the bound is uniform regardless of $\ell \to \infty$.[4] As a consequence, we show that the upper bound of the sNO $+ \varepsilon$I can always be controlled with depth, while the upper bound of the other neural operators diverges.

Notation We denote $d \in \mathbb{N}$ as the number of components of the domain $D \subset \mathbb{R}^d$ used throughout the paper, by d_a the dimension of the image of the function a, $a(x) \in \mathbb{R}^{d_a}$ or $a(x) \in \mathbb{C}^{d_a}$, the meaning is clear from the context. We usually denote $a \in \mathrm{L}^2(D; \mathbb{R}^{d_a})$ and $u \in \mathrm{L}^2(D; \mathbb{R}^{d_u})$ as the input and output functions related by an operator (e.g., if $\mathcal{G}$ is an operator between these two spaces, $\mathcal{G}(a)(x) = u(x)$). We denote the weight matrix at layer ℓ as $W_\ell \in \mathbb{R}^{d_\ell \times d_{\ell-1}}$, and by $b_\ell \in \mathbb{R}^{d_\ell}$ the bias. For the main network (7) and the intermediate architectures Equations (5) to (6), we incorporate a multilayer-perceptron at the block-layer ℓ, and we denote the corresponding weight matrix at the m-th layer in the MLP as $W_{\ell,m} \in \mathbb{R}^{d^w_{\ell,m+1} \times d^w_{\ell,m}}$, in where ℓ refers to the block-layer ℓ. We use f_ℓ as

[2] In transformer applications, datasets are massive, but individual data samples are relatively small compared to those in PDE-related problems, particularly in 3D cases. In areas like vision, attention is typically applied to image patches instead of at a pixel-wise level to reduce computational cost, e.g. [31]

[3] In particular, Fourier basis corresponding to FNOs.

[4] Similar conclusion is obtained in theoretical analysis of OOD.

Table 1
Notations.

Notation	Description	Reference
d	number of components of spatial domain $D \subset \mathbb{R}^d$	
σ	activation function	
$\boldsymbol{\mathcal{K}}_\ell$	integral operator with kernel k_ℓ	
k_ℓ	$d_{\ell+1} \times d_\ell$-kernel matrix for $\boldsymbol{\mathcal{K}}_\ell$	
$\boldsymbol{R}$, $\boldsymbol{Q}$	lifting and projection operator	Fig. A.17
$\mathbf{X}_\ell$	Bernoulli random variables at layer ℓ	Equation (7)
	Spaces, metrics and norms	
L^2	space of square-integrable functions	
$H^s = W^{s,2}$	Sobolev space of smoothness s, with norm $\|\cdot\|_{H^s}$	Section 3.4
$\mathcal{H}_{\mu_X}$	Cameron-Martin space of a Gaussian measure μ_x	Assumption 7.2
$\|\cdot\|_{\mathrm{Lip}}$	Lipschitz norm of the Sobolev space $W^{1,\infty}$	Equation (21)
$\mathcal{W}_1$	Wasserstein-1 distance	Equation (20)
	Experiments	
$\mathcal{G}$	operator $c \mapsto p$	Section 3.2
$\mathcal{F}^f_\omega$	forward operator $(c, f, \omega) \mapsto \{\mathrm{p}(\boldsymbol{x}_j, \omega, f)\}$	Section 3.3
$\mathcal{R}$	restriction operator $\mathcal{R}(\mathrm{p}) = \mathrm{p}\|_\Sigma$	Section 3.3
λ	correlation range of the Whittle–Matérn field	Equation (15)
ν	smoothness of the Whittle–Matérn field	Equation (14)
	Learning theory	
$\mathrm{S} = \{a_n, u_n\}_{n=1}^N$	training dataset drawn from the probability measure μ	Section 8.1
$\mathfrak{R}_{\mathrm{S}}^n(\mathscr{F})$	Rademacher complexity of the class $\mathscr{F}$ given the dataset S	Equation (34)
μ^N	empirical measure $\mu_X^N = N^{-1} \sum_{n \le N} \delta_{a_n}$	Equation (25)
$\hat{d}$	doubling dimension of $D \times D$	Definition A.16
$\mathscr{N}$	hypothesis class of neural operators	Equation (26)
$\widetilde{\mathscr{N}}$	hypothesis class of sequential neural operators (Equations (5) to (7))	Equation (27)

the MLP at layer ℓ in (2) and all the intermediate architectures. For an integral operator with kernel function k_ℓ at layer ℓ, we write it as $\boldsymbol{\mathcal{K}}_\ell$. By $\|\cdot\|_{\mathrm{op}}$ we denote the operator norm induced by the euclidean norm and by $\|\cdot\|_{\mathrm{F}}$ the Frobenius norm of given matrices. By $\|\cdot\|_2$ we denote the ℓ_2-norm and by $\|\cdot\|_{\mathrm{L}^2(D;\mathbb{R}^h)}$ the corresponding L^2-norm. For $s \in [0, \infty)$, we denote $H^s(D)$ the Sobolev space, which for $s = 0$, $H^0(D) = \mathrm{L}^2(D)$. For the *generalization error bound* statements we denote the cardinality of the training dataset as n. The hypothesis class of neural operators, precisely defined in (26), as $\mathscr{N}$ and the sequential neural operator class (27) as $\widetilde{\mathscr{N}}$ (all the architectures defined in Equations (5) to (7)). For probabilistic statements, we will assume a suitable underlying probability space with probability measure μ. We denote the probability measure in $\mathrm{L}^2(D;\mathbb{R}^{d_a}) \times \mathrm{L}^2(D;\mathbb{R}^{d_u})$ as μ (or sometimes $\mathbf{P}$, it is clear from the context), with marginals μ_a (i.e., the marginal of μ on $\mathrm{L}^2(D;\mathbb{R}^{d_a})$) and μ_u the marginal in $\mathrm{L}^2(D;\mathbb{R}^{d_u})$. By $\hat{d}$ we denote the doubling dimension of $D \times D$. By $\|\cdot\|_{\mathrm{Lip}}$ we denote the Lipschitz norm, see (21), and by sampling norm, $\|\ell\|_{\mathrm{S}} := \left(n^{-1} \sum_{i=1}^n \ell(a_i, u_i)^2\right)^{1/2}$. For the *out-of-distribution* statements, we denote the Cameron-Martin space of a Gaussian measure μ_x as $\mathcal{H}_{\mu_X}$ and by μ^N the empirical measure $\mu_X^N = N^{-1} \sum_{n \le N} \delta_{a_n}$.

A summary of the notation used is presented in Table 1.

2. Proposed networks: "metaforming the neural operator"

In this section, we introduce the architecture known as $s\mathrm{NO} + \varepsilon\mathrm{I}$ with stochastic depth. This architecture is designed to enhance the generalization performance and capabilities of neural operators. Here, we provide a comprehensive description of the layers that constitute $s\mathrm{NO} + \varepsilon\mathrm{I}$, which are briefly outlined in Equation (2). However, to understand the impact of different architectural changes, we gradually modify the NOs until obtaining the $s\mathrm{NO} + \varepsilon\mathrm{I}$ with stochastic depth. Throughout the next sections, we provide both numerical evidence and theoretical reasoning to support our choices.

Neural operator: standard structure We briefly review NOs [66,65]. Let $\boldsymbol{\mathcal{K}}_\ell$ be a linear integral operator (non-local), see Definition A.3, and W_ℓ be the weight matrix (local). The standard layer structure is

$$v_{\ell+1} = \sigma \circ (W_\ell + \boldsymbol{\mathcal{K}}_\ell + b_\ell) \circ v_\ell \tag{4}$$

($\ell = 1, \dots, L$), where σ is an element-wise nonlinear activation function, and b_ℓ is a bias. For $\ell = 1$, we have $v_1 = \boldsymbol{R}(a)$, i.e., the parameter a is lifted by the map $\boldsymbol{R}$, and finally, the output is projected back to the corresponding space by $\boldsymbol{Q}$, forming the solution field, $u = \boldsymbol{Q}(v_{L+1})$. We refer to Appendix A.3 for additional explanations.

Sequential neural operators (sNO) Transformers [119,109] adopt a compositional structure, wherein non-local and local layers are arranged sequentially instead of combining the operations within a single layer. The so-called token mixer (e.g. attention) precedes a

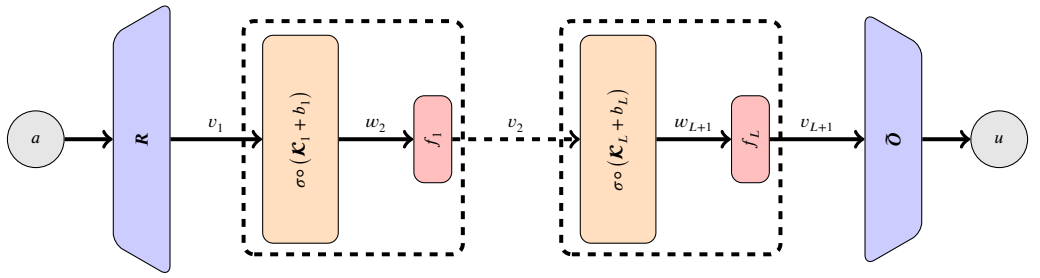


Fig. 1. sNO is called sequential, as the integral operator is followed by a MLP in a sequential manner. For comparison with the NO, see Fig. A.17.

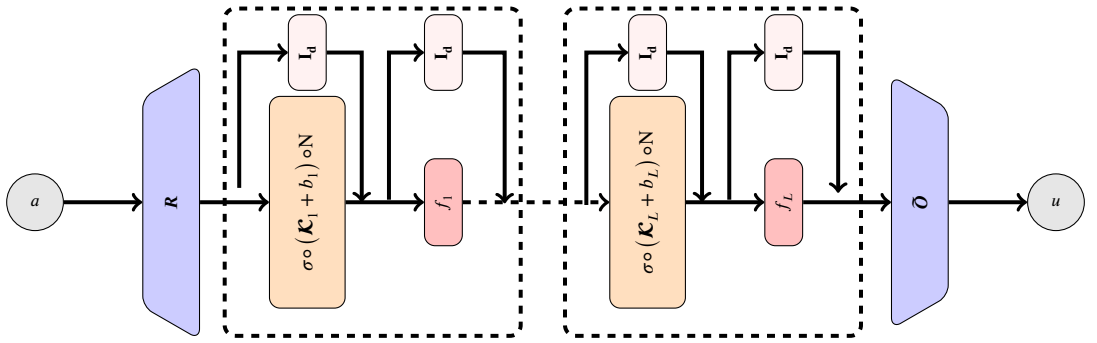


Fig. 2. sNO $+\,\varepsilon$I without stochastic depth. It is a modification based on the sequential structure in where we incorporate layer normalization and skip connection as in transformers. For comparison with the NO, see Fig. A.17.

MLP acting on feature space; see Fig. 1. This structure bears resemblance to the one described by Kovachki et al. [65, Section 2.5.1] for 1-layer NN, and FNOs (the authors of [65] observed that universality is preserved, so it can be expanded to MLP architecture with M-layers, see Appendix A.3.2).
We introduce the sequential neural operator (sNO). Let f_ℓ be a MLP with M-layers (local), [42, Ch. 6]. Then,

$$\begin{cases} w'_\ell &= \sigma\circ(\boldsymbol{\mathcal{K}}_\ell + b_\ell)\circ v_\ell, & \text{(a)} \\ v_{\ell+1} &= f_\ell \circ w'_\ell & \text{(b)} \end{cases} \tag{5}$$

($\ell = 1, \dots, L$). See, Fig. 1. If $\boldsymbol{\mathcal{K}}$ is convolutional, we find a significant improvement over the relative L^2-norm compared to traditional FNOs for similar parameter count, see Section 2 and Fig. 7.

sNO $+\,\varepsilon$I: sNO with the identity map–skip connection We now incorporate the addition of the identity map on the sNO (in the field of machine learning, this is referred to as a *skip connection*). The use of the symbol ε in the name is merely to signify that minor changes have been made to the sNO architecture.

Two variants can be considered: one without, and one with stochastic depth [55], allowing us to access deep versions of sNO $+\,\varepsilon$I. For the sake of brevity, sometimes *we may refer to sNO $+\,\varepsilon$I without stochastic depth as version* 1*, and sNO $+\,\varepsilon$I with stochastic depth as version* 2, in figures, or tables.

sNO $+\,\varepsilon$I without stochastic depth Incorporating skip connections, that is sNO $+\,\mathbf{I_d}$, lead us to Equation (6). The architecture can be seen as an instance of the metaformer [119]; whence, the token mixer is replaced by an integral operator, and the network is extended to functional space.[5] Using a similar notation, we have

$$\begin{cases} w'_\ell &= \big(\mathbf{I_d} + \sigma\circ(\boldsymbol{\mathcal{K}}_\ell + b_\ell)\circ\mathrm{N}\big)\circ v_\ell, & \text{(a)} \\ v_{\ell+1} &= \big(\mathbf{I_d} + f_\ell\circ\mathrm{N}\big)\circ w'_\ell & \text{(b)} \end{cases} \tag{6}$$

($\ell = 1, \dots, L$), where $\mathbf{I_d}$ is the identity operator, and N is the layer normalization (or any other normalization). (See Fig. 2.)

If $\boldsymbol{\mathcal{K}}$ is a convolutional-type kernel, the architecture has similarities with the FNet introduced in Lee-Thorp et al. [77] though these connections have not been explored in the context of parametric PDEs. The addition of a skip connection in the FNOs architecture has been previously investigated in the work of You et al. [118]. However, the specific sequential structure used in here is not presented in the previous work. To provide a comprehensive analysis, we include the ResNet version of FNO in the ablation test (see Section 4.6) to evaluate its performance alongside the other described architectures. It is worth noting that similarities of the

[5] This has not been done in the previously mentioned paper.

skip connection in the work of You et al. [118] can also be drawn with *s*FNO + εI. For example, the skip connection can also be interpreted as unrolling Newton's method, see [60,96,7].

In comparing *s*FNO + εI with NOs and *s*NOs, we observe improvements in terms of loss and wavefield prediction across various settings (see Tables 6 to 11).

sNO + εI with stochastic depth Despite the fact that a neural architecture is theoretically universal, in practice, the parameters are updated using gradient-based methods that cannot exhaustively search the parameter space. It is, therefore, necessary to consider the limitations of the optimization algorithm and the training data, both of which may render the model non-universal in practice.

One possible approach to address this challenge is to enable the exploration of the optimization algorithm. Huang et al. [55] introduced the concept of *stochastic depth*, which involves randomly dropping entire layers of the network using Bernoulli RVs.[6] Practitioners have used this approach to facilitate the efficient training of large models. We conjecture that it also enables further exploration, which intuitively allows the algorithm to find better local minima (this procedure is in the spirit of an *adaptive rejection sampling*). We adopt this technique in the final network design:

$$\begin{cases} w'_\ell &= \big(\mathbf{I_d} + \mathbf{X}_\ell\, \sigma\circ(\boldsymbol{\mathcal{K}}_\ell + b_\ell)\circ \mathrm{N}\big)\circ v_\ell, & \text{(a)} \\ v_{\ell+1} &= \big(\mathbf{I_d} + \mathbf{X}_\ell\, f_\ell \circ \mathrm{N}\big)\circ w'_\ell & \text{(b)} \end{cases} \tag{7}$$

($\ell = 1, \dots, L$), $\mathbf{X}_\ell$ is a Bernoulli RV, such that $\mathbf{P}\{\mathbf{X}_\ell = 1\} = p_\ell$, and $\mathbf{P}\{\mathbf{X}_\ell = 0\} = 1 - p_\ell$ for $p_\ell \in [0,1]$, and $p_1 = 1$, $p_{\ell+1} \le p_\ell$. N is the layer normalization (or any other normalization). In Theorem 7.6 and Corollary 8.7, we shall show the relation of RVs in the generalization error bound (*in-distribution* and *out-of-distribution*).

Remark 2.1. As described in [55], at inference time we use the mean of the RVs.

3. Parametric time-harmonic wave equations, forward operator and data generation

Here we present a comprehensive overview of the coefficient to solution map associated with the Helmholtz equation, as well as the corresponding forward operator. Additionally, we outline the step-by-step procedure for generating the dataset and the guarantees in place to ensure: (a) independent realizations of the wave speed, and (b) sufficient regularity[7] in accordance with the theory of neural operators.

3.1. Time-harmonic wave equations

We consider the propagation of time-harmonic acoustic waves for two dimensional domain $D \subset \mathbb{R}^2$. The waves are given by the (scalar) pressure field p and (vector) particle velocity $\boldsymbol{v}$ solutions to [36,92]

$$\begin{cases} -\mathrm{i}\omega\rho(\boldsymbol{x})\,\mathbf{v}(\boldsymbol{x},\omega) - \nabla \mathrm{p}(\boldsymbol{x},\omega) = 0 & \text{in } D \quad \text{(a)}, \\ -\dfrac{\mathrm{i}\omega}{\kappa(\boldsymbol{x})}\mathrm{p}(x,\omega) + \nabla\cdot\mathbf{v}(\boldsymbol{x},\omega) = f(\boldsymbol{x},\omega) & \text{in } D \quad \text{(b)}, \end{cases} \tag{8}$$

where f is the time-harmonic source of angular frequency ω, ρ is the density and κ the bulk modulus. The boundary of the domain $\partial D = \Gamma_1 \cup \Gamma_2$ is separated into two, following a geophysical configuration: a free-surface condition is imposed at the surface Γ_1 (that is the interface between the medium and the air), while absorbing boundary conditions [32] are imposed elsewhere (that is, to truncate the numerical domain), see Fig. 3. These conditions correspond to

$$\mathrm{p}(\boldsymbol{x},\omega) = 0 \qquad \text{on } \Gamma_1 \text{ (Dirichlet boundary condition)}, \tag{9a}$$

$$\left(\partial_\nu - \frac{i\omega}{c(\boldsymbol{x})}\right)\mathrm{p}(\boldsymbol{x},\omega) = 0 \qquad \text{on } \Gamma_2 \text{ (absorbing boundary conditions)}. \tag{9b}$$

Upon assuming constant density ρ, Problem (8) can be rewritten as the Helmholtz equation (see Faucher et al. [37, Remark 1]),

$$-\left(\Delta + \frac{\omega^2}{c(\boldsymbol{x})^2}\right)\mathrm{p}(\boldsymbol{x},\omega) = -\mathrm{i}\omega\rho\, f(\boldsymbol{x},\omega), \tag{10}$$

where c is the wave speed,

$$c(\boldsymbol{x}) = \sqrt{\frac{\kappa(\boldsymbol{x})}{\rho(\boldsymbol{x})}}. \tag{11}$$

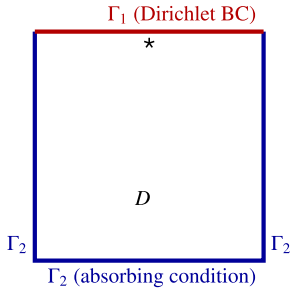


Fig. 3. Illustration of domain D: Dirichlet boundary condition is imposed on the top (red line, Γ_1), while absorbing boundary conditions are imposed elsewhere (blue line, Γ_2). The source ($\star$) is typically positioned near surface. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

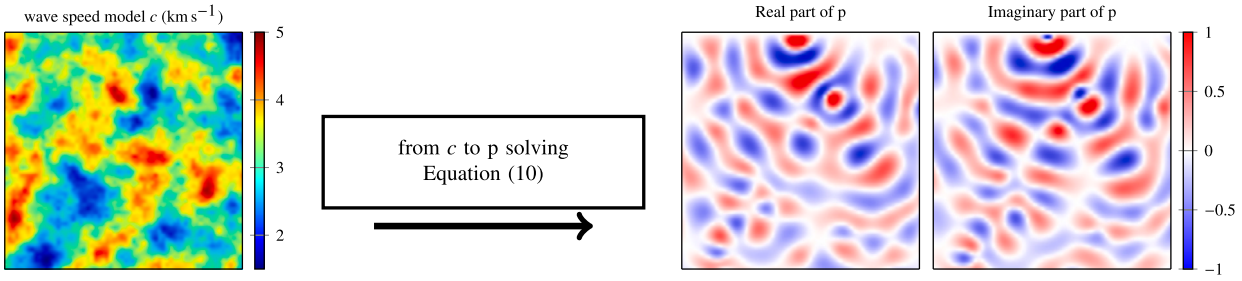


Fig. 4. Illustration of the full-wave dataset for experiment that considers a computational domain of size 1.27×1.27 km^2 with a source near surface. The wave speed and pressure field are represented on a Cartesian grid of size 64×64 with a grid step of 20 m. The complete dataset corresponds to 50 000 couples made up of a wave speed model and associated acoustic wave.

3.2. *Wave speed to solution map,* $\mathcal{G} : c \mapsto \mathrm{p}$

In the first experiment, the source f is fixed, as well as the frequency ω. The operator, $\mathcal{G}$ is defined as a mapping from the wave speed model c to the associated wavefield p, Fig. 4. That is, it gives the solution to the wave equation (10) with boundary conditions Equation (9) for a given physical model c in the entire domain D. $c \mapsto \mathcal{G}(c) = \mathrm{p}$. See Fig. 4 for an illustration of the operator when c is a realization of a Gaussian random field.

The dataset corresponds to N couples of wave speed and pressure field, denoted as, $(c_k, \mathrm{p}_k)_{k=1,\dots,N}$. The pressure field, p_k is obtained by solving (10) with the corresponding wave speed c_k. We use the *hybridizable discontinuous Galerkin method* (HDG, [36]) and the (open-source) software `hawen` [35], to obtain p_k. The source f in (10) is a fixed point-source, and the frequency is set to 15 Hz. We have the following configuration:

$$\text{Experiment with } \mathcal{G} \begin{cases} \text{2D domain of size } 1.27 \times 1.27 \text{ km}^2 \\ \text{50 000 GRF wave speeds generated, imposing } 1.5 \text{ km s}^{-1} \leq c(x) \leq 5 \text{ km s}^{-1} \\ \text{The data are p that solve Equation (8) at frequency } \omega/(2\pi) = 15 \text{ Hz.} \end{cases} \tag{12}$$

To ensure a statistical learning framework, we generate independent identically distributed realizations of a Gaussian random field (GRF) as our wave speed. The process is described in Section 3.4.

3.3. *Forward operator* $\mathcal{F}^f_\omega : (c, f, \omega) \mapsto \{\mathrm{p}(\boldsymbol{x}_j, \omega, f)\}_{j=1,\dots,n_{\mathrm{rcv}}}$

In the following, the term *forward operator* refers to the forward operator in the context of the study of the inverse problem for the Helmholtz equation [10] (which maps parameter and source to the data) $\mathcal{F}^f_\omega$ at frequency ω for a source f such that, $\mathcal{F}^f_\omega(c) = \mathrm{p}^f_\omega|_\Sigma$. The model parameter is the wave speed c from (10), and Σ corresponds to a discrete set of receiver locations. That is, $\mathrm{p}^f_\omega|_\Sigma = \left\{ \mathrm{p}^f_\omega(x_1), \dots \mathrm{p}^f_\omega(x_{n_{\mathrm{rcv}}}) \right\}$, where x_i is the position of the i^{th} receiver for a total of n_{rcv}. For notation, we introduce the *restriction operator* $\mathcal{R}$, which reduces the fields to the set of receivers positions, Σ, such that $\mathcal{R}(\mathrm{p}) = \mathrm{p}|_\Sigma$.

The dataset is composed of N_{src} sources, denoted as f_ℓ and consists of N pairs of wave speed and restricted pressure field, that is $\left(c_k, \mathcal{R}\left(\mathrm{p}^{f_\ell}_k\right)\right)_{k=1,\dots,N;\ l=1,\dots,N_{\mathrm{src}}}$. The restricted pressure field, $\mathcal{R}\left(\mathrm{p}^{f_\ell}_k\right)$, is obtained by solving (10) with the corresponding wave

[6] RVs refers to Random Variables.
[7] Nonnegative Sobolev spaces.

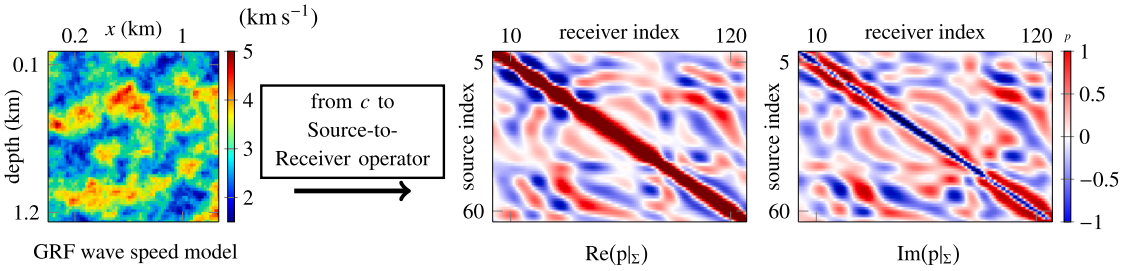


Fig. 5. Illustration of forward operator experiment that considers a computational domain of size 1.27×1.27 km^2 with 64 source near surface, and 128 receivers located slightly beneath the sources' location. The illustration of the wave field represent the "matrix" response, each row corresponds to a source, and each column to the pressure field registered by the receivers' line.

speed c_k and source f_ℓ, then restricted at the set Σ. Similar to the experiment with the full modeling operator, the wave speeds are independent identically distributed realizations of a GRF (see Section 3.4). The data set is illustrated in Fig. 5.

$$\text{Experiment with } \boldsymbol{\mathcal{F}} \begin{cases} \text{2D domain of size } 1.27 \times 1.27 \text{ km}^2 \\ \text{50 000 GRF wave speeds generated, imposing } \ 1.5 \text{ km s}^{-1} \leq c(x) \leq 5 \text{ km s}^{-1} \\ \text{64 point-sources, located at a fixed depth of 10 m, and 20 m apart along the width} \\ \text{The data are } \mathcal{R}\left(\mathrm{p}_k^{f_\ell}\right) \text{ that solve Equation (8) at frequency } \omega/(2\pi) = 15 \text{ Hz.} \\ \text{The line of receivers } \Sigma \text{ is located at a fixed depth 10 m, and 10 m apart along the width} \end{cases} \tag{13}$$

3.4. Wave speeds as Gaussian random fields (GRF) and Whittle–Matérn fields

The wave speed is obtained as the composition of linear transformation and an independent realizations of GRF with the Whittle–Matérn covariance kernel C_ν [40,13,87]. The linear transformation T, is a linked function to ensure that the wave speed is nonnegative, $\mathrm{T} \circ Z \geq 0$ and $Z \sim$ GRF. A most sophisticated version of this idea is presented in Abraham and Nickl [1] for the *conductivity* in the Calderón problem, the conductivity is also restricted to be nonnegative.

An introduction of Gaussian random fields is presented in Appendix A.5. We briefly discuss the Whittle–Matérn kernel, ([15] and [21, Sec. 2.2.3]). A real-valued Gaussian random field Z defined on a spatial domain $D \subset \mathbb{R}^d$ is a Whittle–Matérn field if its covariance function $C : D \times D \to \mathbb{R}$ is given by

$$\mathrm{C}_\nu(\boldsymbol{x}, \boldsymbol{x}') = s^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\, r_\lambda(\boldsymbol{x}, \boldsymbol{x}')\right)^\nu \mathrm{K}_\nu \left(\sqrt{2\nu}\, r_\lambda(\boldsymbol{x}, \boldsymbol{x}')\right) \tag{14}$$

here, s is the variance of the process, Γ is the gamma function [6], K_ν is the modified Bessel function of the second kind [16,5], and ν is a positive parameter. Furthermore, ν is known as the *smoothness* of the random field, and r_λ is the distance, that is, given $\boldsymbol{x} = (x_1, \ldots, x_d) \in \mathbb{R}^d$, and $\boldsymbol{x}' = (x'_1, \ldots, x'_d) \in \mathbb{R}^d$, is defined as

$$r_\lambda(\boldsymbol{x}, \boldsymbol{x}') := \sqrt{\sum_{i=1}^{d} \left(\frac{x_i - x'_i}{\lambda_i}\right)^2}, \tag{15}$$

where the vector coefficient $\lambda = (\lambda_1, \ldots \lambda_d)$ defines the correlation length along two points in $\mathbb{R}^d$.

The regularity of the Whittle-Matérn field and its generalizations can be readily obtained by viewing the field as a stochastic partial differential equation (SPDE for short). That is, a GRF Z with covariance function of Whittle-Matérn solves the fractional SPDE,

$$\boldsymbol{L}^\beta Z = \boldsymbol{d}\mathbf{W} \quad \text{in } D, \quad Z = 0 \quad \text{on } \partial D, \tag{16}$$

for $4\beta = d + 2\nu$ and $d \in \{1, 2, 3\}$ the spatial dimension, $\boldsymbol{d}\mathbf{W}$ is Gaussian white noise, and $\boldsymbol{L}^\beta$ is a second-order elliptic differential operator.

In the case of $\boldsymbol{L} = -\Delta + \kappa^2$,[8] each realization of Whittle–Matérn field, defined in (14) ($\lambda_i = 1$), coincides with the solution of (16), and belongs to the Sobolev space $H^{2\beta - d/2 - \epsilon}(D)$ (**P**-a.s.[9]) [15, Remark 2.4]. In the more general case, when $\boldsymbol{L} = -\nabla \cdot (\sigma \nabla) + \kappa^2$, which is referred to as the generalized Whittle-Matérn field, the regularity of its solution can be established under mild assumptions on σ, κ, and the boundary ∂D, see [22, Lemma 4.2].

[8] Where Δ denotes the Laplacian.

[9] Almost surely with respect to the probability measure **P**.

From (16) and the associated SPDE, we know that the Whittle–Matérn field lies in a "*nicer*" space than Gaussian white noise. The use of GRF is motivated by the following. (a) In our construction the wave speed is generated as $c = \mathrm{T} \circ Z^{\beta}$, in where T is an affine transformation and Z^{β} lies in $H^{2\beta - d/2 - \epsilon}(D)$ a.s., specifically $d = 2$, $\beta = 1$ so $Z^{\beta} \in H^{1-\epsilon}(D)$ a.s., and for all $\epsilon \in [0, 1]$, the wave speed lies on non-negative Sobolev spaces. The field satisfies the conditions in [65, Theorem 2.5]. If we would have $c \in \mathrm{L}^{\infty}(D)$, the operator $\mathcal{G}$ would not be covered by the universality in [65].[10] (b) GRF samples are easily generated, and for the case of Whittle–Matérn field, the variance, smoothness, and correlation length are easy to control; this observation plays a crucial role in Section 5 to test the *out-of-distribution behavior*. (c) This distribution is independent of the grid resolution. (d) GRF are often used in Bayesian statistics as prior probability measures with covariance kernels related to the Laplace operator ([97, Section 2.1], [24] and [21]).

Remark 3.1. The parameters in the experiments are the following: $d = 2$, $s = 1$, $\lambda_1 = \lambda_2 = 0.1$, and smoothness coefficient $\nu = 1$. For the implementation of Gaussian fields, see [30,87,69], and particularly [15].

4. Training and testing in-distribution for $\mathcal{G}$

In this section, our focus lies on training the architectures to accurately predict the coefficient to solution map $\mathcal{G}$ for the Helmholtz equation at a frequency of 15 Hz, (10). For the sake of completeness, more experiments with different frequencies and domain's configuration are presented in Appendix E.4. Throughout this section and Appendix E.4, all the models are tested with in-distribution data. However, we significantly increase the test set compared to traditional applications of deep learning.[11] We choose a test set of the same size than our training. This choice enables us to obtain more reliable estimates of the neural operators' generalization capabilities specifically for in-distribution data. A detailed analysis of the generalization to in-distribution data is presented in Section 8.

Remark 4.1. The code is publicly available at [75], and the dataset is located at [74].

Remark 4.2. In our experiments, we adhere to specific constraints. When adjusting the parameters of the networks, the increase in the parameter count is typically negligible, adding around 100 additional parameters to maintain comparability with the base neural operator. If we increase the number of layers, it is based on mathematical considerations, particularly when incorporating stochastic depth. *We consciously refrain from increasing the training epochs or the size of the training dataset.* Our emphasis is on making fundamental changes to the network architecture rather than compensating for these alterations by merely expanding the model's capacity, dataset size, or training time.

4.1. Neural operator "prediction" of the wavefield

Upon the previous constraints, we conducted training on the wave dataset as described in the previous section for all the neural networks outlined in Section 2. The results shown in Fig. 6 clearly demonstrate that each architecture leads to a superior reconstruction of the wave field. The figure displays only the real part of the wave field. For the approximation of both the real and imaginary parts of the pressure field, we refer to Appendix E.6.

4.2. Hyperparameters of the neural networks

The summary of parameters used in the training is presented in Table 2.

The Fourier modes represent the truncated Fourier modes in the approximation of the integral kernel per layer $\boldsymbol{\mathcal{K}}_{\ell}$ as described in [82]. The number of layers represents the compositions of equations of the form Equations (4) to (7), The positional encoder means that the wave speed, c, is input in the neural operators as a couple $\{\big(c(x_i, y_k), \mathrm{T}(x_i), \mathrm{T}(y_k)\big)\}_{i,k=1}^{n}$. Here, T denotes an affine transformation applied to each grid realization to move the grid to the interval $[0, 1] \times [0, 1]$ usually for training stability.

The feature space refers to the range of the lifting operator (as explained after Equation (4)), denoted as $\boldsymbol{R}$.[12] It maps $\mathbb{R}^3$ to $\mathbb{R}^{36}$, with $\big(c(x_i, y_k), \mathrm{T}(x_i), \mathrm{T}(y_k)\big)$ being transformed to $v_1(z_1, \ldots, z_{36}) = \boldsymbol{R}\left(c(x_i, y_k), \mathrm{T}(x_i), \mathrm{T}(y_k)\right) \in \mathbb{R}^{36}$. It is implemented using a 2-layers MLP with weight matrices, $W_1^{\boldsymbol{R}} \in \mathbb{R}^{18\times 3}$ and $W_2^{\boldsymbol{R}} \in \mathbb{R}^{36\times 18}$, and bias $b_1^{\boldsymbol{R}} \in \mathbb{R}^{18}$, $b_2^{\boldsymbol{R}} \in \mathbb{R}^{18}$.

The projection, $\boldsymbol{Q}$,[13] maps $v_L(z_1, \ldots, z_{36}) \in \mathbb{R}^{36}$ to $c(x_i, y_k) \in \mathbb{R}^2$, with a linear affine transformation such that $W^{\boldsymbol{Q}} \in \mathbb{R}^{2\times 36}$ and $b^{\boldsymbol{Q}} \in \mathbb{R}^2$. We associate $\mathbb{R}^2$ with $\mathbb{C}$ to recover the imaginary and real part of the solution.[14]

In our experiments, we do not implement dropout. For stochastic depth (also known as drop path), the random variables have a linear decay. The probability is set as follows in the experiments, $\mathbf{P}\{\mathbf{X}_1 = 1\} = 1$ for the first layer, and $\mathbf{P}\{\mathbf{X}_L = 1\} = 0.7 = 1 - 0.3$ for the last layer, [114]. For the layers in between, a survival probability is assigned using linear interpolation.

[10] L^{∞} is not a separable Banach space.

[11] We deliberately avoid using the traditional 80:20 split of training and test data.

[12] Lifting map, following notation in [65].

[13] Projection map, following notation in [65].

[14] Pressure field.

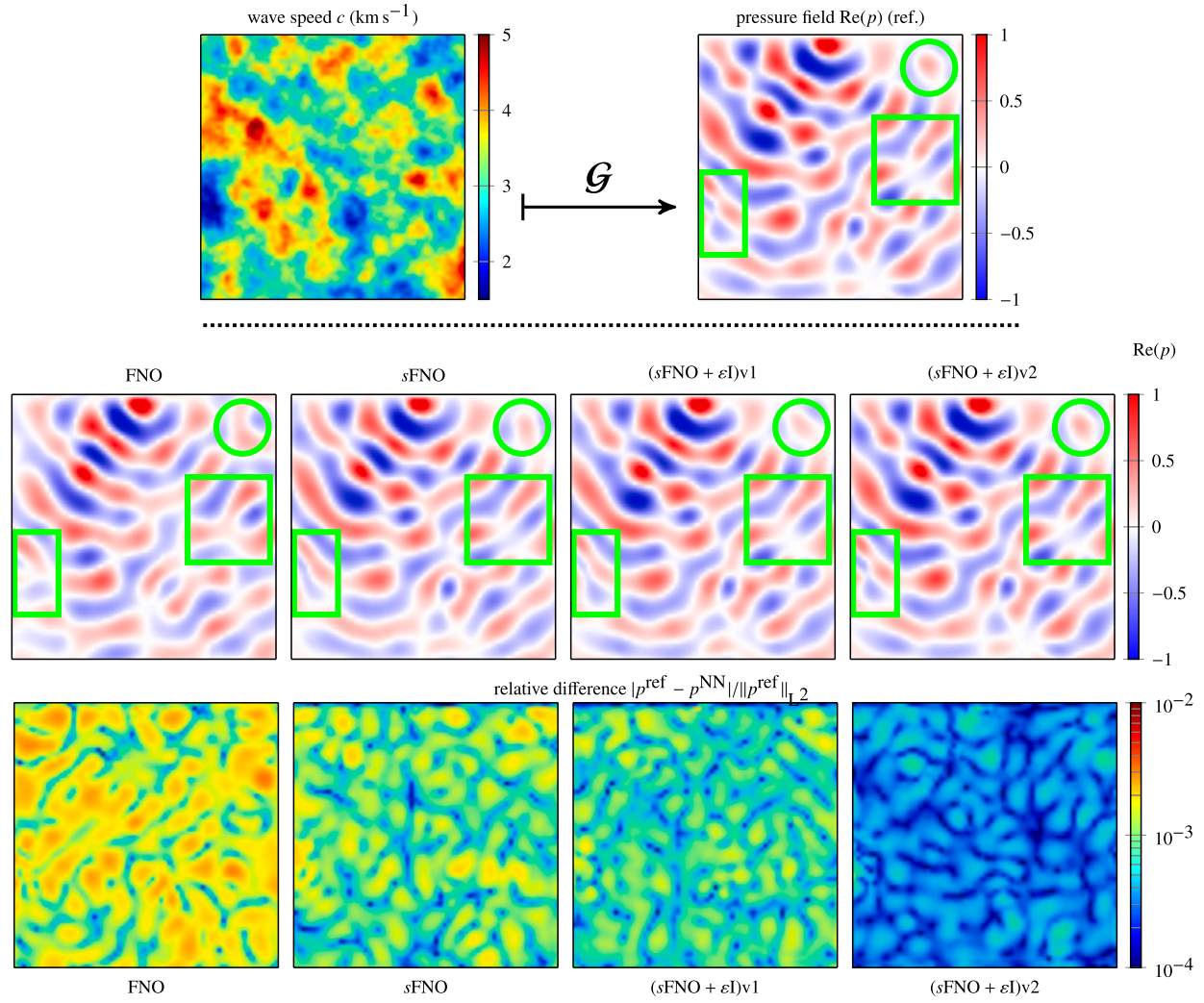


Fig. 6. Comparison of the reconstructed wavefields obtained with the different architectures (middle row) and relative error with the reference solution (bottom row). The circles and rectangles serve as a visual aid to highlight the distinction in the propagation of waves. The dataset corresponds to wave propagation from Gaussian Random Field realizations of wave speed in a domain of size 1.27×1.27 km^2, with reference wavefield obtained by solving the wave PDEs with software `hawen` [35] (top row).

The architectures used in our study have similar parameter counts, except for sFNO + εI with stochastic depth (v2). This architecture consists of four stages, each containing a different number of blocks. Specifically, the number of blocks in each stage is $k \in [3, 3, 9, 3]$, and the blocks follow Equation (7). This results in a total of 21 layers, with each layer truncated to 12 principal modes in the Fourier expansion of $\boldsymbol{\mathcal{K}}_\ell$, and the feature spaces of dimension 36.[15] The parameter for the other networks, namely FNO, sFNO, and sFNO + εI without stochastic depth (v1), are essentially the same.

Number of parameters of the neural operators As mentioned earlier, both sFNO and sFNO + εI have a similar "size" to FNO when stochastic depth is not considered. However, the significant difference arises when stochastic depth is incorporated, resulting in a much deeper neural network. In all the networks, the lifting and projection components have parameter counts of 756 and 685, respectively.

The main part of the networks, which encompasses the "operator" layers described in Equations (4) to (7), are divided into two categories: layers without stochastic depth, and layers with stochastic depth. In the former type (Equations (4) to (6)), the parameters are fixed at 1.5 million for all the layers, while in the case of sFNO + εIv2 (Equation (7)), the parameter count increases to 8.1 million.

[15] That is for each (x_i, y_k) we have $\boldsymbol{R}\left(c(x_i, y_k), \mathrm{T}(x_i), \mathrm{T}(y_k)\right) \in \mathbb{R}^{36}$.

Table 2
Architectures' parameters. The networks recovered the real, and imaginary part of the pressure field, i.e., the output is a vector field in $\mathbb{R}^2$ which can be associated with $\mathbb{C}$, and the projection operator is simplified by a linear layer instead of a MLP to speed up the training process. The only architecture that differs is (sFNO + εI) version 2 (with stochastic depth).

Model	FNO	sFNO	(sFNO + εI)v1	(sFNO + εI)v2
Fourier modes: 12	✓	✓	✓	✓
Layers: 4	✓	✓	✓	[3, 3, 9, 3]
Features: 36	✓	✓	✓	[36, 36, 36, 36]
GeLU	✓	✓	✓	✓
Positional Encoder $[0,1]^2$	✓	✓	✓	✓
Lifting	$3 \mapsto 18 \mapsto 36$	$3 \mapsto 18 \mapsto 36$	$3 \mapsto 18 \mapsto 36$	$3 \mapsto 18 \mapsto 36$
Proj.	$36 \mapsto 2$	$36 \mapsto 2$	$36 \mapsto 2$	$36 \mapsto 2$
Dropout	✗	✗	✗	✗
DropPath	✗	✗	✗	0.3

Table 3
Magnitude of the relative L^2-norm. Multiple realizations of the trained networks with different seeds. Each row represents a different realization, and the values correspond to the test loss after training. The visualization of the table is presented in Fig. 7.

FNO	sFNO	(sFNO + εI)v1	(sFNO + εI)v2
0.174050	0.119564	0.097434	0.046988
0.180532	0.115850	0.089249	0.042121
0.145947	0.110553	0.096739	0.041300
0.153028	0.102238	0.097211	0.045696
0.144907	0.102998	0.102930	0.049157
0.172738	0.103829	0.092119	0.037969

4.3. Training of the experiment

For all the architectures we employ the *AdamW* optimizer [88] with an initial learning rate of 10^{-3}. We utilize a linear step scheduler with parameters: step size $= 40$, and a multiplicative factor of learning rate decay of $\gamma = 0.5$.

The number of epochs is set to 100 (300 epochs yielded the best results for sFNO + εI with stochastic depth, but this is not documented here as we try to keep the same parameters across networks). In all architectures, we apply a small L^2 weight regularizer with a parameter of 10^{-5}. The training process is conducted using $25,000$ out of $50,000$ generated samples Equation (12), while $5,000$ samples are used for validation, and $20,000$ for testing. Our *testing dataset is substantially larger* than what is typically encountered in the machine learning literature. This choice reflects our objective of showcasing the networks' generalization capabilities.

4.4. Multiple random initializations

To ensure the consistency of our results, we train each network using six different random initializations of the parameters and in consequence, different trajectories of the optimization algorithm. The trend is consistently observed across all initializations, as depicted in Fig. 7. The values of the relative L^2-loss among multiple training paths can be found in Table 3.

4.5. Visualization of the loss landscape

The observed differences in the performance of the four considered architectures prompted us to study their respective learning landscapes in search of structural characteristics that could explain the results. To that end, we sampled the training loss in a two-dimensional domain spanned by the first two principal components of the learning trajectory [79]. By construction, this planar domain best captures the portion of the landscape visited during the training of each model and, therefore, may offer valuable insight into the training convergence.

Corresponding results are shown in Fig. 8.

As can be seen, the landscapes fall into three major categories. The FNO landscape is characterized by the presence of a shallow and irregular crease-like structure that runs across the domain. The sFNO and sFNO + εI landscapes share remarkable similarities, which is consistent with the similar loss values shown in Table 3. Both possess a well-delineated and deeper convergence basin. Finally, the sFNO + εI v2 landscape exhibits a crease-like structure similar to the one seen in the FNO landscape, but its topology is much simpler, and the central anisotropic basin is the deepest of all considered models. We refer the reader to Appendix E.1 for more visualization.

The level sets are plotted with a spectral (rainbow) color map that contrasts with the underlying landscape color scale (shown). The associated values are visible. The individual points along the trajectory in each landscape show every 10th training epoch, and

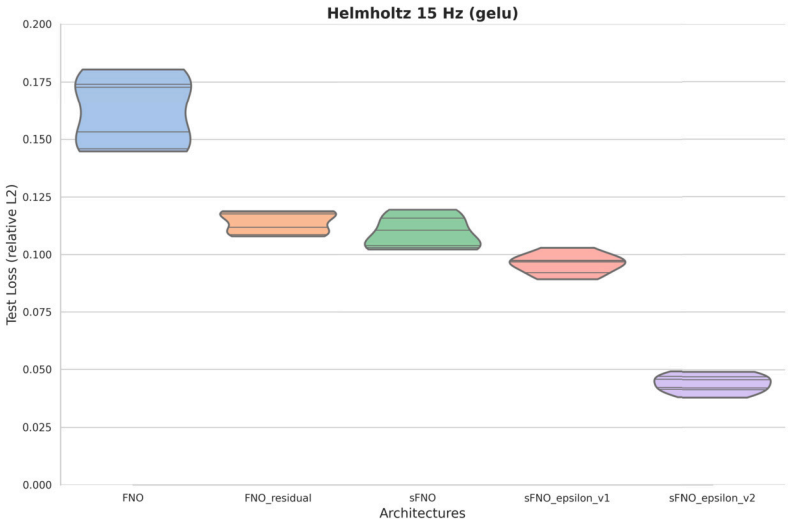


Fig. 7. Violin plot [52] of the test-loss in Experiment 15 Hz of (10). Each architecture is trained 6 times, the rel. L^2-loss, $|\mathcal{G}^{\text{ref}} - \mathcal{G}^{\text{approx}}|_{L^2} / |\mathcal{G}^{\text{ref}}|_{L^2}$, on the test set.

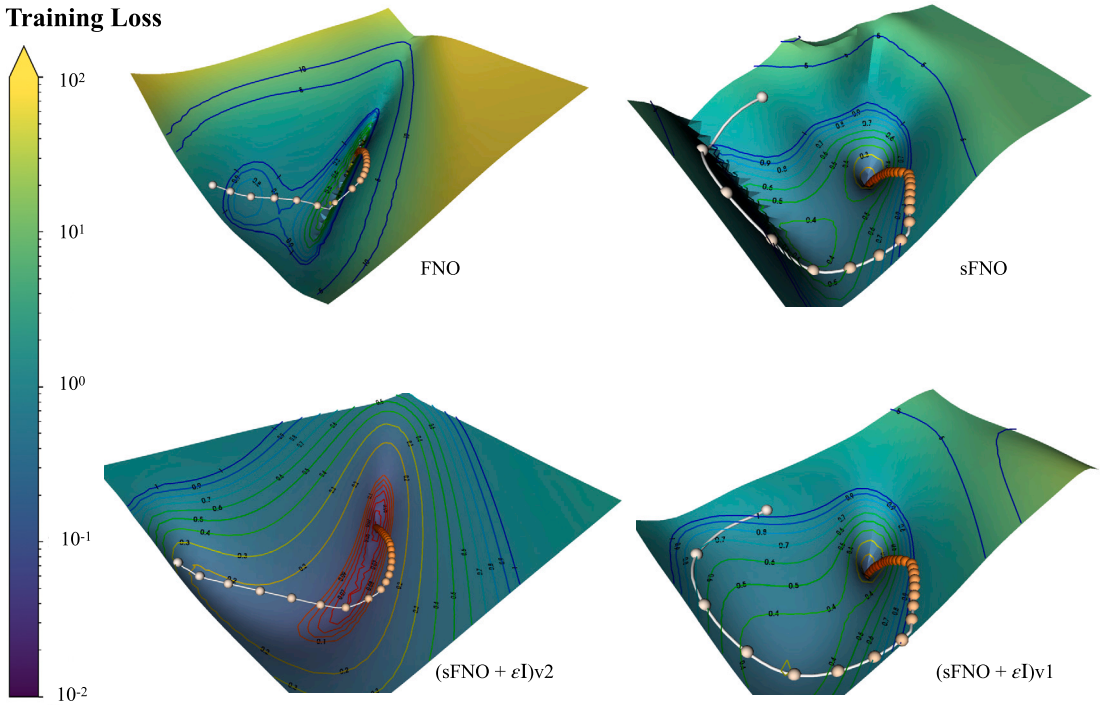


Fig. 8. Learning landscapes of the four considered models. The loss is visualized in logarithmic scale. Level sets reveal significant differences in topologies.

the orange color saturation encodes the epoch. The increased geometric complexity along the diagonal crease present in the FNO and *s*FNO + εI v2 landscapes was handled with a refined sampling of the training loss in the corresponding area. The principal components that span the two-dimensional sampling domain were computed by splitting real and imaginary parts of the layers' complex weights to form the large column vector representations of each model in the covariance matrix.

4.6. Ablation study

We have already conducted a study of ablation to some extent by the design of the networks. For example, when the skip connection is removed, *s*FNO + εI without stochastic depth (V1) reduces to *s*FNO. Similarly, when we set $\mathbf{P}\{\mathbf{X}_\ell = 1\} = 1$ for all layers ℓ, then (*s*FNO + εI)v2 reduces to (*s*FNO + εI)v1.

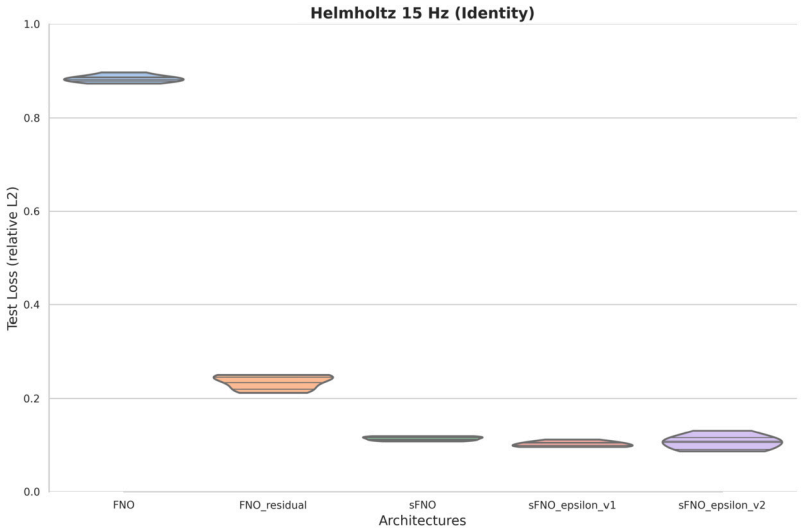


Fig. 9. Test-loss with no activation 15 Hz of (10). Each architecture is trained 6 times, the rel. L^2-loss on the test set.

Table 4
Test-loss with no activation 15 Hz of (10). Each architecture is trained 6 times, the rel. L^2-loss on the test set.

FNO	FNO residual	*s*FNO	(*s*FNO + εI)v1	(*s*FNO + εI)v2
0.873029	0.219886	0.107912	0.096015	0.130741
0.880681	0.250052	0.110204	0.096212	0.115885
0.885039	0.212233	0.119039	0.099064	0.105632
0.896779	0.245763	0.118050	0.105330	0.086690
0.878160	0.233708	0.112514	0.111709	0.108062
0.886912	0.250634	0.115919	0.104147	0.090379

In the following, we explore the changes in activation functions, with a particular focus on the identity activation, $\sigma(x) = x$ for the Fourier layers. Additionally, we investigate the behavior of the residual version of FNO as described in You et al. [118]. The parameters are prescribed in Table 2.

We adopt a strategy similar to Section 4.4, training each network multiple times with different random seeds to ensure the consistency of our empirical findings.

In the case where no activation is used in the Fourier layers, we observe that *s*NO achieves a lower relative L^2 loss compared to FNO supported by results in Fig. 9 and Table 4. Notably, even when FNO is trained with a non-linear activation function (as proposed in Li et al. [82]), *s*NO consistently exhibits a significantly smaller test loss. This distinction can be observed by comparing the first violin plot in Fig. 7 (detailed values are presented in Table 3), Fig. 10, and Fig. 11 representing FNO with GeLU, leaky-ReLU, and ReLU, activation functions respectively, with the third violin plot in Fig. 9 (see values in Table 4). Additionally, the second violin plot in Fig. 9 also presents the residual implementation of FNO. There is a significant improvement observed over FNO.

The results obtained from Figs. 7 to 11 demonstrate that the residual architecture aligns with the findings of You et al. [118]. In every case, we observe a noticeable improvement in the relative L^2-loss.

Among the different architectures, *s*NO achieves the most significant improvement compared to the previous architecture. Within the activation functions, Leaky-ReLU and ReLU exhibit the most significant change when transitioning from the architecture FNO to *s*NO. In contrast, when the identity activation is used in the Fourier layers, *s*FNO + εI with stochastic depth does not show a noticeable improvement compared to its counterpart (*s*FNO + εI)v1. However, in all cases where a nonlinear activation is employed, (*s*FNO + εI)v2 consistently outperforms other architectures without any sign of overfitting. Notably, for ReLU and Leaky-ReLU activations, the potential benefits of the skip connection are difficult to observe, when compared to *s*NO.

5. Testing out-of-distribution analysis (OOD) for $\mathcal{G}$

In this section, we study the out-of-distribution (OOD for short) behavior for all the architectures. Specifically, we investigate how the models perform when faced with perturbations in the covariance operators of the Gaussian fields used for training. Our findings demonstrate that the *s*NO + εI architecture with stochastic depth shows resilience to these perturbations. However, despite these encouraging results, the theoretical understanding of the impact of Bernoulli's random variable on the generalization ability of the neural operators in the context of OOD is still in its early stages. To provide further insights, we present a theoretical analysis in Section 7 for Gaussian measures.

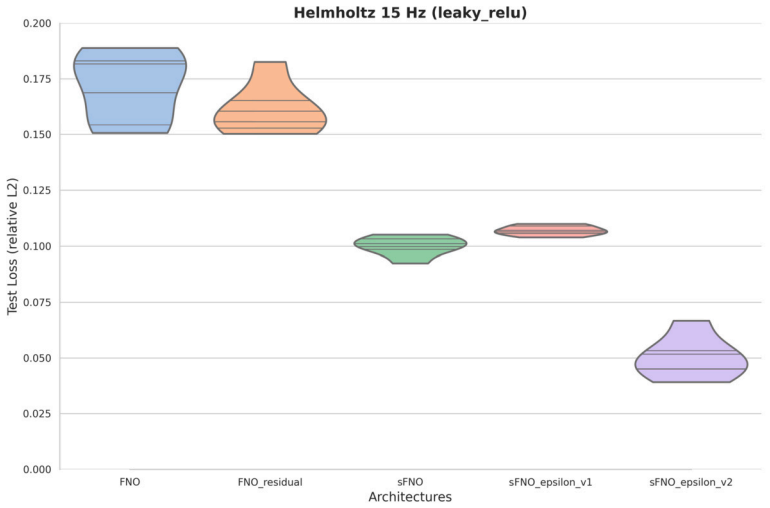


Fig. 10. Test-loss with Leaky-ReLU 15 Hz of (10). Each architecture is trained 6 times, the rel. L^2-loss on the test set.

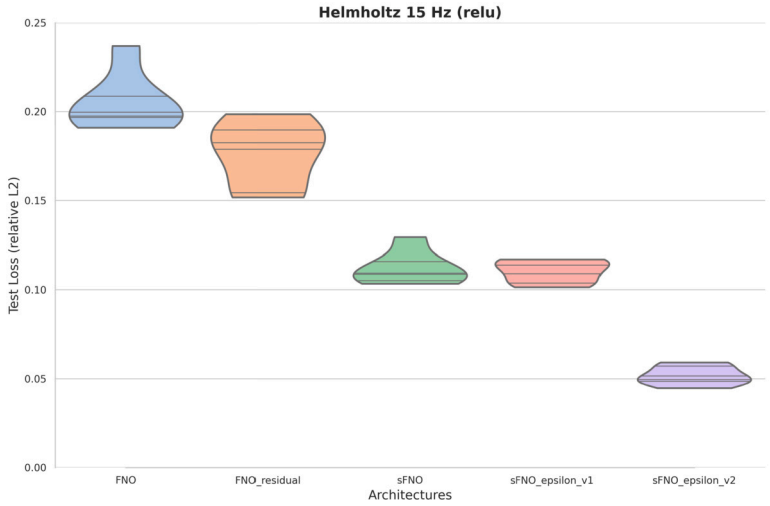


Fig. 11. Test-loss with ReLU 15 Hz of (10). Each architecture is trained 6 times, the rel. L^2-loss on the test set.

We recall from Section 3.4, and particularly (14) that the Whittle-Matérn fields have three essential parameters: variance s, smoothness ν, and correlation range λ. As mentioned in Section 4, the neural operators were trained using Gaussian random fields (GRF) with an isotropic Whittle-Matérn covariance operator such that the wave speed c varied between 1500 and 5000, $\lambda = (1, 1)$, and the smoothness coefficient is $\nu = 1$. *Throughout this section the models obtained in Section 4 are not retrained.* We refer to the settings in (12), and Remark 3.1 for details.

5.1. OOD experiments with different correlation and affine transformation

We investigate the effect of changing the correlation parameter λ and the range on which the wave speeds vary. Adjusting λ modifies the correlation range of the field. The scenario where $\lambda_1 \neq \lambda_2$ in equation (14) is particularly interesting as it introduces non-euclidean distances and leads to the generation of anisotropic fields. The range of the wave speeds are adjusted using a different affine transformation denoted as T, as explained in Section 3.4. Here, we keep the smoothness coefficient fixed, ensuring that the wave speeds remain within the same Sobolev space as the training data. Then, the new realizations of the wave speeds are given by $c' = \mathrm{T}' \circ Z^{(\lambda_1, \lambda_2)}$, in which T' changes the wave speed interval, and (λ_1, λ_2) the correlation of points in the domain.

To generate new samples of the wave speed c, we sample GRF following the parameters described in Table 5. For each family, we generate 100 samples and we obtain the corresponding solution of Helmholtz using the software `hawen`. The smoothness of c, domain's configuration D, source position, and frequency ω are fixed following (12).

Table 5
Parameters for the experiments out-of-distribution. $\lambda = (\lambda_1, \lambda_2)$ is defined in Equation (15). The parameter ν is fixed to 1.

GRF model	λ_1	λ_2	wave speed interval
Training (baseline)	0.10	0.10	[1500, 5000]
OOD family 1	0.20	0.20	[1500, 5000]
OOD family 2	0.10	0.20	[1500, 5000]
OOD family 3	0.20	0.20	[2000, 3500]
OOD family 4	0.10	0.20	[2000, 3500]
OOD family 5	0.10	0.30	[2000, 6000]
OOD family 6	0.25	0.75	[2000, 6000]

Table 6
Relative test loss of three networks tested with the probability defined by family 1.

OOD 1	FNO	sNO	(sFNO + εI)v1	(sFNO + εI) v2
model 1	0.6689	0.6025	0.5341	0.2502
model 2	0.6842	0.5437	0.5451	0.2347
model 3	0.6817	0.5837	0.5404	0.2428

Table 7
Relative test loss of three networks tested with the probability defined by family 2.

OOD 2	FNO	sNO	(sFNO + εI)v1	(sFNO + εI) v2
model 1	0.6602	0.6106	0.5438	0.2340
model 2	0.6715	0.5644	0.5561	0.2239
model 3	0.6726	0.5959	0.5509	0.2407

Table 8
Relative test loss of three networks tested with the probability defined by family 3.

OOD 3	FNO	sNO	(sFNO + εI)v1	(sFNO + εI) v2
model 1	0.5116	0.4368	0.3645	0.1324
model 2	0.4757	0.3490	0.3678	0.1220
model 3	0.5001	0.4061	0.3490	0.1368

Table 9
Relative test loss of three networks tested with the probability defined by family 4.

OOD 4	FNO	sNO	(sFNO + εI)v1	(sFNO + εI) v2
model 1	0.5249	0.4685	0.3845	0.1335
model 2	0.4992	0.3798	0.3869	0.1249
model 3	0.5146	0.4264	0.3713	0.1376

Empirical analysis of OOD for each family For the experiment, we selected three out of the six previously trained models (specifically, the first three models in Fig. 7) that utilized the GeLU activation function, see Section 4.4. We recall that we have obtained an estimation of the expected error within the distribution by evaluating the empirical loss in a test data set of the same size as the training data set,[16] for more details we refer to the training baseline in Table 3, and Section 4.3.

By sampling multiple realizations of new random fields according to the families outlined in Table 5 we are able to estimate the expected error of the trained network with respect to these new probability distributions, and in consequence the robustness of the networks towards these changes. This enables us to assess the models' performance on the new samples and evaluate its generalization capabilities beyond the in-distribution data.

The empirical results for all architectures are presented in Tables 6 to 11.

The families presented in Tables 9 to 11 exhibit anisotropy due to the difference in the values of λ_1 and λ_2. When considering the relative L^2 loss as a reference, it is evident that the trained FNOs perform significantly worse compared to other neural operators. This indicates that FNOs may struggle with generalizing to new distributions. However, we observe that the sNO + εI architecture

[16] We trained the models using a dataset of 25,000 samples and evaluated their performance on a separate test dataset also consisting of 25,000 samples.

Table 10
Relative test loss of three networks tested with the probability defined by family 5.

OOD 5	FNO	*s*NO	(*s*FNO + εI)v1	(*s*FNO + εI) v2
model 1	0.9248	0.8698	0.8827	0.3899
model 2	0.9471	0.8379	0.8209	0.3910
model 3	1.0488	0.9269	0.8130	0.4188

Table 11
Relative test loss of three networks tested with the probability defined by family 6.

OOD 6	FNO	*s*NO	(*s*FNO + εI)v1	(*s*FNO + εI) v2
model 1	0.9707	0.8903	0.9606	0.4426
model 2	1.0087	0.8851	0.8576	0.4585
model 3	1.1578	0.9831	0.8712	0.4864

Table 12
Parameters for the experiments out-of-distribution. $\lambda = (\lambda_1, \lambda_2)$ is defined in Equation (15). The parameter ν is changing.

GRF model	λ_x	λ_y	wavespeed interval	ν
OOD family 7	0.10	0.10	[1500, 5000]	0.5
OOD family 8	0.10	0.10	[1500, 5000]	3.5
OOD family 9	0.25	0.75	[2000, 6000]	0.5
OOD family 10	0.25	0.75	[2000, 6000]	3.5

coupled with stochastic depth demonstrates notable robustness when faced with changes in distribution across all the families. In particular, we notice for the experiments in Table 11 where both T' and λ are changed, the *s*FNO + εIv2 exhibits superior adaptability compared to other architectures, resulting in test losses that are half the values of any other neural operator.

OOD wave field "prediction" by the neural networks In Fig. 12, we present the wave field predictions of the trained networks from family 6. The figure showcases two samples from family 6, illustrating the shortcomings of the FNO in accurately reproducing the desired behavior. In particular, we emphasize the discrepancy within the green rectangle, which indicates a notable deviation between the predicted wave field and the ground truth. This discrepancy further highlights the limitations of the FNO in capturing the complex dynamics of the wave propagation from different distributions. Among the models considered, it is observed that only the *s*FNO + εIv2 model is capable of accurately predicting admissible wave propagation in the family 6.

5.2. OOD experiments changing the smoothness of the field

Here, we change the smoothness of the wave speed by modifying the parameter ν. We recall from (16) that the regularity of the field, β is directly connected with the dimension d of the domain (in our case $d = 2$), and the coefficient ν (in our training $\nu = 1$). Thus, by changing ν, we generate Gaussian random fields of different Sobolev regularity, than those using in the training dataset. Our experiments are divided into two categories. (a) We first keep all but ν parameters fixed, as described in Remark 3.1, that is, we only change the Sobolev class of the wave speed without altering any other factor (e.g. if the field is isotropic or anistropic). (b) Finally, we move the rest of the parameters, by following the description of the family 6 in the Table 5, the "*hardest*" family in terms of solution field prediction and test loss, see Table 11.

Empirical analysis of OOD for each family We follow a similar procedure as in the previous subsection. We select three out of the six trained models shown in Fig. 7) and evaluate their performance against 100 realizations of the wave speed for each of the new families described in Table 12. We notice that the first two families preserves the rest of the parameters as in our training baseline, while in the last two all the parameters are changed.

In the selection of $\nu = 0.5$ and $\nu = 3$, the random fields in families 7 and 9 have Sobolev regularity $H^{1/2-\varepsilon}(D)$, while the random fields Z in families 8 and 10 have Sobolev regularity $H^{3+1/2-\varepsilon}(D)$, almost surely. This is in contrast to the training data set, which lies almost surely in $H^{1-\varepsilon}(D)$ for any $\varepsilon > 0$. For families 9 and 10, the wave speeds $c' = \mathrm{T}' \circ Z^{(\lambda_1, \lambda_2)}$ are different on each aspect than the training set. The affine maps T' are different from those used in the training. The correlation range $\lambda = (0.25, 0.75)$ differs from the training correlation range, introducing anisotropy into the fields. Furthermore, the regularity of each realization in these families also varies from the baseline.

The empirical results for all architectures are presented in Tables 13 to 16.

OOD wave field "prediction" by the neural networks (different smoothness) We showcase the performance of the trained networks by presenting wave field predictions for family 9 and 10, which correspond to rough and smooth anisotropic fields, respectively. These

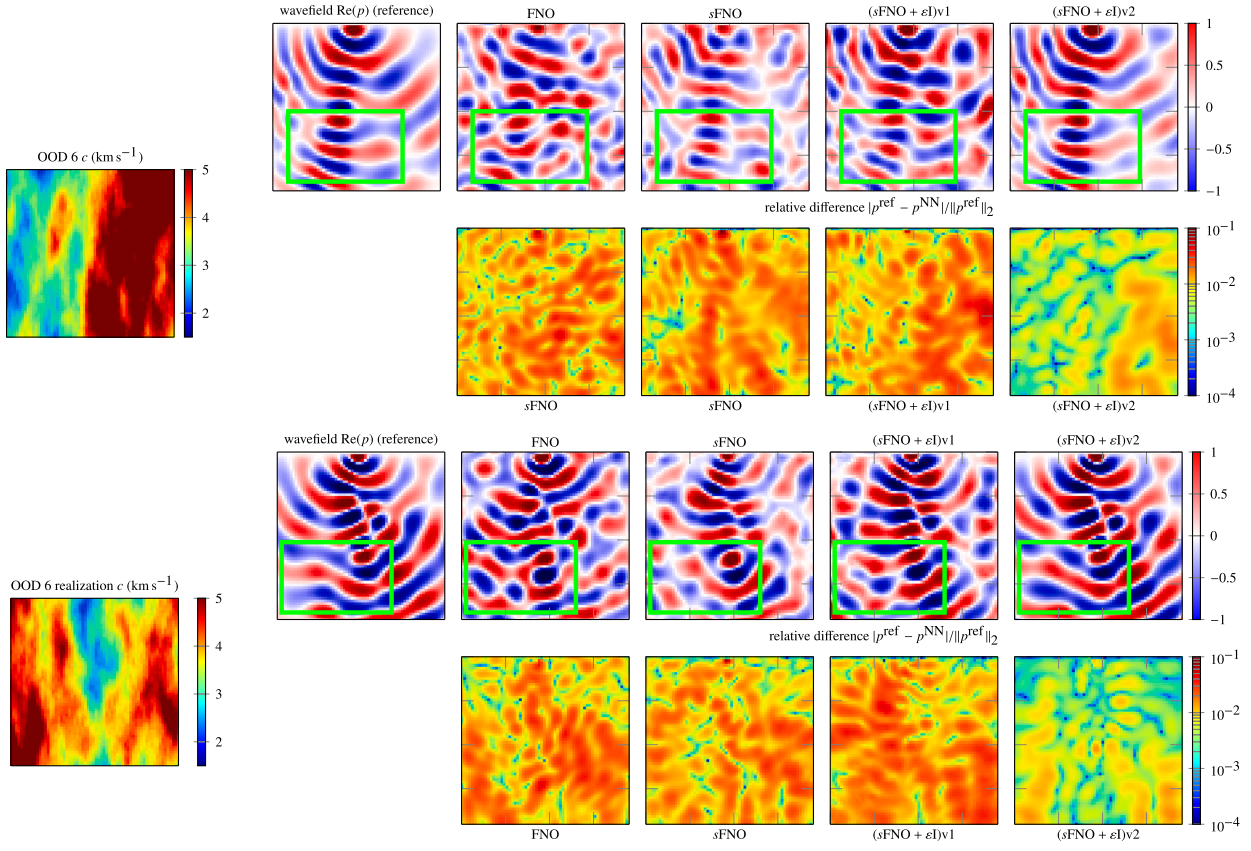


Fig. 12. OOD (family 6). Real part of the wave field of OOD family 6. Anisotropic case, see Table 5 and Appendix E.25. The green square positioned on the image serves as a visual aid to help identify and compare the differences in the reconstructed fields.

Table 13
Relative test loss of three networks tested with the probability defined by family 7.

OOD 7	FNO	sNO	(sFNO + εI)v1	(sFNO + εI) v2
model 1	0.3257	0.3037	0.2889	0.1814
model 2	0.3244	0.3207	0.2905	0.1748
model 3	0.3261	0.3024	0.2921	0.1845

Table 14
Relative test loss of three networks tested with the probability defined by family 8.

OOD 8	FNO	sNO	(sFNO + εI)v1	(sFNO + εI) v2
model 1	0.5508	0.4836	0.4621	0.2547
model 2	0.5527	0.5001	0.4706	0.2137
model 3	0.5547	0.4771	0.4604	0.2235

Table 15
Relative test loss of three networks tested with the probability defined by family 9.

OOD 9	FNO	sNO	(sFNO + εI)v1	(sFNO + εI) v2
model 1	0.9328	0.7053	0.6249	0.4419
model 2	0.9209	0.8811	0.7141	0.3303
model 3	0.8794	0.7270	0.6231	0.3167

Table 16
Relative test loss of three networks tested with the probability defined by family 10.

OOD 10	FNO	*s*NO	(*s*FNO + εI)v1	(*s*FNO + εI) v2
model 1	1.1806	0.8269	0.8217	0.5586
model 2	1.1391	0.9973	0.8727	0.4133
model 3	1.1049	0.8957	0.7825	0.4096

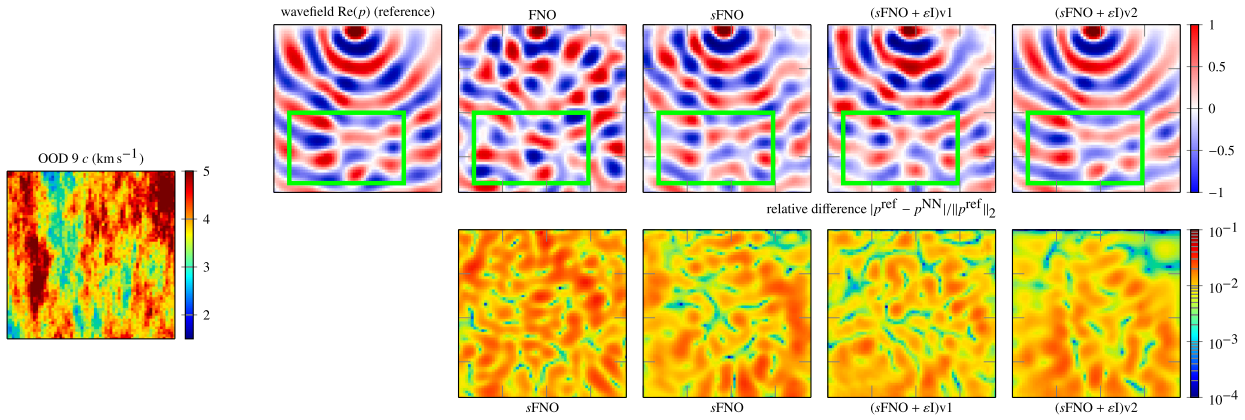


Fig. 13. OOD (family 9). Real part of the wave field of OOD family 9. Anisotropic case, with $\nu = 0.5$ see Table 12. The green square positioned on the image serves as a visual aid to help identify and compare the differences in the reconstructed fields.

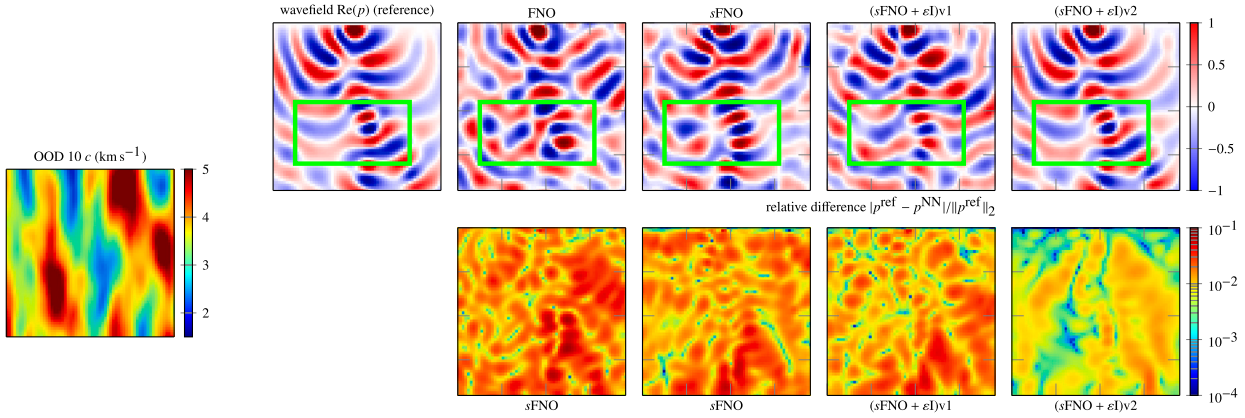


Fig. 14. OOD (family 10). Real part of the wave field of OOD family 10. Anisotropic case, with $\nu = 3.5$ see Table 12. The green square positioned on the image serves as a visual aid to help identify and compare the differences in the reconstructed fields.

families pose a greater challenge for the neural operators, as evidenced by the test loss values shown in Tables 15 to 16. In the figures, we highlight in green some of the main discrepancies between the predicted wave fields of the architectures, and the reconstruction by *numerical methods*. These discrepancies serve to illustrate the limitations and areas where the models may fall short in accurately reproducing the desired behavior. (See Figs. 13 and 14.)

Remark 5.1. We finally notice that the *s*FNO + εI network has promising results with respect to the BP 2004 [12] model. See Appendix E.3. It is worth noting that these findings go beyond the scope of the current theoretical framework described in Section 7.

6. Hyperneural operator as a surrogate model of the forward operator: $\mathcal{F}^f : (c, f) \mapsto \{\mathbf{p}(x_j, f)\}_{j=1,\dots,n_{\text{rcv}}}$

We propose a hyperneural operator as a surrogate model for the forward operator associated with the inverse boundary value problem for the Helmholtz equation, as discussed in Section 3.3. Our experiments are based on two key assumptions that persist throughout this work: (1) the sources are point sources, and (2) the output is a fixed-size vector, representing measurements of p at the receiver locations. These assumptions align with the typical practical considerations of seismic wave propagation in an acoustic medium. However, in Equation (19), we provide a potential relaxation of the first assumption to accommodate more general sources.

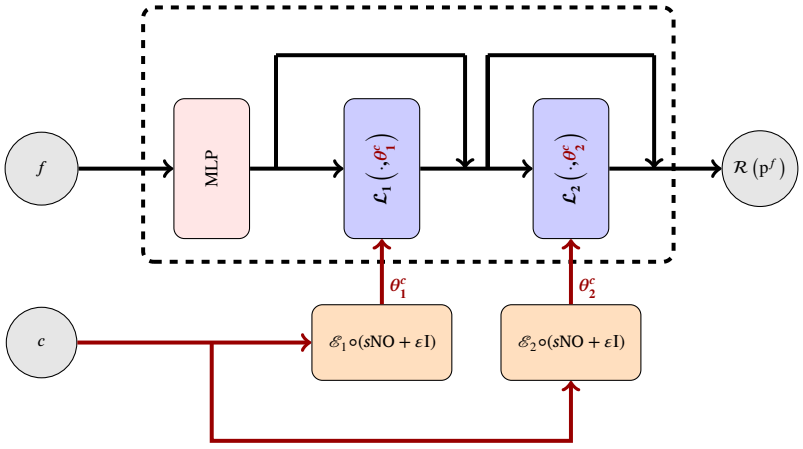


Fig. 15. Hypernetwork surrogate of the forward operator used in the experiments. We call the network inside the dashed rectangle a *metanetwork*, and the bottom network a *hyperneural operator*.

Notice that the direct application of a neural operator or any other derived architecture is difficult for the following reasons.

(a) The representation of the forward operator using neural networks faces challenges due to the distinct computational properties of point sources and wave speed. Wave speed can be discretized as a matrix $(\boldsymbol{x}, c(\boldsymbol{x}))$, while point sources are defined by their spatial position $\delta_{\boldsymbol{x}} \leftrightarrow \boldsymbol{x}$. To approximate the forward operator, a neural network must handle inputs of different natures (a point and a matrix) and generate an output with fixed discretization, based on the receiver positions.

In our experiments, the wave speed is discretized with a spacing of approximately 20 meters. However, point sources may not align precisely with the grid points of the wave speed field c. For example, the support of $\delta_{\boldsymbol{x}}$, where $\boldsymbol{x} = (x_1, x_2)$, may not necessarily be a multiple of 20. These discrepancies require careful consideration in designing the neural network architecture. The networks need to exhibit discretization invariance for both the wave speed parameter and the position of sources while ensuring that the output is discretized based on the receiver locations.

(b) From a theoretical perspective, the forward operator is as a map from a function space to a linear bounded operator (the data operator). By construction, neural operators only deal with maps from functions to functions, not from functions to operators (some interesting alternatives are proposed in [95] and [25]). See Beretta et al. [10, Sec 2.1] for the description of the forward operator in the time-harmonic case.

Given the previous difficulties, we proposed a hypernetwork solution, partially inspired by the empirical work of [120] and the theoretical results on hypernetworks of [2], subsequently improved in [38]. See Fig. 15.

Remark 6.1. Although our primary focus has been on the experimental implementation of the forward problem, we will consider the inverse problem in the future. Bayesian statistical approaches of the inverse problems such as Markov chain Monte Carlo [103] and ensemble Kalman filter [56,57] are commonly employed in inversion. However, these methods rely solely on the forward operators, but the computational challenge arises from multiple forward models. Our approach provides a **surrogate forward operator**, that, once trained, enables straightforward and efficient computation of multiple forward models. Therefore, we anticipate that by combining Bayesian statistical approaches with our method, we will be able to solve *Bayesian statistical inverse problems*.

Architecture The layers $\mathcal{L}_k$ for $k = 1, 2$ in the Fig. 15 are simple layers of Euclidean neural networks, that is $\mathcal{L}_k(\boldsymbol{x}) = \sigma \circ \left(W_k^c + b_k^c\right) \circ \boldsymbol{x}$. So that, $[W_k^c, b_k^c] = \mathscr{E}_k \circ \mathcal{G}(c)$, where $\mathscr{E}_k$ is an encoder sending the values of $\mathcal{G}(c)$ to a fixed parameter size, determining the capacity of the metanetwork (dashed rectangle from Fig. 15). A more general setting can be considered from the layers $\mathcal{L}$, depending on the nature of the sources (point-sources type or more general sources). However, given the simplicity of the Source-to-Receiver map, and the imposed discretization in the output, we can associate the point-source with its support $\boldsymbol{x} = (x_1, x_2)$ and the output is discretized by the number of receivers, corresponding to the columns of the of response matrix in Fig. 5. The main difficulty of the approximation is coming from the nonlinear dependency of the Helmholtz equation with respect to the wave speed.

We have the following association, if we call the metanetwork $\mathcal{NN}$ and the hyperneural operator $\mathcal{G}$, then for a point-source $\delta_{\boldsymbol{x}}^{\omega}$ we have

$$\mathcal{NN}(\delta_{\boldsymbol{x}}^{\omega}, \Theta(c)) = \mathcal{NN}(\delta_{\boldsymbol{x}}^{\omega}, \mathscr{E}^{\mathrm{hyper}} \circ \mathcal{G}(c)) \approx \mathcal{R}(\mathrm{p}^{\delta_{\boldsymbol{x}}, \omega}), \tag{17}$$

where $\mathcal{R}$ is a restriction operator which reduces the fields to the set of receivers positions Section 3.3, and $\Theta(c) = \mathscr{E}^{\mathrm{hyper}} \circ \mathcal{G}(c) \in \mathbb{C}^m$. In experiments, we see that a layer-wise form as Fig. 15 is more stable in the presence of the optimization algorithm. Similar conclusions were drawn in [120].

From Equation (17), it is evident that the point source is independent of the discretization used for c, and multiple sources can be implemented efficiently, by increasing the vector inputs, indicating the source's position. This means that the support of the point source can be finer than the discretization of c. On the other hand, the encoder, $\mathscr{E}^{\text{hyper}}$, is used in a similar manner as in *DeepOnet* described in Lanthaler et al. [71]. Its purpose is to map the range of $\mathcal{G}$ (the functional space) to a finite-dimensional space $\mathbb{C}^m$, which contains the parameters of the neural network $\mathcal{NN}$. Namely, $\mathscr{E}\circ\mathcal{G} : H \to \mathbb{C}^m$, where H represents the functional space where each realization of c lies. In our case, H can be identified with $H^{2\beta-d/2-\epsilon}(D)$, as the wave speed are realizations of the Whittle–Matérn field (see Section 3.4). Finally, the dimension m in $\mathbb{C}^m$ depends on the capacity chosen for the metanet family, $\mathcal{NN}$. In our experiments, we restrict it to a small two-layers network.

If f are not point-sources, $\mathcal{NN}$ can be expressed firstly by one global operator layer, followed by a second encoder $\mathscr{E}^{\text{meta}}$ as the output is always discrete given the position of the receivers. That is,

$$\mathcal{NN}(f,\Theta(c)) = \mathscr{E}^{\text{meta}}\circ\text{MLP}^{\theta_2}\circ\mathbf{IDFT}\left(G^{\theta_1}_{k,\ell}(\xi)\,\mathbf{DFT}(f)\right) \approx \mathcal{R}\left(\mathrm{p}^{f,\omega}\right), \tag{18}$$

and $\Theta(c) = [\theta_1,\theta_2] = \mathscr{E}^{\text{hyper}}\circ\mathcal{G}(c)$. Rather than (18), more general operator layers, neural operators, or DeepOnet networks can be used as a metanetwork. However, Equation (10) is linear with respect to f for a fixed c and ω.

The most general form of the $\mathcal{NN}$ is

$$\mathcal{NN}(f,\Theta(c)) = \mathscr{E}^{\text{meta}}\circ\mathcal{G}^{\text{meta}}(f,\mathscr{E}^{\text{hyper}}\circ\mathcal{G}(c)) \approx \mathcal{R}\left(\mathrm{p}^{f,\omega}\right) \tag{19}$$

for $\Theta(c) = \mathscr{E}^{\text{hyper}}\circ\mathcal{G}(c)$, and $\mathcal{NN} = \mathscr{E}^{\text{meta}}\circ\mathcal{G}^{\text{meta}}$, composition of an encoder sending the values to the position of the receiver, and an operator network $\mathcal{G}^{\text{meta}}$. Notice that $\mathscr{E}^{\text{meta}}$ is playing a similar role to the restriction operator $\mathcal{R}$. Moreover, $c \mapsto \mathscr{E}^{\text{meta}}\circ\mathcal{G}^{\text{meta}}(\cdot,\mathscr{E}^{\text{hyper}}\circ\mathcal{G}(c))$ can be realized as an observational operator.

"Prediction" of the "matrix" response for the forward operator The wave field reconstruction at the receiver position, by probing multiple point sources is presented in Fig. 16. The rows correspond to the multiple point sources, and the columns to the pressure field detected at multiple positions of the domain. In the top left side, we appreciate the wave speed, and bottom left side, the error of the approximation. The dataset of the experiment is described in Section 3.3, and the network is a special case of (19), exactly described in Fig. 15.

Details of the experiment We employ the *AdamW* optimizer [88] with an initial learning rate of 10^{-3}. We utilize a linear step scheduler with parameters: step size $= 40$, and a multiplicative factor of learning rate decay of $\gamma = 0.5$. The number of epochs is set to 100. In all architectures, we apply a small ℓ_2 weight regularizer with a parameter of 10^{-5}. Given that we already restricted the training to the empirical analysis of the architectures, the training process is conducted using $40,000$ out of $50,000$ generated samples, while $5,000$ samples are used for validation and $5,000$ for testing.

The relative L^2-error is 3×10^{-2}. In the implementation, $\mathscr{E}\circ\mathcal{G}(c) \in \mathbb{R}^{2m}$ and the complex-valued product is defined independently. Also, we did not split the learning rate from the metanetwork and hypernetwork, nor did we incorporate a more robust feature-extractor, as [120]. Provided that the sources are point sources, the complexity of the task is encoded in the high capacity of the neural operator defining the metanetwork. The latter is a residual network with 2 layers, and leaky ReLU activation.

7. Out-of-distribution under Gaussian sampling

While Section 5 presents the empirical out-of-distribution performance of our network design, specifically in the context of time-harmonic waves, we consider here an analysis of the out-of-distribution phenomenon under centered Gaussian measures for Banach spaces. We recall from Section 3.4 that the Whittle–Matérn field belongs to the spaces $H^{2\beta-d/2-\epsilon}(D)$ a.s. for all $\epsilon > 0$. The Whittle–Matérn field generates a centered Gaussian measure on the Hilbert spaces to which it belongs, see e.g. [23, Proposition 2.18]. This property holds under mild assumptions of the negative fractional power $L^{-2\beta}$, as described in [22]. Here, L represents the second-order elliptic operator presented in Equation (16).

We introduce the general framework for analyzing the out-of-distribution risk. (a) We defined the centered Gaussian measures on Banach spaces, (b) the Cameron-Martin spaces, and (c) the Wasserstein distance. Building upon these concepts and the powerful tools provided by Gaussian measures, we establish upper-bounds for the out-of-distribution risks associated with each of the architectures discussed in this paper. These bounds are expressed in terms of the Lipschitz norms of the neural operators. This theoretical foundation allows us to gain insights into the behavior of the neural operator family when confronted with data distributions that differ from the training distribution, as demonstrated in the experimental results in Section 5.

Remark 7.1. The main distinctions between the measures presented in this chapter and the ones discussed in Section 5 can be summarized as follows: (a) in Section 5, we apply an affine transformation T to the Whittle–Matérn field, $\mathrm{T}\circ Z$, in order to ensure non-negativity, which corresponds to a non-negative of the wave speed. (b) The measures introduced in this chapter exhibit a higher level of generality.

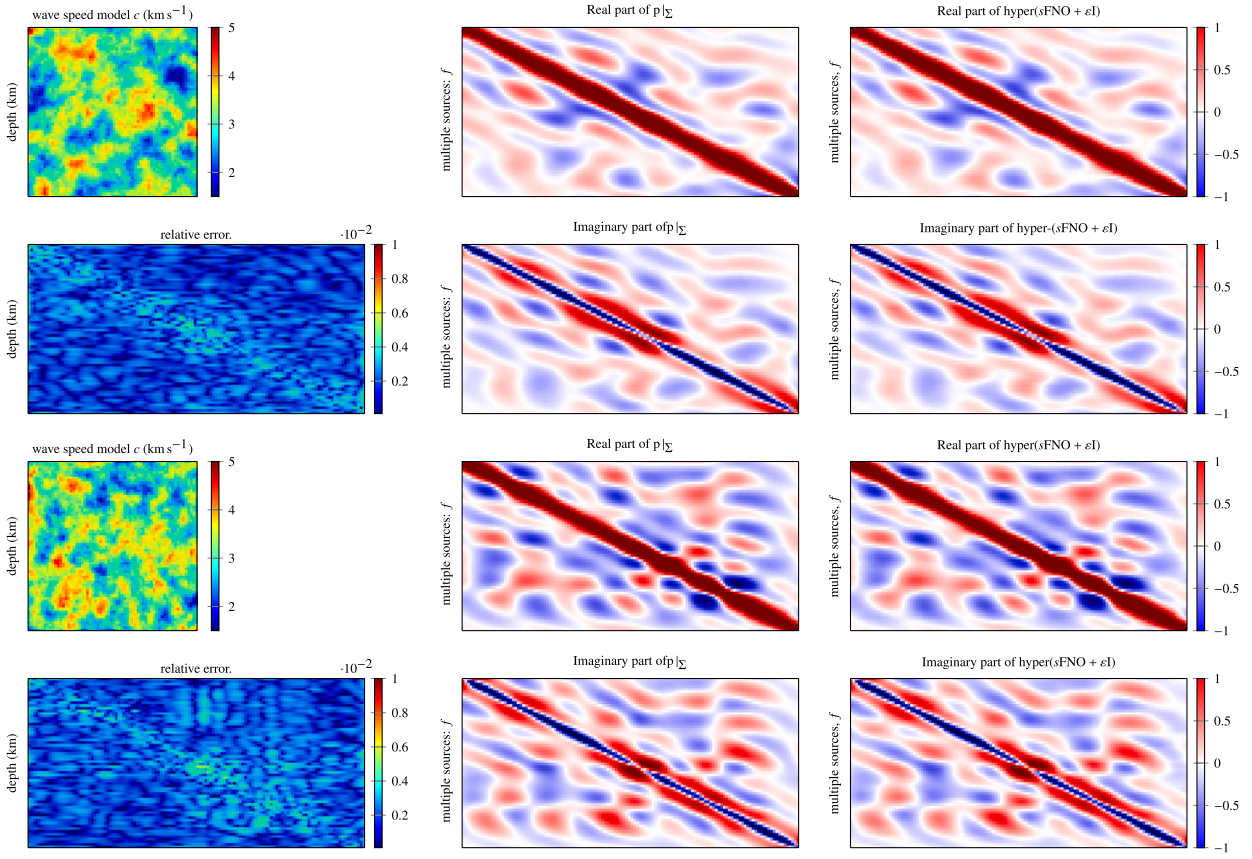


Fig. 16. Forward operator. Approximation of the forward operator by hyperneural operators.

7.1. Preliminaries

Our main theoretical results supporting the out-of-distribution performance of our neural operators require some background notions from optimal transport and the theory of Gaussian measures on Banach spaces, which we now review.

The order one Wasserstein distance In what follows, we make use of the *Wasserstein distance* of the order one between any two probability measures μ and ν, denoted by $\mathcal{W}_1(\mu,\nu)$. By the *Kantorovich–Rubinstein* duality, see [110, Theorem 5.10], $\mathcal{W}_1(\mu,\nu)$ has the form

$$\mathcal{W}_1(\mu,\nu) = \sup_{\substack{f\in\mathscr{F}\\ \|f\|_{\mathrm{Lip}}\le 1}} \mathbb{E}_{(a,u)\sim\mu}[f(a,u)] - \mathbb{E}_{(a,u)\sim\nu}[f(a,u)], \tag{20}$$

where $\mathscr{F}$ is a class of the Lipschitz continuous operators mapping from $\mathcal{X}\times\mathcal{Y}$ to $\mathbb{R}$, and $\|\cdot\|_{\mathrm{Lip}}$ is the Lipschitz norm[17] defined by

$$\|f\|_{\mathrm{Lip}} := \sup_{(a,u)} |f(a,u)| + \sup_{(a,u)\neq(b,v)} \frac{|f(a,u)-f(b,v)|}{\|(a,u)-(b,v)\|_{\mathcal{X}\times\mathcal{Y}}} \ge \mathrm{Lip}(f), \tag{21}$$

where $\mathrm{Lip}(f) := \sup_{(a,u)\neq(b,v)} \frac{|f(a,u)-f(b,v)|}{\|(a,u)-(b,v)\|_{\mathcal{X}\times\mathcal{Y}}}$.

Centered Gaussian measures on Banach spaces Let $\mathcal{X}$ be a separable Banach space of functions from D to $\mathbb{R}^{d_a}$ and $\mathcal{Y}$ be a separable Banach space of functions from D to $\mathbb{R}^{d_u}$. Recall that $\mathcal{X}\times\mathcal{Y}$ is also a separable Banach space, when normed by

$$\|(x,y)\|_{\mathcal{X}\times\mathcal{Y}} := (\|x\|_{\mathcal{X}}^2 + \|y\|_{\mathcal{Y}}^2)^{1/2}.$$

Let us briefly recall the definition of a Gaussian measure on a Banach space.

[17] $\|\cdot\|_{\mathrm{Lip}}$ is simply the $W^{1,\infty}$ norm. See Appendix A.1 and the reference therein.

Definition 7.1 *(Gaussian measure).* A measure $\mu_X \in \mathcal{P}_1(\mathcal{X})$ is said to be centered and Gaussian if, for every continuous linear functional $E \in \mathcal{X}^\star$ the measure $E_\# \mu_X$ is a zero-mean Gaussian on $\mathbb{R}$. The *weak variance* Σ of μ_X is defined to be

$$\Sigma = \sup_{E \in \mathcal{X}^\star, \|E\| \leq 1} \mathbb{E}_{a \sim \mu_X}[E^2(a)]^{1/2}.$$

Associated to every centered Gaussian measure, we may define a *small ball function* $\psi : (0, \infty) \to \mathbb{R}$ as

$$\psi(\eta) := -\log\left(\mu_X(B(0,\eta))\right),$$

for every $\eta > 0$. There is a reproducing kernel Hilbert space $\mathcal{H}_\mu$ naturally associated to μ_X which is the completion of the range of the map $S : \mathcal{X}^\star \to \mathcal{X}$ sending any $E \in \mathcal{X}^\star$ to the Bochner integral[18] $S(E) := \int_{a \in \mathcal{X}} E(a) \cdot a \, \mu_X(da)$ with respect to the inner product $\langle \cdot, \cdot \rangle_{\mu_X}$, defined for any $E, F \in \mathcal{X}^\star$ by $\langle S(E), S(F) \rangle_\mu := \int_{a \in \mathcal{X}} E(a) F(a) \, \mu_X(da)$. We denote the induced norm on $\mathcal{H}_{\mu_X}$ by $\| \cdot \|_{\mathcal{H}_{\mu_X}}$; which is induced by an inner product, see [67]. In fact, $\mathcal{H}_{\mu_X}$ is a reproducing kernel Hilbert space with a relatively compact unit ball, called the *Cameron-Martin space* associated with the centered Gaussian measure μ_X. In fact, the Cameron-Martin space $\mathcal{H}_{\mu_X}$ characterizes μ_X, see [76, Chapter 8] for details (we will briefly review the Cameron-Martin space in Section A.6).
Since the closed unit ball $\overline{B_{\mathcal{H}_{\mu_X}}(0,1)}$ of $\mathcal{H}_{\mu_X}$ is compact (see[19] [76, Lemma 8.4]), then its *metric entropy* $H_{\mu_X}(\varepsilon) := \log(N_{\mu_X}(\varepsilon))$ are finite; where $N_{\mu_X}(\varepsilon) := \min\{n \in \mathbb{N}_+ : \exists x_1, \ldots, x_n \in \overline{B_{\mathcal{H}_{\mu_X}}(0,1)} \text{ s.t. } \forall x \in \overline{B_{\mathcal{H}_{\mu_X}}(0,1)} \, \exists i \in [n] \text{ s.t. } \|x - x_i\|_{\mathcal{H}_{\mu_X}} < \varepsilon\}$ is the *covering number*[20] of $\overline{B_{\mathcal{H}_{\mu_X}}(0,1)}$. The key connection between the *small ball function* ψ, a probabilistic notion, and entropy numbers, a constructive approximation theoretic tool, is that estimates on the growth of one imply estimates on the growth of the other.

7.2. Out-of-distributional generalization

Consider an "unknown" $L^\star$-Lipschitz (non-linear forward) operator $\mathcal{G}^\star : \mathcal{X} \to \mathcal{Y}$, a sampling distribution $\mu_X \in \mathcal{P}_1(\mathcal{X})$, that is $\mathbb{E}_{X \sim \mu_X}[\|X\|_{\mathcal{X}}] < \infty$, and a sequence of i.i.d. samples $(a_n)_{n=1}^\infty$ defined on a common measurable space $(\Omega, \mathcal{A})$, where a_1 has law μ_X and where $L^\star \geq 0$. We also consider an *out-of-distributional* sampling measure $\tilde{\mu}_X$ in $\mathcal{P}_1(\mathcal{X})$. We consider a *common irreducible measurement noises* ϵ taking values in $\mathcal{Y}$, and quantifying hardware and sampling limitations, defined on $(\Omega, \mathcal{A})$ and independent from $\{a_n\}_{n=1}^\infty$ with law $\mu_\epsilon \in \mathcal{P}_1(\mathcal{Y})$.

The data-generating and out-of-distribution, laws defined $\mu_{OOD:X}$ and μ are respectively defined by

$$\mu := (\mathbf{I_d} \times \mathcal{G}^\star)_\# \mu_X \star \mu_\epsilon \text{ and } \mu_{OOD} := (\mathbf{I_d} \times \mathcal{G}^\star)_\# \mu_{OOD:X} \star \mu_\epsilon, \tag{22}$$

where $\star$ is the convolution operation and $\mathbf{I_d}$ is the identity map on $\mathcal{X}$. The out-of-distributional measure μ_{OOD} and data-generating measures μ are *coupled* via the following condition: there is a $\varepsilon \geq 0$ such that

$$\mathcal{W}_1(\mu_{OOD}, \mu) \leq \varepsilon. \tag{Coupling}$$

Remark 7.2. Intuitively, (22) states that one considers out-of-distributional shifts which arise as perturbations to the sampling mechanism (distribution) μ_X on $\mathcal{X}$. The coupling condition (Coupling) then quantifies the ε-magnitude of these perturbations; note that, in principle, $\varepsilon > 0$ can be large or small.

Example 7.1 *(Interpretation of Coupling Condition).* Let X and N be a random variables on $\mathcal{X}$ with laws μ_X and ν with finite means $\mathbb{E}_{\mu_X}[\|X\|_{\mathcal{X}}], \mathbb{E}_\nu[\|N\|_{\mathcal{X}}] < \infty$, and denote $\varepsilon_s := \mathbb{E}_{N \sim \nu}[\|N\|_{\mathcal{X}}]$. The random variable $X + N$ represents a sample X corrupted by "sampling noise" N. The law of $X + N$ is $\mu_{OOD:X} := \mu_X \star \nu$. Furthermore, one has

$$\mathcal{W}_1(\mu_X, \mu_{OOD:X}) \leq \mathbb{E}\left[\|X - (X+N)\|_{\mathcal{X}}\right] = \varepsilon_s. \tag{23}$$

The Kantorovich-Rubinstein duality implies that the push-forward map $(\mathbf{I_d} \times \mathcal{G}^\star)_\# : \mathcal{P}_1(\mathcal{X}) \to \mathcal{P}_1(\mathcal{X} \times \mathcal{Y})$ is at-most $2\max\{1, L\}$-Lipschitz; whence, (23) implies that

$$\mathcal{W}_1\left((\mathbf{I_d} \times \mathcal{G}^\star)_\# \mu_X, (\mathbf{I_d} \times \mathcal{G}^\star)_\# \mu_{OOD:X}\right) \leq 2\max\{1, L\}\varepsilon_s. \tag{24}$$

Let E be a random variable on $\mathcal{Y}$ satisfying $\varepsilon_m := \mathbb{E}_{E \sim \mu_\epsilon}[\|E\|_Y] < \infty$, quantifying "measurement noise". Let μ_ϵ be the law of the random variable $(0, E)$ where 0 is the zero-vector on $\mathcal{X}$. Then $(X + N, \mathcal{G}^\star(X+N) + E)$ quantifies a noisy training pair with sampling and measurement noise, whose law is μ, $(X, \mathcal{G}^\star(X) + E)$ quantifies a training sample only corrupted by measurement noise, whose law is μ_{OOD}, and both laws are related by

$$\mathcal{W}_1(\mu, \mu_{OOD}) \leq \varepsilon_m + \mathcal{W}_1\left((\mathbf{I_d} \times \mathcal{G}^\star)_\# \mu_X, (\mathbf{I_d} \times \mathcal{G}^\star)_\# \mu_{OOD:X}\right) \leq 2\varepsilon_m + 2\max\{1, L\}\varepsilon_s =: \varepsilon,$$

[18] Cf. Appendix A.4.

[19] And the remark following its proof at the bottom of page 209.

[20] See [102, Chapter 27] for details in the context of learning theory or [20] in the context of approximation theory.

where the right-hand side was obtain by (24) together with a similar computation to (23).

A key advantage of coupling μ and μ_{OOD} using the $\mathcal{W}_1$ distance, over other notions, esp. f-divergences, is that the data-generating and out-of-distribution laws can be *mutually singular*[21] but still remain comparable; this is, of course, not possible with classical divergences.

When training input-output pairs are generated by sampling μ, by which we mean that we have access to the following (random) *empirical* measure

$$\mu^N := (\mathbf{I_d} \times \mathcal{G}^\star)_{\#} \mu_X^N \star \mu_\epsilon, \tag{25}$$

where the empirical (random) probability measure μ_X^N is defined by $\mu_X^N = \frac{1}{N} \sum_{n=1}^N \delta_{a_n}$.

We now state our main out-of-distribution bound, which operates under the following conditions.

Assumption 7.2 *(Regularity of the Cameron-Martin space).* Suppose further that μ_X is a center Gaussian measure on $\mathcal{X}$ with weak variance Σ and that the small ball function ψ satisfies:

(i) There exists a constant $c > 0$ such that $\psi(\eta) \leq c\,\psi(2\eta)$ for every η small enough,[22]
(ii) For every $\alpha > 0$ and each positive integer N, $N^{-\alpha} = o\big(\psi^{-1}(\log(N))\big)$.

Remark 7.3. Assumption 7.2 requires that Gaussian "*sampling*" measure μ_X defined on the input space $\mathcal{X}$ does not place mass too far away from the origin; i.e. that it is sufficiently well concentrated, and the function ψ quantifies how well this measure is concentrated.

Neural operator class Let us define the family of standard Neural Operator as follows:

$$\begin{aligned}\mathscr{N} = \Big\{ & \mathcal{G}_\theta : \mathrm{L}^2(D;\mathbb{R}^{d_a}) \to \mathrm{L}^2(D;\mathbb{R}^{d_u}) : \ \mathcal{G}_\theta = (W_L + \boldsymbol{\mathcal{K}}_L) \circ \sigma(W_{L-1} + \boldsymbol{\mathcal{K}}_{L-1}) \circ \cdots \circ \sigma(W_0 + \boldsymbol{\mathcal{K}}_0) \\ & \theta = (W_\ell, \boldsymbol{\mathcal{K}}_\ell)_{\ell=0,\dots,L},\ W_\ell \in \mathbb{R}^{d_{\ell+1} \times d_\ell},\ \boldsymbol{\mathcal{K}}_\ell : \mathrm{L}^2(D;\mathbb{R}^{d_\ell}) \to \mathrm{L}^2(D;\mathbb{R}^{d_{\ell+1}}),\ \text{and } d_0 = d_a,\ d_{L+1} = d_u \Big\}.\end{aligned} \tag{26}$$

$\sigma : \mathbb{R} \to \mathbb{R}$ is an element-wise nonlinear map, and $\boldsymbol{\mathcal{K}}_\ell$ are linear integral operators with kernel function, $k_\ell : D \times D \to \mathbb{R}^{d_{\ell+1} \times d_\ell}$, i.e., $x \mapsto \big(\boldsymbol{\mathcal{K}}_\ell u\big)(x) := \int_D k_\ell(x,y) u(y)\, dy$ and $u \in \mathrm{L}^2(D;\mathbb{R}^{d_\ell})$. We shall write, $w_{\ell,ij} = (W_\ell)_{i,j} \in \mathbb{R}$ and $k_{\ell,ij} = (k_\ell)_{i,j} : D \times D \to \mathbb{R}$ as (i,j)-element of W_ℓ and k_ℓ, respectively.

Assumption 7.3. There exist positive constants C_w, C_k, C_d, C_a, C_σ, and C_β such that

(i). $\|W_\ell\|_{\mathrm{op}} \leq C_w$, and $d_\ell \leq C_d$ for all $\ell = 0, \dots, L$, where $\|\cdot\|_{\mathrm{op}}$ is the operator norm.
(ii). $\|\boldsymbol{\mathcal{K}}_\ell\|_{\mathrm{L}^2,\mathrm{F}} := \left(\sum_{i,j} \left\| k_{\ell,ij} \right\|^2_{\mathrm{L}^2(D\times D)} \right)^{1/2} \leq C_k$ for all $\ell = 0, \dots, L$, where $|D| = \int \mathbf{1}_D\, d\lambda$,[23] and $k_\ell : D \times D \to \mathbb{R}^{d_{\ell+1} \times d_\ell}$ is the kernel function.
(iii). $\|a\|_{\mathrm{L}^2(D;\mathbb{R}^{d_a})} \leq C_a$ for all $a \in \mathrm{supp}(\mu_a)$.
(iv). σ is C_σ-Lipschitz, i.e., $|\sigma(s) - \sigma(t)| \leq C_\sigma |s - t|$ for $s, t \in \mathbb{R}$.
(v). $\sup_{x,y \in D} |k_{\ell,ij}(x,y)| \leq C_\alpha$ for $\ell = 0, \dots, L$, $i = 1, .., d_\ell$, and $j = 1, \dots, d_{\ell+1}$.
(vi). $k_{\ell,ij} : D \times D \to \mathbb{R}$ is C_β-Lipschitz, see Definition A.4, for $\ell = 0, \dots, L$, $i = 1, .., d_\ell$, and $j = 1, \dots, d_{\ell+1}$.

Sequential neural operator class We define the family, see Section 2, as

$$\begin{aligned}\widetilde{\mathscr{N}} = \Big\{ & \mathcal{G}_\theta : \mathrm{L}^2(D;\mathbb{R}^{d_a}) \to \mathrm{L}^2(D;\mathbb{R}^{d_u}) : \\ & \mathcal{G}_\theta = (\mathbf{Z}_L \mathbf{I_d} + \mathbf{X}_L f_L) \circ (\mathbf{Z}_L \mathbf{I_d} + \mathbf{X}_L \sigma \circ \boldsymbol{\mathcal{K}}_L) \circ \cdots \circ (\mathbf{Z}_0 \mathbf{I_d} + \mathbf{X}_0 f_0) \circ (\mathbf{Z}_0 \mathbf{I_d} + \mathbf{X}_0 \sigma \circ \boldsymbol{\mathcal{K}}_0) \\ & \mathbf{Z}_\ell, \mathbf{X}_\ell \in \{0,1\},\ f_\ell = W_{\ell,M} \circ \sigma(W_{\ell,M-1}) \circ \cdots \circ \sigma(W_{\ell,0}) \text{ is an Mth layer MLP} \\ & \theta = (W_{\ell,m}, \boldsymbol{\mathcal{K}}_\ell)_{\substack{\ell=0,\dots,L \\ m=0,\dots,M}},\ W_{\ell,m} \in \mathbb{R}^{d^w_{\ell,m+1} \times d^w_{\ell,m}} \text{ and } \boldsymbol{\mathcal{K}}_\ell : \mathrm{L}^2(D;\mathbb{R}^{d^k_\ell}) \to \mathrm{L}^2(D;\mathbb{R}^{d^k_{\ell+1}}) \\ & d^w_{\ell,0} = d^k_{\ell+1},\ d^w_{\ell,M} = d^k_{\ell+1},\ d^k_0 = d_a,\ d^k_{L+1} = d_u \Big\}.\end{aligned} \tag{27}$$

[21] For example, if ν is the standard Gaussian measure on $\mathbb{R}$ then any finitely supported measure $\sum_{n=1}^N w_n \delta_{x_n}$ is singular with respect to ν and vice versa.
[22] I.e.: There exists some $\eta_0 > 0$ such that (i) holds whenever $0 < \eta \leq \eta_0$.
[23] λ is the Lebesgue measure.

Assumption 7.4. There exist positive constants C_w, C_k, C_d, C_a, C_σ, and C_β such that

(i). $\|W_{\ell,m}\|_{\text{op}} \le C_w$, and $d^k_\ell, d^w_{\ell,m} \le C_d$, for $\ell = 0, ..., L$, $m = 0, ..., M$.
(ii). $\|\mathcal{K}_\ell\|_{\text{L}^2,\text{F}} \le C_k$, for $\ell = 0, ..., L$.
(iii). $\|a\|_{\text{L}^2(D;\mathbb{R}^{d_a})} \le C_a$, for $a \in \text{supp}(\mu_a)$.
(iv). σ is C_σ-Lipschitz, i.e., $|\sigma(s) - \sigma(t)| \le C_\sigma |s - t|$ for $s, t \in \mathbb{R}$.
(v). $\sup_{x,y\in D} |k_{\ell,ij}(x, y)| \le C_\alpha$ for $\ell = 0, ..., L$, $i = 1, .., d_\ell$, and $j = 1, ..., d^k_{\ell+1}$.
(vi). $k_{\ell,ij} : D \times D \to \mathbb{R}$ is C_β-Lipschitz, for $\ell = 0, ..., L$, $i = 1, .., d^k_\ell$, and $j = 1, ..., d^k_{\ell+1}$.

Remark 7.4. Assumptions, 7.3-7.4, restrict the capacity of our class of neural operators. Without such a capacity restriction, which the user can freely adjust, the corresponding class would exhibit unbounded Rademacher complexity, as described in Section 8 and thus, generalization is not guaranteed.

Lipschitz bounds We have to estimate the Lipschitz norms for $\mathcal{N}$ and $\widetilde{\mathcal{N}}$, corresponding to standard NO and $s\text{NO} + \varepsilon\text{I}$, respectively.

Lemma 7.5 *(Lipschitz stability of the hypothesis classes ($\mathcal{N}$ and $\widetilde{\mathcal{N}}$)). (i) Let Assumption 7.3 hold. Then, we have that*

$$\|\mathcal{G}\|_{\text{Lip}} \le (C_w + C_k)^{L+1} C_\sigma^L, \ \mathcal{G} \in \mathcal{N}.$$

(ii) Let Assumption 7.4 hold. Then, we have that

$$\|\mathcal{G}\|_{\text{Lip}} \le \left[\prod_{\ell=0}^{L} (\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma) \right], \ \mathcal{G} \in \widetilde{\mathcal{N}}.$$

The proof is given by the same argument in the proofs of Corollaries 8.6 and 8.7.

Theorem 7.6 *(Out-of-distributional generalization bounds for the NO and $s\text{NO} + \varepsilon\text{I}$v2 hypothesis classes). Suppose that either of Assumption 7.3 or Assumption 7.4, that the small ball function ψ satisfies Assumption 7.2, and that there is an $\varepsilon \ge 0$ such that the coupling condition (Coupling) holds. Then there exists a constant C_μ, depending only on μ_X, such that: for every $0 < \delta \le 1$*

$$\sup_{\mathcal{G}\in\mathscr{G}} \mathbb{E}_{(a,u)\sim\mu_{OOD}} \left[\ell(\mathcal{G}(a), u)\right] - \bar{L}\, \mathbb{E}_{(a,u)\sim\mu^N} \left[\ell(\mathcal{G}(a), u)\right] \le \bar{L} \left(\varepsilon + C_\mu \psi^{-1}\big(\log(N)\big) + \frac{\Sigma\sqrt{-2\log(\delta)}}{\sqrt{N}} \right), \tag{28}$$

holds with probability at-least $1 - \delta$; where $\bar{L} := L_\ell \max\{1, L^\star\} \max\{1, L_{\mathscr{G}}\}$; where $L_{\mathscr{G}} \ge 0$ depends on which if Assumption 7.3 or Assumption 7.4 hold, and is respectively given by:

(i) If Assumption 7.3 holds and $\mathscr{G} = \mathcal{N}$ (defined in (26)), Lemma 7.5 implies:

$$L_{\mathcal{N}} \le (C_w + C_k)^{L+1} C_\sigma^L, \tag{29}$$

(ii) If Assumption 7.4 holds and $\mathscr{G} = \widetilde{\mathcal{N}}$ (defined in (27)), Lemma 7.5 implies:

$$L_{\widetilde{\mathcal{N}}} \le \left[\prod_{\ell=0}^{L} (\mathbf{Z}_L + \mathbf{X}_L C_w^{M+1} C_\sigma^M)(\mathbf{Z}_L + \mathbf{X}_L C_k C_\sigma) \right], \tag{30}$$

Furthermore, if the metric entropy H_μ of the unit ball in the Cameron-Martin space associated with the sampling measure μ_X satisfies $H_\mu(r) \in \Theta\big(\frac{\log(1/r)^{2\beta/(2+\alpha)}}{r^{2\alpha/(2+\alpha)}}\big)$ then the right-hand side of (28)

$$\sup_{\mathcal{G}\in\mathscr{G}} \mathbb{E}_{(a,u)\sim\mu_{OOD}} \left[\ell(\mathcal{G}(a), u)\right] - \bar{L}\, \mathbb{E}_{(a,u)\sim\mu^N} \left[\ell(\mathcal{G}(a), u)\right] \le \bar{L} \left(\varepsilon + C_\mu C\, \Psi\big(\log(N)\big) + \frac{\Sigma\sqrt{-2\log(\delta)}}{\sqrt{N}} \right), \tag{31}$$

where Ψ is the inverse[24] of the map $\eta \mapsto \frac{\log(1/\eta)^\beta}{\eta^\alpha}$ and $C > 0$ is an absolute constant.

The case of $\mathscr{G} = \mathcal{N}$ corresponds to the family of standard Neural Operators, while the case of $\mathscr{G} = \widetilde{\mathcal{N}}$ corresponds to the family of proposed Neural Operator $s\text{NO} + \varepsilon\text{I}$. $L_{\mathscr{G}}$ represents the upper bound of Lipschitz norms for hypothesis classes $\mathscr{G}$, and $L_{\mathcal{N}}$ and $L_{\widetilde{\mathcal{N}}}$ are estimated by Lemma 7.5. Analogous observations can be made as in Remark 8.2 regarding the upper bound of Lipschitz norms. Specifically, if $(C_w + C_k)C_\sigma > 1$, then the upper bound in (29) diverges with depth L. On the other hands, if $\mathbf{Z}_\ell = 1$ and

[24] For example, if $\beta = \alpha = 1$ then $\Psi(\eta) = W(\eta)/\eta$, where W is the Lambert W function.

Table 17
Rates for Different Sampling Measures and Banach Spaces.

Space	Covariance function	Entropy estimate	Small ball asymptotics ($\psi(\eta)$)
$L^2([0,1]^2)$	$\min\{s_1,t_1\}\min\{s_2,t_2\}$	-	$\Theta\left(\frac{\log(1/\eta^2)^2}{\eta^2}\right)$
$C([0,1]^d)$	$\frac{\alpha}{2^d}\prod_{i=1}^d s_i^{h_i}+t_i^{h_i}-\lvert s_i-t_i\rvert^{h_i}$	-	$\Theta\left(\frac{1}{\eta^{2/h}}\right)$
General	General	$\Theta\left(\frac{\log(1/r)^{2\beta/(2+\alpha)}}{r^{2\alpha/(2+\alpha)}}\right)$	$\Theta\left(\frac{\log(1/\eta)^{\beta}}{\eta^{\alpha}}\right)$

The "entropy estimates" the required condition on the *metric entropy* of the unit ball in the Cameron-Martin space associated to the centered Gaussian "sampling" measure μ_X. Here $h := \min_{i=1,\dots,d} h_i$ is the minimal "regularity" of the Brownian sheet of $C([0,1]^d)$ in all directions.

$\mathbf{X}_\ell$ follows a Bernoulli distribution (which corresponds to (sNO + εIv2) with an appropriate choice of p_ℓ, then upper bound in (30) remain bounded as $L \to \infty$.

We now show that the conditions of Theorem 7.6, namely the regularity of the Cameron-Martin space associated with the data-generating measure μ are easily satisfied. We consider two examples, one of a Brownian sheet and a fractional Brownian sheet on different hypercubes with respect to different norms on their associated function spaces.

Table 17 reports the rates implied by Theorem 7.6 in the case of a Brownian sheet on $[0,1]^2$ and $[0,1]^d$ with respect to the L^2 and uniform norms. More generally, we report the rates implied by the result for centered Gaussian measures μ a general Banach space, when we have access to tight asymptotics on the covering number of the unit ball in the *Cameron-Martin*[25] RHKS associated to μ.

Lemma 7.7 *(Estimates on small ball functions for Gaussian sheets in uniform topology [94, Theorem 2.1]). Let $D = [0,1]^d$ for a positive integer d, fix "Hurst parameters" $0 < h_1, \dots, h_d < 2$, a parameter $0 < \alpha < 2$, and let μ be the continuous centered Gaussian measure on the Banach space $C_0([0,1]^d, \mathbb{R})$ equipped with the supremum norm and with covariance function*

$$\mathbb{E}\left[X_{s_1,\dots,s_d} X_{t_1,\dots,t_d}\right] = \frac{\alpha}{2^d}\prod_{i=1}^d s_i^{h_i} + t_i^{h_i} - |s_i - t_i|^{h_i}.$$

Then, Assumptions (i)-(ii) on the small ball function ψ, in Lemma 7.10, hold and there exists a constant $0 < C_1 \le C_2$, depending only on d, a, and on α, such that

$$C_1 \frac{1}{\eta^{2/h}} \le \psi(\eta) \le C_2 \frac{1}{\eta^{2/h}}$$

for η small enough,[26] *where $h := \min_{i=1,\dots,d} h_i$.*

Example 7.2 *(Estimate on the standard Brownian sheet on $[0,1]^2$ [68, Equation (5.37)]).* Let $1 \le p \le 2$. Let $D = [0,1]^2$ and consider the centered continuous Gaussian process $X := (X_{s,t})_{0\le s,t\le 1}$ in $L^2(D)$ with covariance function

$$\mathbb{E}[X_{s_1,t_1} X_{s_2,t_2}] = \min\{s_1,t_1\}\min\{s_2,t_2\}.$$

Then, Assumptions (i)-(ii) on the small ball function ψ, in Lemma 7.10, hold and there exists a constant $0 < C_1 \le C_2$ such that

$$C_1 \frac{\log(1/\eta)^2}{\eta^2} \le \psi(\eta) \le C_2 \frac{\log(1/\eta^2)^2}{\eta^2}.$$

We now derive Theorem 7.6 via a sequence of lemmata.

7.3. Proof of Theorem 7.6

The proof of Theorem 7.6 extends the transport-theoretic approach to deriving generalization bounds of [53], to the infinite-dimensional setting, by incorporating elements of the geometry of Cameron-Martin spaces. We begin, with the following "change-of-measure lemma" which bounds the gap between the risks from data samples from any two distinct, arbitrary, probability measures μ and ν in $\mathcal{P}_1(\mathcal{X}\times\mathcal{Y})$.

Lemma 7.8 *(Change of distribution). Consider an in-distribution measure μ and out-of-distribution probability measure $\mathbb{Q}$, where $\mu, \mathbb{Q}, \nu \in \mathcal{P}_1(\mathcal{X}\times\mathcal{Y})$. Let $\mathcal{G}$ be a family of L-Lipschitz functions from $\mathcal{X}$ to $\mathcal{Y}$, and suppose that*

$$\mathcal{W}_1(\mathbb{Q}, \mu) \le \varepsilon,$$

[25] See [76, Chapter 8] for details.
[26] That is, there is some $\eta_0 > 0$ such that the condition holds for every $0 < \eta \le \eta_0$.

for some $\varepsilon > 0$. For any L_ℓ-Lipschitz loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to [0, \infty)$ we have

$$\mathbb{E}_{(a,u)\sim\mathbb{Q}}\left[\ell(\mathcal{G}(a),u)\right] \leq \mathrm{L}_\ell \max\{1, L\}\left(\varepsilon + \mathcal{W}_1(\nu,\mu) + \mathbb{E}_{(a,u)\sim\nu}\left[\ell(\mathcal{G}(a),u)\right]\right),$$

for each $\mathcal{G} \in \mathscr{G}$.

Proof. See Appendix B.1. □

Next, we incorporate the structure of μ and μ^N into Lemma 7.8, in place of the arbitrary measures μ and ν, respectively.

Lemma 7.9 *(Structured change-of-measure). Assume that μ and μ^N are respectively given by (22) and (25). Then, we have that*

$$\mathcal{W}_1\left(\mu, \mu^N\right) \leq \max\{1, L^\star\}\, \mathcal{W}_1(\mu_X, \mu_X^N).$$

Proof. See Appendix B.2. □

We will assume that our samples of $\mathcal{G}^\star$, distributed according to μ_X, are drafted from a Gaussian measure on $\mathcal{X}$.

Lemma 7.10 *(General concentration inequality for Lipschitz hypotheses). Assume the setting of Lemma 7.9 and fix a positive integer N. Suppose further that μ_X is a center Gaussian measure on $\mathcal{X}$ with weak variance Σ and that the small ball function ψ satisfies:*

(i) There exists a constant $c > 0$ such that $\psi(\eta) \leq c\,\psi(2\eta)$ for every η small enough,[27]
(ii) For every $\alpha > 0$ and each positive integer N, $N^{-\alpha} = o\big(\psi^{-1}(\log(N))\big)$.

There exists a constant C_μ, depending only on μ_X, such that: for every $0 < \delta \leq 1$

$$\sup_{\mathcal{G}\in\mathscr{G}} \mathbb{E}_{(a,u)\sim\mathbb{Q}}\left[\ell(\mathcal{G}(a),u)\right] - \bar{L}\,\mathbb{E}_{(a,u)\sim\mu^N}\left[\ell(\mathcal{G}(a),u)\right] \leq \bar{L}\left(\varepsilon + C_\mu \psi^{-1}\big(\log(N)\big) + \frac{\Sigma\sqrt{-2\log(\delta)}}{\sqrt{N}}\right), \tag{32}$$

holds with probability at-least $1-\delta$; where $\bar{L} := L_\ell \max\{1, L\}\max\{1, L^\star\}$.
Furthermore, suppose that $H_\mu(r) \in \Theta\big(\frac{\log(1/r)^{2\beta/(2+\alpha)}}{r^{2\alpha/(2+\alpha)}}\big)$ then the right-hand side of (28)

$$\sup_{\mathcal{G}\in\mathscr{G}} \mathbb{E}_{(a,u)\sim\mathbb{Q}}\left[\ell(\mathcal{G}(a),u)\right] - \bar{L}\,\mathbb{E}_{(a,u)\sim\mu^N}\left[\ell(\mathcal{G}(a),u)\right] \leq \bar{L}\left(\varepsilon + C_\mu \tilde{\psi}^{-1}\big(\log(N)\big) + \frac{\Sigma\sqrt{-2\log(\delta)}}{\sqrt{N}}\right), \tag{33}$$

where $\tilde{\psi}(\eta) = C\frac{\log(1/\eta)^\beta}{\eta^\alpha}$ and $C > 0$ is an absolute constant.

Remark 7.5. As remarked on [14, page 542], condition (ii) in Lemma 7.10 implies that the centered Gaussian measure μ is not supported on a finite-dimensional Banach subspace of $\mathcal{X} \times \mathcal{Y}$.

Proof of Lemma 7.10. See Appendix B.3. □

Applying Lemma 7.10 to the hypothesis classes $\mathscr{N}$ and $\widetilde{\mathscr{N}}$, defined in (26) and (27), respectively, yields our main generalization bound for out-of-sample distribution learning; i.e., Theorem 7.6.

Proof of Theorem 7.6. Set $\mathbb{Q} := \mu_{OOD}$. Lemma 7.5 implies that under the respective assumptions: Assumption 7.3 and 7.4, the hypothesis classes $\mathscr{N}$ and $\widetilde{\mathscr{N}}$ are Lipschitz and it provides explicit estimates on the Lipschitz constants $L_{\mathcal{F}}$ of these neural operators. The result then follows from Lemma 7.10. □

Discussion Theorem 7.6 supports our experimental evidence that the risk-bounds for the $(sNO + \varepsilon I)v2$ are much tighter than those for the $sNO + \varepsilon I$ model, precisely since the constant of the former is much tighter than that of the latter. We expect that comparable lower-bounds could be derived. However, since lower-bounds with tight constants can take years to perfect, as seen by the time gap between [104] and [64], then we will in future research.

[27] I.e.: There exists some $\eta_0 > 0$ such that (i) holds whenever $0 < \eta \leq \eta_0$.

8. Generalization error bounds of the neural operators

Through experimental observation, we have found that our proposed network exhibits superior performance compared to standard networks, specifically in terms of lower test errors. The test error is synonymous with generalization error in the field of statistical learning theory. This section provides the theoretical analysis of generalization error for both standard networks and our proposed networks.

It is important to mention that Kovachki et al. [66] established the standard universal approximation theorem that shows that any continuous operator can be approximated in compact sets by standard neural operators. As our network is an extension of the standard network, universality also holds for our proposed networks. Consequently, in the context of universality, we are unable to distinguish differences. Therefore, our primary focus on this section will be on the complexity of networks and their corresponding generalization error bounds.

8.1. Preliminaries

Let $D \subset \mathbb{R}^d$ be a bounded domain, and $\mathrm{L}^2(D;\mathbb{R}^h)$ be the L^2 space of $\mathbb{R}^h$-value function on D. Let $S = \{(a_i, u_i) : 1 \le i \le n\}$ be the sequence of independent samples of μ, i.e. $(a_i, u_i) \overset{\text{i.i.d}}{\sim} \mu$,[28] with marginals μ_a in $\mathrm{L}^2(D;\mathbb{R}^{d_a})$ and μ_u in $\mathrm{L}^2(D;\mathbb{R}^{d_u})$. Let $\mathscr{G}$ be the class of operators mapping from $\mathrm{L}^2(D;\mathbb{R}^{d_a})$ to $\mathrm{L}^2(D;\mathbb{R}^{d_u})$, and $\ell : \mathrm{L}^2(D;\mathbb{R}^{d_u}) \times \mathrm{L}^2(D;\mathbb{R}^{d_u}) \to \mathbb{R}_{\ge 0}$ be the loss function. We denote by the expected risk $\mathcal{L}$ and empirical risk $\widehat{\mathcal{L}}_{\mathrm{S}}$, defined rigorously in Appendix A.8.

We review the Rademacher complexity, which measures the richness of a class of real-valued functions.

Definition 8.1. (Rademacher complexity) Let $\mathscr{F}$ be the set of real-valued measurable functions on a measurable space $(S, \mathcal{S})$. Let $\{\epsilon_i\}_{i=1}^n$ is a sequence of i.i.d. RV's with Rademacher distribution; i.e., $\mathbf{P}\{\epsilon_i = 1\} = 1/2 = \mathbf{P}\{\epsilon_i = -1\}$. The Rademacher complexity of the class $\mathscr{F}$ is defined as

$$\mathfrak{R}_{\mathrm{S}}^n(\mathscr{F}) := \mathbb{E}_{\epsilon\sim\mathrm{Rad}}\left[\sup_{f\in\mathscr{F}} \frac{1}{n}\left|\sum_{i=1}^n \epsilon_i f(a_i, u_i)\right|\right], \tag{34}$$

(Cf. Giné and Nickl [41, Definition 3.1.19]).

Assumption 8.2. There exist positive constants $\rho > 0$, $R_u > 0$ such that

(i). ℓ is ρ-Lipschitz continuous, i.e., $|\ell(u_1, v) - \ell(u_2, v)| \le \rho \left\|u_1 - u_2\right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})}$ for $u_1, u_2, v \in \mathrm{L}^2(D;\mathbb{R}^{d_u})$.
(ii). $\ell(\mathbf{0}, \cdot)$ is bounded above by R_u, i.e., $|\ell(\mathbf{0}, u)| \le R_u$ for $u \in \mathrm{supp}(\mu_u)$.[29]

First, we estimate the generalization error bound for the general setting.

Lemma 8.3 *(Generalization error bound). Let Assumption 8.2 hold and suppose there exists $R > 0$ such that $\|\mathcal{G}(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})} \le R$, for all $\mathcal{G} \in \mathscr{G}$, and $a \in \mathrm{supp}(\mu_a)$ for the hypothesis class, $\mathscr{G}$. Hence, for any $\delta > \log 2$, the following inequality holds with probability greater than $1 - 2\exp(-\delta)$,*

$$\mathcal{L}(\mathcal{G}) \le \widehat{\mathcal{L}}_{\mathrm{S}}(\mathcal{G}) + 2\mathfrak{R}_{\mathrm{S}}^n(\mathscr{F}_{\mathscr{G}}) + (\rho R + R_u)\sqrt{\frac{2\delta}{n}}, \quad \forall \mathcal{G} \in \mathscr{G}, \tag{35}$$

where $\mathfrak{R}_{\mathrm{S}}^n(\mathscr{F}_{\mathscr{G}})$ is the Rademacher complexity of the class $\mathscr{F}_{\mathscr{G}}$, and the class $\mathscr{F}_{\mathscr{G}}$ is defined as

$$\mathscr{F}_{\mathscr{G}} := \{(a, u) \mapsto \ell(\mathcal{G}(a), u) \ : \ (a, u) \in \mathrm{supp}(\mu), \mathcal{G} \in \mathscr{G}\}.$$

See Appendix C.1 for the proof. The idea is to break down the generalization error $\mathcal{L}(\mathcal{G})$ into two components: the approximation error $\widehat{\mathcal{L}}_{\mathrm{S}}(\mathcal{G})$ and the complexity error $\mathcal{L}(\mathcal{G}) - \widehat{\mathcal{L}}_{\mathrm{S}}(\mathcal{G})$. The upper bound of the complexity error $\mathcal{L}(\mathcal{G}) - \widehat{\mathcal{L}}_{\mathrm{S}}(\mathcal{G})$ can be established using the *Rademacher complexity*, $\mathfrak{R}_{\mathrm{S}}^n(\mathscr{F}_{\mathscr{G}})$, by the *Uniform laws of large numbers* (Lemma A.12).

If the class $\mathscr{G}$ is a universal approximator, the approximation error $\widehat{\mathcal{L}}_{\mathrm{S}}(\mathcal{G})$ can be made "*small enough*" through training. In fact, if $\mathscr{G}$ is chosen to be the classes of neural operators [66] and DeepONets [71], both of which are universal approximators. In the following, we focus on the analysis of the Rademacher complexity for both standard neural operators (NO*s*) and proposed neural operators (*s*NO).

[28] Independent identically distributed samples drawn from, μ, on $\mathrm{L}^2(D;\mathbb{R}^{d_a}) \times \mathrm{L}^2(D;\mathbb{R}^{d_u})$.
[29] Support of a measure μ is defined $\mathrm{supp}(\mu) := \{x \in X : \mu(\mathcal{U}) > 0 \text{ for all open neighborhood } \mathcal{U} \text{ of } x\}$ (Cf. Ambrosio et al. [4, Ch. 5]).

8.2. Related work of generalization error bound (GEB)

References such as Bartlett et al. [9], Jakubovitz et al. [58] have extensively investigated generalization error bounds (GEB) for networks that map between finite-dimensional spaces. However, to the best of our knowledge, there has been limited exploration of GEB for operators on infinite dimensional spaces. De Ryck and Mishra [27] provided the GEB for (general) operator architectures using Hoeffding's inequality, without involving the analysis of the Rademacher complexity. Gopalani et al. [43] and Kim and Kang [62] have provided GEB for DeepOnet and FNOs, respectively, by the Rademacher complexity. However, in these works, the authors assumed that the trainable parameters are *finite-dimensional* (such as matrices), *while our work does not need this assumption*. Our study distinguishes itself from Kim and Kang [62] in several key aspects. Firstly, we directly analyze the integral operator under the assumption of Lipschitz continuity of the kernel, whereas Kim and Kang [62] assumes a truncated expansion for FNOs and evaluates the Rademacher complexity based on the number of truncations. Secondly, our work not only generalizes the findings of Kim and Kang [62] but also provides sharper bounds on the Rademacher complexity with the order $\mathcal{O}(1/n^{\frac{1}{\hat{d}+1}})$, compared to $\mathcal{O}(1)$ in Kim and Kang [62].

8.3. Rademacher complexity of neural operators

We analyze the Rademacher Complexity of Neural Operators, [66]. Under Assumption 7.3, we obtain the following upper bound for Rademacher Complexity for NOs.

Theorem 8.4 *(Rademacher Complexity for NOs). Let suppose Assumptions 8.2 and 7.3 hold. Then,*

$$\mathfrak{R}_{\mathrm{S}}^{n}(\mathscr{F}_{\mathscr{N}}) \leq \gamma\, L^{\frac{\hat{d}+2}{\hat{d}+1}} \{(C_w + C_k)C_\sigma\}^L \left(\frac{1}{n}\right)^{\frac{1}{\hat{d}+1}}, \tag{36}$$

where $\hat{d} := \mathrm{ddim}(D \times D)$ is the doubling dimension of $D \times D$ (see Definition A.16), and γ is the positive constant independent of L and n, defined in (C.16).

See Appendix C.2 for the proof. The idea behind the proof is as follows, the Rademacher Complexity, $\mathfrak{R}_{\mathrm{S}}^{n}(\mathscr{F}_{\mathscr{N}})$, is evaluated by using *Dudley's Theorem* (Lemma A.15, and Kakade and Tewari [59], Bartlett et al. [8]). The upper bound is then determined by the *covering number* (as defined in Definition A.14). Since NOs are parameterized by their weight matrices and (kernel) Lipschitz continuous functions, the evaluation of the covering number ultimately involves analyzing these components, by using Wainwright [111] and Gottlieb et al. [44], respectively.
See Remark D.1 for finite basis expansion (applicable integral kernel).

8.4. Rademacher of sNO and intermediate architectures

In this section, we analyze the Rademacher Complexity of the proposed networks. With Assumption 7.4, we obtain the following upper bound for the Rademacher Complexity of $\widetilde{\mathscr{N}}$.

Theorem 8.5 *(Rademacher Complexity of proposed network(s)). Let Assumptions 8.2 and 7.4 hold. Then,*

$$\mathfrak{R}_{\mathrm{S}}^{n}(\mathscr{F}_{\widetilde{\mathscr{N}}}) \leq \widetilde{\gamma} L^{\frac{1}{\hat{d}+1}} \left(\sum_{\ell=0}^{L} \frac{\mathbf{X}_\ell C_w^{M+1} C_\sigma^M}{\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M} + \frac{\mathbf{X}_\ell}{\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma} \right) \left[\prod_{\ell=0}^{L} (\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma) \right] \left(\frac{1}{n}\right)^{\frac{1}{\hat{d}+1}}, \tag{37}$$

where $\widetilde{\gamma}$ is the positive constant independent of L and n, defined in (C.27).

Theorem 8.5 can be proved by similar arguments in Theorem 8.4. See Appendix C.3 for the proof.

8.5. GEB and comparison among architectures

By Lemma 8.3, and Theorems 8.4 and 8.5, we get

Corollary 8.6. *Let Assumptions 8.2 and 7.3 hold. Then, for any $\delta > \log 2$ and $\mathcal{G} \in \mathscr{N}$, the following inequality holds, with probability greater than $1 - 2\exp(-\delta)$:*

$$\mathcal{L}(\mathcal{G}) \leq \widehat{\mathcal{L}}_{\mathrm{S}}(\mathcal{G}) + 2\gamma L^{\frac{\hat{d}+2}{\hat{d}+1}} \{(C_w + C_k)C_\sigma\}^L \left(\frac{1}{n}\right)^{\frac{1}{\hat{d}+1}} + \left(\rho\{(C_w + C_k)C_\sigma\}^L (C_w + C_k)C_a + R_u\right) \sqrt{\frac{2\delta}{n}}. \tag{38}$$

See Appendix C.4 for the proof.

Corollary 8.7. *Let Assumptions 8.2 and 7.4 hold. Then, for any $\delta > \log 2$ and $\mathcal{G} \in \widetilde{\mathcal{N}}$, the following inequality with probability greater than $1 - 2\exp(-\delta)$:*

$$\begin{aligned}
&\mathcal{L}(\mathcal{G}) \leq \widehat{\mathcal{L}}_S(\mathcal{G}) \\
&+ 2\widetilde{\gamma} L^{\frac{1}{d+1}} \left(\sum_{\ell=0}^{L} \frac{\mathbf{X}_\ell C_w^{M+1} C_\sigma^M}{\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M} + \frac{\mathbf{X}_\ell}{\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma} \right) \left[\prod_{\ell=0}^{L} (\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma) \right] \left(\frac{1}{n} \right)^{\frac{1}{d+1}} \\
&+ \left(\rho \left[\prod_{\ell=0}^{L} (\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma) \right] C_a + R_u \right) \sqrt{\frac{2\delta}{n}}.
\end{aligned} \tag{39}$$

See Appendix C.4 for the proof.

Remark 8.1. When $\mathbf{Z}_\ell = 0$ and $\mathbf{X}_\ell = 1$ corresponds to sNO, if $\mathbf{Z}_\ell = 1$ and $\mathbf{X}_\ell = 1$ to (sNO + εI)v1. Finally, if $\mathbf{Z}_\ell = 1$ and $\mathbf{X}_\ell$ is a Bernoulli RV with $\mathbf{P}\{\mathbf{X}_\ell = 1\} = p_\ell$, and $\mathbf{P}\{\mathbf{X}_\ell = 0\} = 1 - p_\ell$ for $p_\ell \in [0, 1]$ corresponds to (sNO + εI)v2.

Remark 8.2. The 2nd, and 3rd terms decay as the samples increases, $n \to \infty$, with orders $\mathcal{O}\left(1/n^{\frac{1}{d+1}}\right)$ and $\mathcal{O}(1/n^{\frac{1}{2}})$, respectively. We finally observe the coefficients depending on the number of layers, L (see also Remark C.2–submitted version of the paper– and D.2 in the revised version of the paper–.

1. If $(C_w + C_k)C_\sigma < 1$ (or $C_w^{M+1} C_\sigma^{M+1} C_k < 1$), the upper bounds of standard NO (or sNO) remain bounded as L tends to infinity. On the other hand, if $(C_w + C_k)C_\sigma > 1$ (or $C_w^{M+1} C_\sigma^{M+1} C_k > 1$), then, the upper-bounds diverges with depth, similarly than finite-dimensional networks Truong [108].
2. If the condition $C_w < 1$ and $C_\sigma \leq 1$ holds true, then $C_w^{M+1} C_\sigma^{M+1} C_k \leq (C_w + C_k)C_\sigma$, which implies that the upper bound of sNO are smaller than standard NOs. See Remark D.2.
3. Since $C_w^{M+1} C_\sigma^M C_k \leq (1 + C_w^{M+1} C_\sigma^M)(1 + C_\sigma C_k)$, the upper bound of standard NOs are smaller than (sNO + εI)v1, despite the outcomes of our experiments, see Fig. 7.
4. Finally the RVs can control the GEB. If $\mathbf{P}\{\boldsymbol{X}_\ell = 1\} = p_\ell = x_\ell / L^{\frac{1}{d+1}}$, where $x_\ell \in [0, 1]$ satisfies $\sum_{\ell=0}^{\infty} x_\ell < \infty$, the upper bound for ($s$NO + εI)v2 does not blow up as L increases, *regardless of* C_w, C_k, *and* C_σ. The expectation with respect to $\boldsymbol{\mathcal{X}} = (\mathbf{X}_0, ..., \mathbf{X}_L)$ is bounded above by the expression (see Lemma D.1 in Appendix D)

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\mathcal{X}}}[\text{RHS of (39)}] \lesssim \widehat{\mathcal{L}}_S(\mathcal{G}) + \left(\sum_{\ell=1}^{L} x_\ell \right) \prod_{\ell=0}^{L} \left[1 + (C_w^{M+1} C_\sigma^M + C_k C_\sigma + C_w^{M+1} C_k C_\sigma^{M+1}) x_\ell \right] \left(\frac{1}{n} \right)^{\frac{1}{d+1}} \\
+ \left(\rho \prod_{\ell=0}^{L} \left[1 + (C_w^{M+1} C_\sigma^M + C_k C_\sigma + C_w^{M+1} C_k C_\sigma^{M+1}) x_\ell \right] C_a + R_u \right) \sqrt{\frac{2\delta}{n}},
\end{aligned}$$

whose coefficients do not blow up as $L \to \infty$ (the infinite products converge because $\sum_{\ell=0}^{\infty} x_\ell < \infty$, see, Trench [106]). Here, $\lesssim$ implies that the left-hand side is bounded above by the right-hand side times a constant independent of n and L. For example, if x_ℓ decay with order $\mathcal{O}(\ell^{-(1+\varepsilon)})$ for some $\varepsilon > 0$, then it holds that $\sum_{\ell=0}^{\infty} x_\ell < \infty$.[30] [55] proposed linear decay, which does not satisfy $\sum_{\ell=0}^{\infty} x_\ell < \infty$. However, it is assumed that the number of layers L is finite (typically around 100), our analysis on the other hand showed that the upper bound is valid regardless of the number of layers if the Bernoulli RVs satisfied the above-mentioned condition. A less *restrictive* decay on the RVs can be chosen.

Therefore, our proposed architecture, especially (sNO + εI)v2, would have a smaller generalization error than the standard architecture under assumptions of the RVs.

Remark 8.3. Any looseness in the existing bounds results from the techniques employed in our proofs, which hopefully can be tightened in future works. These are the best available generalization bounds for both models and both are compared on fair footing as they are derived by the same argument. Nevertheless, the bounds are not intrinsic in the sense that there is no lower bound for the classical NO model.

9. Summary and discussion

We perform a detailed empirical and theoretical analysis of the generalization capabilities of neural operators and sNO + εI for approximating the parametric form of the Helmholtz equation, as well as a surrogate model for the forward operator associated with

[30] Notice that, the assumption of $\sum_{\ell=0}^{\infty} p_\ell < \infty$ by the Borel–Cantelli lemma, implies that the probability that infinitely many of $X_\ell = 1$ (layers that are active) occur is zero.

the study of the inverse boundary value problem for the Helmholtz equation. We work with high-frequency given the documented difficulties of numerical methods, [34,33,39], and the amount of previous work associated with other PDEs, which traditional neural operators already approximate remarkably well.[31]

The sNO + εI family demonstrated improved performance without increasing the number of parameters (in the case without stochastic depth) or compromising the approximation capabilities of traditional neural operators for high-frequency Helmholtz problems. We maintained strict constraints throughout our analysis, including not increasing the size of the training dataset, and testing on datasets of comparable size as those used in the training.

We conduct a thorough empirical analysis of the stability of the trained networks to different realizations of the wave speed, and (sFNO + εI)v2 demonstrated resilience to these changes. In light of these results, we derive upper bounds for out-of-distribution generalization for Gaussian measures in abstract Banach spaces, and we link the experimental behavior to the presence of the random variables presented in stochastic depth. For the results in-distribution, we also provide an upper bound of the generalization error by estimating the Rademacher complexity of each of the networks. Similarly, showing that the random variables in stochastic depth are effectively controlling the complexity of the hypothesis class for the (sNO + εI)v2 family.

We have made progress in understanding the theoretical guarantees of neural operators and similar architectures, going beyond their approximation property. However, it is worth noting that one of the limitations of our work is that the bounds we derived are not tight. Although deriving lower bounds presents a challenge, we remain optimistic about the possibility of making further advancements in this area.

On the experimental side, our results suggest that it is possible to capture the forward operator effectively (mapping functions to operators), and we expect to apply this surrogate model to solve inverse problems, particularly for Bayesian inversion and for using algorithms that only require multiple evaluations of the costly forward operator, such as the derivative-free ensemble Kalman method [56].

CRediT authorship contribution statement

Jose Antonio Lara Benitez: Conceptualization, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Takashi Furuya:** Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Methodology. **Florian Faucher:** Data curation, Software, Writing – original draft, Writing – review & editing. **Anastasis Kratsios:** Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Conceptualization. **Xavier Tricoche:** Software, Visualization, Writing – original draft, Writing – review & editing. **Maarten V. de Hoop:** Funding acquisition, Methodology, Project administration, Supervision, Conceptualization, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The code is publicly available on [75], and the dataset can be found on [74].

Acknowledgements

J.A.L.B. is grateful to colleagues at Petroleum Geo-Services for their insightful discussions during their 2022 internship, valuable assistance with resources, and diligent work on applying a previous version of the proposed network to field data [54]. J.A.L.B. also appreciates the support from ChatGPT in optimizing TikZ code.[32] T.F. was supported by Research Grant for Young Scholars funded by Yamanashi Prefecture. F.F. acknowledges the use of the cluster PlaFRIM[33] for the dataset generation. F.F. also acknowledges funding by the European Union with ERC Project INCORWAVE – grant 101116288. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency (ERCEA). Neither the European Union nor the granting authority can be held responsible for them. X.M.T. would like to thank the Community Cluster Program and the Rosen Center for Advanced Computing and Purdue. M.V. de H. gratefully acknowledges support from the Department of Energy under grant DE-SC0020345, the National Science Foundation under grant DMS-2108175, the Simons Foundation under the MATH + X program (Award# 271853, Project "Simons Chair in Computational and Applied Mathematics and Earth Science at Rice University"), and the corporate members of the Geo-Mathematical Imaging Group at Rice University. A.K. was funded by the NSERC (grants no. RGPIN-2023-04482 and DGECR-2023-00230).

[31] Darcy flow.

[32] The author reviewed and edited the content as needed and take full responsibility for the content.

[33] https://www.plafrim.fr/.

Appendix A. Preliminaries

A.1. Vector-valued L^2 spaces and Sobolev spaces

$L^2(D;\mathbb{R}^{d_a})$ is the L^2 space of $\mathbb{R}^{d_a}$-value functions on $D \subset \mathbb{R}^d$. It is defined as the space of functions such that,

$$\|a\|^2_{L^2(D;\mathbb{R}^{d_a})} := \int_D \|a(x)\|_2^2 \, dx < \infty,$$

where $D \ni x \mapsto \|a(x)\|_2^2 = \sum_j a_j^2(x)$; notices that, $\|\cdot\|_2^2$ is the usual ℓ_2-norm in $\mathbb{R}^{d_a}$.
For natural number $k \in \mathbb{N}_0$, we define Sobolev space $H^k(D;\mathbb{R}^{d_a})$ by

$$H^k(D;\mathbb{R}^{d_a}) := \left\{ u \in L^2(D;\mathbb{R}^{d_a}) : \partial_x^\alpha u \in L^2(D;\mathbb{R}^{d_a}) \ \forall |\alpha| \le k \right\}.$$

For positive non-integer $s > 0$, we define Sobolev space $H^s(D;\mathbb{R}^{d_a})$ by

$$H^s(D;\mathbb{R}^{d_a}) := \left\{ u \in H^{\lfloor s \rfloor}(D;\mathbb{R}^{d_a}) : \sup_{|\alpha| = \lfloor s \rfloor} [\partial_x^\alpha u]_{\theta_s, D} < \infty \right\}, \tag{A.1}$$

where $\theta_s := s - \lfloor s \rfloor \in (0,1)$. Here, $[f]_{\theta,D}$ is defined by

$$[f]_{\theta,D} := \left(\int_D \int_D \frac{\|f(x) - f(y)\|_2^2}{\|x - y\|_2^{2\theta + d}} dx dy \right)^{1/2}.$$

For further details, we refer to, e.g., Adams and Fournier [3].

A.2. Bounded linear operator

Definition A.1 *(Bounded linear operator).* We say that $\boldsymbol{A} : X \to Y$ is a bounded linear operator mapping from a Banach space X to a Banach space Y, if it is linear and if there exists a positive constant $C > 0$ such that,

$$\|\boldsymbol{A}x\|_Y \le C \|x\|_X, \ x \in X.$$

Definition A.2 *(Operator norm).* The operator norm $\|A\|_{\text{op}}$ for a bounded linear operator A is

$$\|\boldsymbol{A}\|_{\text{op}} := \inf \left\{ C \in \mathbb{R}_{\ge 0} \ : \ \|\boldsymbol{A}x\|_Y \le C \|x\|_X \right\}.$$

Neural Operators [82] are typically built from bounded linear integral operators

Definition A.3 *(Bounded linear integral operator).* It is an Linear Bounded Operator $\boldsymbol{\mathcal{K}} : L^2(D;\mathbb{R}^n) \to L^2(D;\mathbb{R}^m)$ defined by

$$x \mapsto (\boldsymbol{\mathcal{K}} g)(x) := \int_D k(x,y) \, g(y) \, dy, \quad x \in D, \ g \in L^2(D;\mathbb{R}^n),$$

where $k : D \times D \subset \mathbb{R}^{d \times d} \to \mathbb{R}^{m \times n}$ is the Integral Kernel.

Definition A.4 *(Lipschitz kernel).* We say a vector-valued Integral Kernel is Lipschitz continuous if there exists $C > 0$ such that

$$|k_{i,j}(x,y) - k_{i,j}(x',y')| \le C \left\| (x,y) - (x',y') \right\|_2, \quad (x,y), (x',y') \in D \times D,$$

for $i, j \in \{1, \dots, d\}$.

A.3. Neural operator

Let D a bounded domain and let $\mathcal{A}(D;\mathbb{R}^{d_a})$, $\mathcal{U}(D;\mathbb{R}^{d_{v_i}})$, and $\mathcal{U}(D;\mathbb{R}^{d_u})$ be abstract (separable) Banach spaces.

Definition A.5 *(Neural Operator).* Let define $\mathcal{G}_\theta : \mathcal{A}(D;\mathbb{R}^{d_a}) \to \mathcal{U}(D;\mathbb{R}^{d_u})$ such that

$$u = \mathcal{G}_\theta(a) = \boldsymbol{Q} \circ \mathcal{L}_k \circ \dots \circ \mathcal{L}_1 \circ \boldsymbol{R}(a), \tag{A.2}$$

in where $\boldsymbol{Q} : \mathcal{A}(D;\mathbb{R}^{d_a}) \to \mathcal{U}(D;\mathbb{R}^{d_{v_1}})$ (Lifting map), and $\boldsymbol{R} : \mathcal{U}(D;\mathbb{R}^{d_{v_{k+1}}}) \to \mathcal{U}(D;\mathbb{R}^{d_u})$ (Projection map), such that

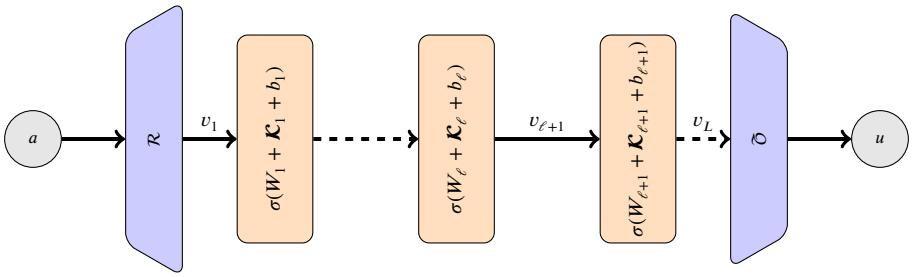


Fig. A.17. NO. Neural Operator architecture.

$$\boldsymbol{R}(a)(x) := (Ra(x)), \quad R \in \mathbb{R}^{d_{v_1} \times d_a}. \tag{A.3a}$$

$$\boldsymbol{Q}(v)(x) := (Qv(x)), \quad Q \in \mathbb{R}^{d_u \times d_{v_{k+1}}}, \tag{A.3b}$$

and $\mathcal{L}_i$, $(i = 1, \dots, k)$ is defined as

$$D \ni x \mapsto \left(\mathcal{L}_i v\right)(x) := \sigma\left(W_i v(x) + (\boldsymbol{\mathcal{K}}_i v)(x)\right), \quad W_i \in \mathbb{R}^{v_{i+1} \times v_i}. \tag{Layers}$$

$i = 1, \dots, k$, and $\boldsymbol{\mathcal{K}}_i$ is an integral operator mapping from $\mathcal{U}(D; \mathbb{R}^{d_{v_i}})$ to $\mathcal{U}(D; \mathbb{R}^{d_{v_{i+1}}})$, see Definition A.3. In the definition of Kovachki et al. [66, Section 9.1], Neural Operators parameterize the integral kernel as neural networks, which satisfies the Lipschitz continuity used in the Assumption 7.4.

A.3.1. Fourier neural operators (FNOs)

A natural ansatz in the integral operator is assuming to be convolutional, so that,

$$(k \star v) = \boldsymbol{\mathcal{F}}^{-1}\left(\boldsymbol{\mathcal{F}}(k) \cdot \boldsymbol{\mathcal{F}}(v)\right). \tag{A.4}$$

[34] if the kernel function and v lies on the adequate space, say L^2. When Equation (A.4) is estimated by the FFT algorithm, the Neural Operator is efficiently implemented, leading to the network presented in Li et al. [82].

A.3.2. Remark: universality of sNO

Kovachki et al. [66, Theorem 11] have shown that the compositional operator $(\sigma \circ \boldsymbol{\mathcal{K}}_L) \circ \cdots \circ (\sigma \circ \boldsymbol{\mathcal{K}}_1)$ of the linear integral operator $\boldsymbol{\mathcal{K}}_\ell$ and the element-wise nonlinear activation function σ, can approximate any nonlinear continuous operator. Therefore, the addition of any local operation in Neural Operators does not affect the universality property, i.e., standard, and sequential NOs have the same universality property.

A.4. Bochner integral

In the study of generalization error bounds, the Expected error, see Appendix A.8, is defined through the Bochner Integral. We briefly introduce it, informally, as the natural generalization of the Lebesgue integral on (separable) Banach spaces.

For our purpose, it suffices to define the integral (informally) on $\mathrm{L}^2(D; \mathbb{R}^{d_a}) \times \mathrm{L}^2(D; \mathbb{R}^{d_u})$. Assume that a function $(a, u) \mapsto f(a, u) \in \mathbb{R}$ is Bochner integrable with respect to the measure μ on $\mathrm{L}^2(D; \mathbb{R}^{d_a}) \times \mathrm{L}^2(D; \mathbb{R}^{d_u})$, i.e., there exists a sequence of integrable simple functions s_n (the finite linear combination of indicator functions of measurable sets) such that

$$\lim_{n \to \infty} \int \left| f(a, u) - s_n(a, u) \right| d\mu(a, u) = 0.$$

Thus, the Bochner Integral is defined by

$$\int \ell(\boldsymbol{\mathcal{G}}(a), u)\, d\mu(a, u) = \lim_{n \to \infty} \int s_n(a, u)\, d\mu(a, u).$$

For a detailed (formal) definition of the Bochner integral, as well as its properties, see Yoshida [117].

A.5. Gaussian measure

The typical choice of the measure μ in the context of PDEs is the Gaussian Measure, which will be reviewed as follows (refer to, e.g., Stuart [103, Section 6]): First, a function $m \in X$ is called the mean of μ if for all $\ell \in X^*$, where X^* denote the dual space of linear functionals on X,

[34] $\boldsymbol{\mathcal{F}}$, and $\boldsymbol{\mathcal{F}}^{-1}$ represents the Fourier and Inverse Fourier transform respectively.

$$\ell(m) = \int_X \ell(x)\mu(dx).$$

A linear operator $\boldsymbol{C} : X^* \to X$ is called the Covariance Operator if for all $k, \ell \in X^*$,

$$k(\boldsymbol{C}\ell) = \int_X k(x-m)\ell(x-m)\mu(dx).$$

We say that u draws from *Gaussian Measure* $\mathcal{N}(m, \boldsymbol{C})$ (write $u \sim \mathcal{N}(m, \boldsymbol{C})$) if for all $\ell \in X^*$, $\ell(u)$ draws from the one-dimensional Gaussian distribution $\mathcal{N}(\ell(m), \ell(\boldsymbol{C}\ell))$.

If X is a Hilbert space, then we can characterize random draws from a Gaussian Measure by using the *Karhunen-Loéve expansion* as follows (see, e.g., Stuart [103, Theorem 6.19]):

Theorem A.6. *Let X be a Hilbert space, and let $\boldsymbol{C} : X \to X$ be a self-adjoint, positive semi-definite, compact operator, and let $m \in X$. Let $\{\phi_k, \gamma_k\}_{k=1}^{\infty}$ be an orthonormal set of eigenvectors and eigenvalues for $\boldsymbol{C}$ ordered so that*

$$\gamma_1 \geq \gamma_2 \geq \cdots.$$

Take $\{\xi_k\}_{k=1}^{\infty}$ to be an i.i.d. sequence with $\xi_1 \sim \mathcal{N}(0,1)$. Then, the random variable $u \in X$ given by the Karhunen-Loéve expansion

$$u = m + \sum_{k=1}^{\infty} \sqrt{\gamma_k}\xi_k\phi_k \tag{A.5}$$

draws from $\mathcal{N}(m, \boldsymbol{C})$.

A.6. Cameron-Martin space

We briefly review the definition of the Cameron-Martin space (refer to, e.g., Hairer [49, Section 3.2.]).

Definition A.7. Let μ be a Gaussian Measure on a separable Banach space X. The Cameron-Martin space $\mathcal{H}_\mu$ of μ is the completion of the linear subspace

$$\{h \in X : \exists h^* \in X^* \text{ with } C_\mu(h^*, \ell) = \ell(h)\ \forall \ell \in X^*\},$$

under the norm

$$\|h\|_\mu^2 = \langle h, h\rangle_\mu = C_\mu(h^*, h^*),$$

where $C_\mu : X^* \times X^* \to \mathbb{R}$ is defined by

$$C_\mu(k, \ell) := \int_X k(x)\ell(x)\mu(dx), \quad k, \ell \in X^*.$$

In can be shown that $\mathcal{H}_\mu$ is a reproducing kernel Hilbert space with the inner product $\langle h, k\rangle_\mu = C_\mu(h^*, k^*)$.

When X is a finite-dimensional space, the Cameron-Martin space is given by the range of the covariance matrix [49, Exercise 3.28].
We now review properties of the Cameron-Martin space (see Hairer [49, Theorem 3.41 and Proposition 3.4.2]).

Theorem A.8. *For $h \in X$, we define the map $T_h : X \to X$ by $T_h(x) = x + h$. Then, the push-forward measure $T_{h\sharp}\mu$ of μ by T_h is absolutely continuous with respect to μ if and only if $h \in \mathcal{H}_\mu$.*

Proposition A.9. *The space $\mathcal{H}_\mu \subset B$ is the intersection of all (measurable) linear subspaces of full measure. However, if $\mathcal{H}_\mu$ is infinite-dimensional, then one has $\mu(\mathcal{H}_\mu) = 0$.*

That is, the Cameron-Martin space $\mathcal{H}_\mu$ of μ represents the directions in X where translation is invariant, meaning that the translated measure has the same null sets as the original measure. Furthermore, when $\dim(\mathcal{H}_\mu) = \infty$, $\mathcal{H}_\mu$ is "smaller" than X in the sense that $\mu(\mathcal{H}_\mu) = 0$. In contrast, the finite-dimensional Lebesgue measure is invariant under translations in any direction. This is an illustration of the tendency for measures in infinite-dimensional spaces to be mutually singular.

A.7. Gaussian random field

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space. We say that a function $u : D \times \Omega \to \mathbb{R}$ is a Gaussian Random Field (GRF) if $u(x, \cdot) \in L^2(\Omega)$, and for any $x_1, \ldots, x_M \in D$ and any $M \in \mathbb{N}$,

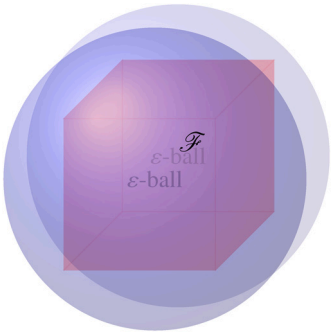


Fig. A.18. Illustration of the covering number on $\mathscr{F}$.

$$\boldsymbol{u}_M := (u(x_1,\cdot),...,u(x_M,\cdot))^{\mathrm{T}}$$

draws from the multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{m}_M, \boldsymbol{C}_M)$. Here, $m(x) := \mathbb{E}_\omega[u(x,\omega)]$ is the mean function, and $c(x,y) = \mathbb{E}_\omega[(u(x,\omega)-m(x))(u(y,\omega)-m(y))^*]$ is the covariance function. We have denoted by $\boldsymbol{m}_M := (m_1,...,m_M)^{\mathrm{T}}$ and $\boldsymbol{C}_M = (c_{ij})_{i,j=1}^M$, where $m_i := m(x_i)$, and $c_{ij} := c(x_i,x_j)$. The GRF also has the Karhunen-Loéve expansion with (A.5) as $X = L^2(D)$, m is the mean function, and $\boldsymbol{C}$ is the integral operator with the kernel given by the covariance function (see Lord et al. [87, Theorem 7.52]).

We can construct the GRF drawing from a certain Gaussian Measure. We simply consider the Gaussian Measure $\mathcal{N}(0,(-\Delta)^{-\alpha})$ where Δ is the Laplacian with domain $H_0^1(D)\cap H^2(D)$ where $D=[0,1]^2$ and $\alpha>1$. Then, the draw u from $\mathcal{N}(0,(-\Delta)^{-\alpha})$ are almost surely in $C(D)$ (see Stuart [103, Example 6.28]), which means that the function u can be point-wisely defined, and then, for any $x_1,...,x_M\in D$ and any $M\in\mathbb{N}$, $(u(x_1,\cdot),...,u(x_M,\cdot))^{\mathrm{T}}$ draws from the multivariate Gaussian distribution, that is, u is the GRF.

A.8. Statistical learning

Definition A.10 *(Expected Risk/Loss).* The Expected risk is defined by

$$\mathcal{L}(\mathcal{G}) := \mathbb{E}_{(a,u)\sim\mu}\left[\ell(\mathcal{G}(a),u)\right] = \int_{\mathrm{supp}(\mu)} \ell(\mathcal{G}(a),u)\,\mu(d(a,u)),$$

with respect to $\mathcal{G}\in\mathscr{G}$, where the set $\mathscr{G}$ is the hypothesis class. For the purpose of this paper, the class corresponds to Neural Operators or sequential Neural Operators, and $\ell : \mathrm{L}^2(D;\mathbb{R}^{d_u})\times\mathrm{L}^2(D;\mathbb{R}^{d_a})\to[0,\infty)$ is the loss function.

Definition A.11 *(Empirical Risk/Loss).* It is defined as the unbiased estimator of the Expected risk, that is

$$\widehat{\mathcal{L}}_S(\mathcal{G}) := \frac{1}{n}\sum_{i=1}^n \ell(\mathcal{G}(a_i),u_i),$$

where $(a_i,u_i)\overset{\text{i.i.d}}{\sim}\mu$.

The generalization error $\mathcal{L}(\mathcal{G})$ is decomposed into $\widehat{\mathcal{L}}_S(\mathcal{G})$ and $\mathcal{L}(\mathcal{G})-\widehat{\mathcal{L}}_S(\mathcal{G})$. The difference, $\mathcal{L}(\mathcal{G})-\widehat{\mathcal{L}}_S(\mathcal{G})$ between the generalization and empirical errors is evaluated using the Uniform Laws of Large Numbers (see, e.g., [111, Theorem 4.10] or [41, Theorem 3.4.5]).

Lemma A.12 *(Uniform Laws of Large Numbers). Let $\mathscr{F}$ be the set of real-valued measurable functions on a measurable space $(S,\mathcal{S})$ with absolute values bounded by R, let X_i $(i\in\mathbb{N})$ be i.i.d., S-valued random variables with common probability law $\mathbf{P}$, and let ϵ_i $(i\in\mathbb{N})$ be a sequence of i.i.d. Rademacher RVs, i.e., ϵ_i are independent, and $\mathbf{P}\{\epsilon_i=1\}=1/2=\mathbf{P}\{\epsilon_i=-1\}$. Then, for all $n\in\mathbb{N}$ and $\delta>0$, the following inequality holds with probability greater than $1-2\exp(-\delta)$,*

$$\sup_{f\in\mathscr{F}}\left|\frac{1}{n}\sum_{i=1}^n f(X_i)-\mathbb{E}[f(X)]\right| \le 2\mathfrak{R}_{\mathrm{S}}^n(\mathscr{F}) + R\sqrt{\frac{2\delta}{n}},$$

where $\mathfrak{R}_{\mathrm{S}}^n(\mathscr{F})$ is the Rademacher complexity of the class $\mathscr{F}$ defined above.

The Rademacher complexity $\mathfrak{R}_{\mathrm{S}}^n(\mathscr{F})$ of the class $\mathscr{F}$ is defined as follows.

Definition A.13. (Rademacher complexity) Let $\mathscr{F}$ be the set of real-valued measurable functions on a measurable space $(S,\mathcal{S})$. Let $\{\epsilon_i\}_{i=1}^n$ is a sequence of i.i.d. RV's with Rademacher distribution; i.e., $\mathbf{P}\{\epsilon_i=1\}=1/2=\mathbf{P}\{\epsilon_i=-1\}$. The Rademacher Complexity of the class $\mathscr{F}$ is defined as

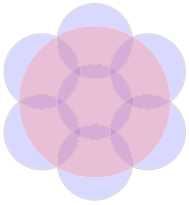


Fig. A.19. Illustration of the doubling number.

$$\mathfrak{R}_{\mathrm{S}}^{n}(\mathscr{F}) := \mathbb{E}_{\epsilon\sim\mathrm{Rad}}\left[\sup_{f\in\mathscr{F}}\frac{1}{n}\left|\sum_{i=1}^{n}\epsilon_i f(a_i,u_i)\right|\right],$$

(Cf. Giné and Nickl [41, Definition 3.1.19]).

Intuitively, Rademacher complexity $\mathfrak{R}_{\mathrm{S}}^{n}(\mathscr{F})$ measures richness of a class $\mathscr{F}$ of real-valued functions.

Definition A.14 *(Covering number).* Let $(\mathscr{F}, \|\cdot\|)$ be a normed vector space. We define, $N(\varepsilon, \mathscr{F}, \|\cdot\|)$, the covering number of $\mathscr{F}$ (sometimes known as entropy number) which means the minimal cardinality of a subset $C \subset \mathscr{F}$ that covers $\mathscr{F}$ at scale ε with respect to the norm $\|\cdot\|$. (See Fig. A.18.)

Roughly speaking, the covering number $N(\varepsilon, \mathscr{F}, \|\cdot\|)$ is the necessary number of ε-balls with respect to norm $\|\cdot\|$ to completely cover a space $\mathscr{F}$ (see e.g., Wainwright [111, Definition 5.1]). Furthermore, it is possible to estimate Rademacher Complexity $\mathfrak{R}_{\mathrm{S}}^{n}(\mathscr{F})$ by using the *covering number*. The following lemma is known as *Dudley's Theorem* (see, e.g., Bartlett et al. [8, Lemma A.5]).

Lemma A.15 *(Dudley's Theorem). Let $\mathscr{F}$ be the set of real-valued functions. Then,*

$$\mathfrak{R}_{\mathrm{S}}^{n}(\mathscr{F}) \le \inf_{\alpha\ge 0}\left\{4\alpha + \frac{12}{\sqrt{n}}\int_{\alpha}^{\infty}\sqrt{\log N(\varepsilon,\mathscr{F},\|\cdot\|_{\mathrm{S}})}\,d\varepsilon\right\}$$

where $\|f\|_{\mathrm{S}} := \left(\frac{1}{n}\sum_{i=1}^{n} f(X_i)^2\right)^{1/2}$.

One of the main result in this paper is to apply these lemmas as $\mathscr{F}$ is the set of loss function $\ell(\mathcal{G}(\cdot),\cdot)$ where $\mathcal{G}$ is the class of Neural Operators or sequential Neural Operators. Neural Operators $\mathcal{G}$ are parameterized by weight matrices and Lipschitz continuous functions, and finally we will arrive at evaluating their covering number, which are referred to [111].

When we analyze the *covering number* of Lipschitz continuous functions, the doubling dimension of $D \times D$ appears. We will now review the definition of the doubling dimension of a metric space (see, e.g., [48]).

Definition A.16 *(Doubling dimension).* A metric space $(\mathbf{X}, \mathbf{d})$ with metric $\mathbf{d}$ is called doubling, if there exists a constant $M > 0$ such that for any $x \in \mathbf{X}$ and $r > 0$, it is possible to cover the ball $B_r(x) := \{y \in X \mid \mathbf{d}(x,y) < r\}$ with the union of at most M balls of radius $\frac{r}{2}$. The doubling dimension of $\mathbf{X}$ is defined by $\mathrm{ddim}(\mathbf{X}) = \log_2(M)$. (See Fig. A.19.)

Appendix B. Proofs for Section 7

B.1. Proof of Lemma 7.8

Proof. Let $1_{\mathcal{Y}}$ denote the identity map on $\mathcal{Y}$. Fix $\mathcal{G} \in \mathscr{G}$ and consider the map

$$f := \ell \circ (\mathcal{G} \times 1_{\mathcal{Y}}).$$

If $\mathcal{G}$ is constant, we are done. Therefore, assume that $\mathcal{G}$ is non-constant; whence, $\mathrm{Lip}(\mathcal{G}) > 0$. Therefore, the map $\tilde{f} : \mathcal{X} \times \mathcal{Y} \to [0,\infty)$ given by

$$\tilde{f} := \frac{1}{\mathrm{Lip}(f)}\, f,$$

is 1-Lipschitz. The Kantorovich-Rubinstein duality [110, Theorem 5.10] implies that

$$\mathbb{E}_{(a,u)\sim\mathbb{Q}}\left[\tilde{f}(a,u)\right] - \mathbb{E}_{(a,u)\sim\nu}\left[\tilde{f}(a,u)\right] \le \mathcal{W}_1(\mathbb{Q},\nu). \tag{B.1}$$

The triangle inequality and the assumption that $\mathcal{W}_1(\mathbb{Q},\mu) \le \varepsilon$ imply that the right-hand side of (B.2) may be further bounded by

$$\mathbb{E}_{(a,u)\sim\mathbb{Q}}\left[\tilde{f}(a,u)\right]-\mathbb{E}_{(a,u)\sim\nu}\left[\tilde{f}(a,u)\right]\leq\mathcal{W}_1(\mathbb{Q},\mu)+\mathcal{W}_1(\nu,\mu)\leq\varepsilon+\mathcal{W}_1(\nu,\mu), \tag{B.2}$$

Multiplying across (B.2) by $\mathrm{Lip}(f)$, using the linearity of integration, and re-arranging yields

$$\mathbb{E}_{(a,u)\sim\mathbb{Q}}\left[f(a,u)\right]\leq\mathrm{Lip}(f)\big(\varepsilon+\mathbb{E}_{(a,u)\sim\nu}\left[f(a,u)\right]\big). \tag{B.3}$$

It remains to compute the Lipschitz constant of f. Let $(a_1,u_1),(a_2,u_2)\in\mathcal{X}\times\mathcal{Y}$ and note that

$$\begin{aligned}\left|f(a_1,u_1)-f(a_2,u_2)\right|\leq&L_\ell\left(\|\mathcal{G}(a_1)-\mathcal{G}(a_2)\|_{\mathcal{Y}}^2+\|u_1-u_2\|_{\mathcal{Y}}^2\right)^{1/2} &\text{(B.4)}\\ \leq&L_\ell\left(\mathrm{Lip}(\mathcal{G})^2\|a_1-a_2\|_{\mathcal{X}}^2+1\,\|u_1-u_2\|_{\mathcal{Y}}^2\right)^{1/2}\\ \leq&L_\ell\left(\max\{\mathrm{Lip}(\mathcal{G})^2,1\}\,\|a_1-a_2\|_{\mathcal{X}}^2+\max\{\mathrm{Lip}(\mathcal{G})^2,1\}\,\|u_1-u_2\|_{\mathcal{Y}}^2\right)^{1/2}\\ =&L_\ell\,\max\{\mathrm{Lip}(\mathcal{G}),1\}\left(\|a_1-a_2\|_{\mathcal{X}}^2+\|u_1-u_2\|_{\mathcal{Y}}^2\right)^{1/2}\\ :=&L_\ell\,\max\{\mathrm{Lip}(\mathcal{G}),1\}\|(a_1,u_1)-(a_2,u_2)\|_{\mathcal{X}\times\mathcal{Y}}^2, &\text{(B.5)}\end{aligned}$$

where the right-hand side of (B.4) follows from definition of the 2-product metric on $\mathcal{X}\times\mathcal{Y}$. Incorporating the estimate of $\mathrm{Lip}(f)$ computed in (B.4)-(B.5) into (B.3) completes the proof. □

B.2. Proof of Lemma 7.9

Proof. Arguing as in [101, Lemma 5.2], we see that

$$\mathcal{W}_1\left(\mu,\mu^N\right)\leq\mathcal{W}_1\left((\mathbf{I_d}\times\mathcal{G}^\star)_\#\mu_X,(\mathbf{I_d}\times\mathcal{G}^\star)_\#\mu_X^N\right). \tag{B.6}$$

Arguing analogously to (B.4)- (B.5) we find that $\mathbf{I_d}\times\mathcal{G}^\star$ is $\max\{1,L^\star\}$-Lipschitz. Therefore, the Kantorovich-Rubinstein duality [110, Theorem 5.10] and the estimate (B.6) imply that

$$\begin{aligned}\mathcal{W}_1\left(\mu,\mu^N\right)\leq&\mathcal{W}_1\left((\mathbf{I_d}\times\mathcal{G}^\star)_\#\mu_X,(\mathbf{I_d}\times\mathcal{G}^\star)_\#\mu_X^N\right)\\ \leq&\mathrm{Lip}(I_{\mathcal{X}\times\mathcal{G}^\star})\,\mathcal{W}_1(\mu_X,\mu_X^N)\\ \leq&\max\{1,L^\star\}\,\mathcal{W}_1(\mu_X,\mu_X^N).\end{aligned}$$

This completes the proof. □

B.3. Proof of Lemma 7.10

Proof. Reduction to estimating the concentration of the empirical Sampling measure μ_X^N to μ_X: By Lemma 7.9, we have

$$\mathcal{W}_1(\nu,\mu):=\mathcal{W}_1(\mu^N,\mu)\leq\max\{1,L^\star\}\,\mathcal{W}_1(\mu_X,\mu_X^N). \tag{B.7}$$

Set $\nu:=\mu^N$, in the notation of (25). Applying Lemma 7.8 yields

$$\begin{aligned}\mathbb{E}_{(a,u)\sim\mathbb{Q}}\left[\ell(f(a),u)\right]\leq&\mathrm{L}_\ell\max\{1,L\}\left(\varepsilon+\mathcal{W}_1(\nu,\mu)+\mathbb{E}_{(a,u)\sim\mu^N}\left[\ell(f(a),u)\right]\right)\\ \leq&\mathrm{L}_\ell\max\{1,L\}\left(\varepsilon+\max\{1,L^\star\}\,\mathcal{W}_1(\mu_X,\mu_X^N)+\mathbb{E}_{(a,u)\sim\mu^N}\left[\ell(f(a),u)\right]\right),\end{aligned} \tag{B.8}$$

for each $f\in\mathcal{F}$ (for each $\omega\in\Omega$).

Applying the sampling estimates for μ_X: Under our assumptions on the small ball function ψ, [14, Theorem 1.4] implies that there exists a constant $C_\mu>0$, depending only on μ_X, such that for every $\eta>0$

$$\mathcal{W}_2\left(\mu_X^N,\mu_X\right)\leq(C_\mu+\eta)\,\psi^{-1}(\log(N)), \tag{B.9}$$

holds with probability at-least $1-\exp(-N\,(\psi^{-1}(\log(N)))^2\,\frac{\lambda^2}{2\Sigma^2})$. Here, we have denoted by $\mathcal{W}_2(\mu_X^N,\mu_X)$ the Wasserstein distance of the order two that measures the distance between two distributions μ_X^N and μ_X.

Set, $\eta:=-\log(\delta)^{1/2}\,2^{1/2}\Sigma/(N^{1/2}\psi^{-1}(\log(N)))$, then (B.9) implies that

$$\mathcal{W}_1\left(\mu_X^N,\mu_X\right)\leq\mathcal{W}_2\left(\mu_X^N,\mu_X\right)\leq C_\mu\psi^{-1}\big(\log(N)\big)+\Sigma\frac{\sqrt{-2\log(\delta)}}{\sqrt{N}}, \tag{B.10}$$

holds with probability at-least $1-\delta$; where we used the fact that $\mathcal{W}_1\leq\mathcal{W}_2$ (see e.g. [110, Remark 6.6]) to deduce the left-hand side of (B.9). Combining (28) with (B.10) implies that: for every $0<\delta\leq1$ and each $\mathcal{G}\in\mathscr{G}$ we have

$$\mathbb{E}_{(a,u)\sim\mathbb{Q}}\left[\ell(\mathcal{G}(a),u)\right]-\bar{L}\,\mathbb{E}_{(a,u)\sim\mu^N}\left[\ell(\mathcal{G}(a),u)\right]\leq\bar{L}\left(\varepsilon+C_\mu\psi^{-1}\big(\log(N)\big)+\frac{\Sigma\sqrt{-2\log(\delta)}}{\sqrt{N}}\right), \tag{B.11}$$

holds with probability at-least $1-\delta$; where $\bar{L} := L_\ell \max\{1, L\} \max\{1, L^\star\}$. Since the right-hand side of (B.11) was in-dependant of $\mathcal{G}$, then taking the supremum over the class $\mathscr{G}$ on both sides of (B.11) yields the conclusion.

Finally, if $H_\mu(r) \in \Theta\big(\frac{\log(1/r)^{2\beta/(2+\alpha)}}{r^{2\alpha/(2+\alpha)}}\big)$ then [80, Theorem 1.2] implies that $\psi(\eta) \in \Theta\left(\frac{\log(1/\eta)^\beta}{\eta^\alpha}\right)$. □

Appendix C. Proofs for Section 8

C.1. Proof of Lemma 8.3

Proof. By using (35), we have for $f = \ell(\mathcal{G}(\cdot), \cdot) \in \mathscr{F}_{\mathscr{G}}$ and $\mathcal{G} \in \mathscr{G}$,

$$\begin{aligned} |f(a,u)| &\le |\ell(\mathcal{G}(a),u) - \ell(0,u)| + |\ell(0,u)| \\ &\le \rho \|\mathcal{G}(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})} + R_u \le \rho R + R_u, \end{aligned} \tag{C.1}$$

for $(a,u) \in \mathrm{L}^2(D;\mathbb{R}^{d_a}) \times \mathrm{L}^2(D;\mathbb{R}^{d_u})$, where (C.1) followed from Assumption 8.2 (i) and (ii). This implies that by employing Wainwright [111, Theorem 4.10] or Giné and Nickl [41, Theorem 3.4.5], we have the following inequality with probability greater than $1-2e^{-\delta}$,

$$\begin{aligned} |\mathcal{L}(\mathcal{G}) - \widehat{\mathcal{L}}_{\mathrm{S}}(\mathcal{G})| &\le \sup_{f \in \mathscr{F}_{\mathscr{G}}} \left| \frac{1}{n} \sum_{i=1}^n f(a_i,u_i) - \mathbb{E}_{(a,u)\sim\mu}[f(a,u)] \right| \\ &\le 2\mathbb{E}_\epsilon \left[\sup_{f \in \mathscr{F}_{\mathscr{G}}} \frac{1}{n} \left| \sum_{i=1}^n \epsilon_i f(a_i,u_i) \right| \right] + (\rho R + R_u)\sqrt{\frac{2\delta}{n}}, \ \mathcal{G} \in \mathscr{G}, \end{aligned}$$

where $\{\epsilon_i\}_{i=1}^n$ is a sequence of i.i.d. RV's with Rademacher distribution; i.e., $\mathbf{P}\{\epsilon_i = 1\} = 1/2 = \mathbf{P}\{\epsilon_i = -1\}$. □

C.2. Proof of Theorem 8.4

Proof. By employing Bartlett et al. [8, Lemma A.5] or Kakade and Tewari [59, Theorem 1.1], we have

$$\mathfrak{R}_{\mathrm{S}}^n(\mathscr{F}_{\mathscr{N}}) \le \inf_{\alpha \ge 0} \left\{ 4\alpha + \frac{12}{\sqrt{n}} \int_\alpha^\infty \left(\log N(\varepsilon, \mathscr{F}_{\mathscr{N}}, \|\cdot\|_{\mathrm{S}}) \right)^{\frac{1}{2}} d\varepsilon \right\}, \tag{C.2}$$

where $\|f\|_{\mathrm{S}} := \left(\frac{1}{n} \sum_{i=1}^n f(a_i,u_i)^2 \right)^{\frac{1}{2}}$. Here, we denote by $N(\varepsilon, \mathscr{F}, \|\cdot\|)$ the covering number of the function space $\mathscr{F}$ which means the minimal cardinality of a subset $C \subset \mathscr{F}$ that covers $\mathscr{F}$ at scale ε with respect to the norm $\|\cdot\|$. In the following, we will estimate the covering number $N(\varepsilon, \mathscr{F}_{\mathscr{N}}, \|\cdot\|_{\mathrm{S}})$.

Let $f = \ell(\mathcal{G}(\cdot), \cdot)$ and $f' = \ell(\mathcal{G}'(\cdot), \cdot)$ where $\mathcal{G}, \mathcal{G}' \in \mathscr{N}$. By (i) of Assumption 8.2, we calculate

$$|f(a,u) - f'(a,u)| = |\ell(\mathcal{G}(a),u) - \ell(\mathcal{G}'(a),u)| \le \rho \left\| \mathcal{G}(a) - \mathcal{G}'(a) \right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})}. \tag{C.3}$$

Denoting by

$$\begin{aligned} \mathcal{G}_\ell &:= (W_\ell + \mathcal{K}_\ell) \circ \sigma(W_{\ell-1} + \mathcal{K}_{\ell-1}) \circ \cdots \circ \sigma(W_0 + \mathcal{K}_0), \\ \mathcal{G}'_\ell &:= (W'_\ell + \mathcal{K}'_\ell) \circ \sigma(W'_{\ell-1} + \mathcal{K}'_{\ell-1}) \circ \cdots \circ \sigma(W'_0 + \mathcal{K}'_0), \end{aligned}$$

the quantity $\left\| \mathcal{G}(a) - \mathcal{G}'(a) \right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})}$ is evaluated by

$$\begin{aligned} &\left\| \mathcal{G}(a) - \mathcal{G}'(a) \right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})} = \left\| \mathcal{G}_L(a) - \mathcal{G}'_L(a) \right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_{L+1}})} \\ &= \Big\| (W_L + \mathcal{K}_L) \circ \sigma(\mathcal{G}_{L-1}(a)) - (W_L + \mathcal{K}_L) \circ \sigma(\mathcal{G}'_{L-1}(a)) \\ &\qquad + (W_L + \mathcal{K}_L) \circ \sigma(\mathcal{G}'_{L-1}(a)) - (W'_L + \mathcal{K}'_L) \circ \sigma(\mathcal{G}'_{L-1}(a)) \Big\|_{\mathrm{L}^2(D;\mathbb{R}^{d_{L+1}})} \\ &\qquad \text{Assumption 7.3(vi)} \\ &\le \Bigg(\underbrace{\|W_L\|_{\mathrm{op}}}_{\le C_w \ (\text{C.5})} + \underbrace{\|\mathcal{K}_L\|_{\mathrm{op}}}_{\le C_k \ (\text{C.6})} \Bigg) C_\sigma \left\| \mathcal{G}_{L-1}(a) - \mathcal{G}'_{L-1}(a) \right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_L})} \\ &\qquad + \left(\left\| W_L - W'_L \right\|_{\mathrm{op}} + \left\| \mathcal{K}_L - \mathcal{K}'_L \right\|_{\mathrm{op}} \right) C_\sigma \left\| \mathcal{G}'_{L-1}(a) \right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_L})}, \end{aligned} \tag{C.4}$$

where $\|\cdot\|_{\mathrm{op}}$ is the Operator norm. Here, we have employed the following estimations:

$$\|W_L g\|^2_{\mathrm{L}^2(D;\mathbb{R}^{d_{L+1}})} \le \int_D \underbrace{\|W_L g(x)\|_2^2}_{\le \|W_L\|_{\mathrm{F}}^2 \|g(x)\|_2^2} dx \underset{\text{Assumption 7.3}(i)}{\le} C_w^2 \|g\|^2_{\mathrm{L}^2(D;\mathbb{R}^{d_L})}, \tag{C.5}$$

$$\begin{aligned}\|\mathcal{K}_L g\|^2_{\mathrm{L}^2(D;\mathbb{R}^{d_{L+1}})} &\le \int_D \underbrace{\left\| \int_D \mathcal{K}_L(x,y) g(y) dy \right\|_2^2}_{\le \left(\sum_{i,j} \left\|k_{L,ij}(x,\cdot)\right\|^2_{\mathrm{L}^2(D)}\right) \|g\|^2_{\mathrm{L}^2(D;\mathbb{R}^{d_L})}} dx \le \|\mathcal{K}_L\|^2_{\mathrm{L}^2,\mathrm{F}} \|g\|^2_{\mathrm{L}^2(D;\mathbb{R}^{d_L})} \\ &\underset{\text{Assumption 7.3}(ii)}{\le} C_k^2 \|g\|^2_{\mathrm{L}^2(D;\mathbb{R}^{d_L})},\end{aligned} \tag{C.6}$$

for $g \in \mathrm{L}^2(D;\mathbb{R}^{d_L})$, where $\|\cdot\|_2$ is the ℓ_2-norm. By the same argument in (C.4)–(C.6), we evaluate

$$\begin{aligned}&\left\|\mathcal{G}_{L-1}(a) - \mathcal{G}'_{L-1}(a)\right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_L})} \\ &\le (C_w + C_k) C_\sigma \left\|\mathcal{G}_{L-2}(a) - \mathcal{G}'_{L-2}(a)\right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_{L-1}})} \\ &\quad + \left(\left\|W_{L-1} - W'_{L-1}\right\|_{\mathrm{op}} + \left\|\mathcal{K}_{L-1} - \mathcal{K}'_{L-1}\right\|_{\mathrm{op}} \right) C_\sigma \left\|\mathcal{G}'_{L-2}(a)\right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_{L-1}})}.\end{aligned} \tag{C.7}$$

By repeatedly evaluating $\left\|\mathcal{G}_\ell(a) - \mathcal{G}'_\ell(a)\right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_{\ell+1}})}$ $(\ell = L, L-1, ..., 0)$, we obtain

$$\begin{aligned}&\left\|\mathcal{G}(a) - \mathcal{G}'(a)\right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})} \\ &\le \left\{(C_w + C_k) C_\sigma\right\}^L \underbrace{\left\|(W_0 + \mathcal{K}_0)(a) - (W'_0 + \mathcal{K}'_0)(a)\right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_1})}}_{\text{Assumption 7.3}(iii) \le C_a\left(\left\|W_0 - W'_0\right\|_{\mathrm{op}} + \left\|\mathcal{K}_0 - \mathcal{K}'_0\right\|_{\mathrm{op}}\right)} \\ &\quad + \sum_{\ell=0}^{L-1} \left(\left\|W_{\ell+1} - W'_{\ell+1}\right\|_{\mathrm{op}} + \left\|\mathcal{K}_{\ell+1} - \mathcal{K}'_{\ell+1}\right\|_{\mathrm{op}} \right) \\ &\qquad \times \underbrace{(C_w + C_k)^{L-1-\ell} C_\sigma^{L-\ell} \left\|\mathcal{G}'_\ell(a)\right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_{\ell+1}})}}_{(\mathrm{C.9}) \le \left\{(C_w + C_k) C_\sigma\right\}^L C_a} \\ &\le \left\{(C_w + C_k) C_\sigma\right\}^L C_a \sum_{\ell=0}^{L} \left(\left\|W_\ell - W'_\ell\right\|_{\mathrm{op}} + \left\|\mathcal{K}_\ell - \mathcal{K}'_\ell\right\|_{\mathrm{op}} \right).\end{aligned} \tag{C.8}$$

Here, we have employed the following estimation:

$$\left\|\mathcal{G}'_\ell\right\|^2_{\mathrm{L}^2(D;\mathbb{R}^{d_{\ell+1}})} \le (C_w + C_k)^{\ell+1} C_\sigma^\ell C_a. \tag{C.9}$$

Furthermore, by using ideas of (C.5) and (C.6), we estimate

$$\begin{aligned}\left\|W_\ell - W'_\ell\right\|_{\mathrm{op}} &\le \left\|W_\ell - W'_\ell\right\|_{\mathrm{F}} \\ &\le \sum_{j=1}^{d_{\ell+1}} \sum_{i=1}^{d_\ell} |w_{\ell,ij} - w'_{\ell,ij}| \le \sum_{j=1}^{d_{\ell+1}} \sum_{i=1}^{d_\ell} C_w \left| \frac{w_{\ell,ij}}{C_w} - \frac{w'_{\ell,ij}}{C_w} \right|,\end{aligned} \tag{C.10}$$

$$\left\|\mathcal{K}_\ell - \mathcal{K}'_\ell\right\|_{\mathrm{op}} \le \sum_{j=1}^{d_{\ell+1}} \sum_{i=1}^{d_\ell} |D| \left\|k_{\ell,ij} - k'_{\ell,ij}\right\|_{\mathbf{L}^\infty(D\times D;\mathbb{R})} \le \sum_{j=1}^{d_{\ell+1}} \sum_{i=1}^{d_\ell} |D| C_\alpha \left\| \frac{k_{\ell,ij}}{C_\alpha} - \frac{k'_{\ell,ij}}{C_\alpha} \right\|_{\mathbf{L}^\infty(D\times D;\mathbb{R})}. \tag{C.11}$$

Combining (C.3), (C.8), (C.10), and (C.11), the norm $\|f - f'\|_{\mathrm{S}}$ is estimated by

$$\begin{aligned}\|f - f'\|_{\mathrm{S}} &= \left(\frac{1}{n} \sum_{i=1}^{n} |f(a_i,u_i) - f'(a_i,u_i)|^2 \right)^{\frac{1}{2}} \\ &\le \sum_{\ell=0}^{L} \sum_{j=1}^{d_{\ell+1}} \sum_{i=1}^{d_\ell} \left[\rho \left\{(C_w + C_k) C_\sigma\right\}^L C_a C_w \left| \frac{w_{\ell,ij}}{C_w} - \frac{w'_{\ell,ij}}{C_w} \right| \right. \\ &\qquad \left. + \rho \left\{(C_w + C_k) C_\sigma\right\}^L C_a |D| C_\alpha \left\| \frac{k_{\ell,ij}}{C_\alpha} - \frac{k'_{\ell,ij}}{C_\alpha} \right\|_{\mathbf{L}^\infty(D\times D;\mathbb{R})} \right],\end{aligned}$$

which implies that we have

$$\begin{aligned} &N(\varepsilon, \mathscr{F}_{\mathscr{N}}, \|\cdot\|_{\mathrm{S}}) \\ &\leq \prod_{\ell=0}^{L} \prod_{j=1}^{d_{\ell+1}} \prod_{i=1}^{d_\ell} N\left(\frac{\varepsilon}{2\left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right) \rho\left\{(C_w + C_k) C_\sigma\right\}^L C_a C_w}, [-1,1], |\cdot| \right) \\ &\times N\left(\frac{\varepsilon}{2\left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right) \rho\left\{(C_w + C_k) C_\sigma\right\}^L C_a |D| C_\alpha}, \mathcal{F}_{C_\beta}, \|\cdot\|_{\mathbf{L}^\infty(D\times D;\mathbb{R})} \right), \end{aligned} \tag{C.12}$$

where $\mathcal{F}_{C_\beta} := \{k : D \times D \to [-1,1] \mid k \text{ is } C_\beta - Lipschitz\}$ (see (vi) in Assumption 7.3).

By taking logarithmic functions in (C.12), we have

$$\begin{aligned} &\log N(\varepsilon, \mathscr{F}_{\mathscr{N}}, \|\cdot\|_{\mathrm{S}}) \\ &\leq \left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right) \Bigg\{ \underbrace{\log N\left(\frac{\varepsilon}{2\left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right) \rho\left\{(C_w + C_k) C_\sigma\right\}^L C_a C_w}, [-1,1], |\cdot| \right)}_{=: H_w(\varepsilon)} \\ &\quad + \underbrace{\log N\left(\frac{\varepsilon}{2\left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right) \rho\left\{(C_w + C_k) C_\sigma\right\}^L C_a |D| C_\alpha}, \mathcal{F}_{C_\beta}, \|\cdot\|_{\mathbf{L}^\infty(D\times D;\mathbb{R})} \right)}_{=: H_k(\varepsilon)} \Bigg\}. \end{aligned} \tag{C.13}$$

By using Wainwright [111, Example 5.3] and Gottlieb et al. [44, Lemmas 2.1 and 4.2], we estimate H_w and H_k, respectively as follows:

$$\begin{aligned} H_w(\varepsilon) &\leq \log\left(1 + \frac{2\left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right) \rho\left\{(C_w + C_k) C_\sigma\right\}^L C_a C_w}{\varepsilon} \right) \\ &\leq \left(\frac{I_w}{\varepsilon}\right) \leq \left(\frac{I_w}{\varepsilon}\right)^{\hat{d}+1}, \quad \text{for } 0 < \varepsilon < 2\left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right) \rho\left\{(C_w + C_k) C_\sigma\right\}^L C_a C_w, \end{aligned} \tag{C.14}$$

$$\begin{aligned} H_k(\varepsilon) &\leq \left(\frac{8 C_\beta \mathrm{diag}(D\times D)\left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right) \rho\left\{(C_w + C_k) C_\sigma\right\}^L C_a |D| C_\alpha}{\varepsilon} \right)^{\hat{d}} \\ &\times \log\left(\frac{16\left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right) \rho\left\{(C_w + C_k) C_\sigma\right\}^L C_a |D| C_\alpha}{\varepsilon} \right) \\ &\leq \left(\frac{I_k}{\varepsilon}\right)^{\hat{d}+1}, \quad \text{for } 0 < \varepsilon < 2\left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right) \rho\left\{(C_w + C_k) C_\sigma\right\}^L C_a |D| C_\alpha, \end{aligned} \tag{C.15}$$

where we denoted by

$$I_w := 2\left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right) \rho\left\{(C_w + C_k) C_\sigma\right\}^L C_a C_w,$$

$$I_k := 8 \max\left\{C_\beta \mathrm{diag}(D\times D), 2\right\}\left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right) \rho\left\{(C_w + C_k) C_\sigma\right\}^L C_a |D| C_\alpha.$$

By employing (C.13), (C.14), and (C.15), we calculate

$$\int_\alpha^\infty \left(\log N(\varepsilon, \mathscr{F}_{\mathscr{N}}, \|\cdot\|_{\mathrm{S}})\right)^{\frac{1}{2}} d\varepsilon \leq \left(\sum_{\ell=0}^{L} d_\ell d_{\ell+1}\right)^{\frac{1}{2}} \underbrace{\int_\alpha^\infty \left(H_w(\varepsilon) + H_k(\varepsilon)\right)^{\frac{1}{2}} d\varepsilon}_{=: (*)}$$

$$
\begin{aligned}
(*) &\le \int_\alpha^\infty H_w(\varepsilon)^{\frac{1}{2}} d\varepsilon + \int_\alpha^\infty H_w(\varepsilon)^{\frac{1}{2}} d\varepsilon \\
&\le \int_\alpha^{2\left(\sum_{\ell=0}^L d_\ell d_{\ell+1}\right)\rho\{(C_w+C_k)C_\sigma\}^L C_a C_w} \left(\frac{I_w}{\varepsilon}\right)^{\frac{\hat{d}+1}{2}} d\varepsilon \\
&\qquad + \int_\alpha^{2\left(\sum_{\ell=0}^L d_\ell d_{\ell+1}\right)\rho\{(C_w+C_k)C_\sigma\}^L C_a |D| C_\alpha} \left(\frac{I_k}{\varepsilon}\right)^{\frac{\hat{d}+1}{2}} d\varepsilon \\
&\le \left(I_w^{\frac{\hat{d}+1}{2}} + I_k^{\frac{\hat{d}+1}{2}}\right) \frac{2}{\hat{d}-1} \alpha^{-\frac{\hat{d}-1}{2}} \\
&\le \frac{4}{\hat{d}-1} \left(\max\left[2C_w, 8|D|C_\alpha \max\left\{C_\beta \mathrm{diag}(D\times D), 2\right\}\right] \left(\sum_{\ell=0}^L d_\ell d_{\ell+1}\right) \rho\left\{(C_w+C_k)C_\sigma\right\}^L C_a\right)^{\frac{\hat{d}+1}{2}} \alpha^{-\frac{\hat{d}-1}{2}},
\end{aligned}
$$

that is, we have by (i) of Assumption 7.3.

$$
\begin{aligned}
&\int_\alpha^\infty \left(\log N(\varepsilon, \mathscr{F}_{\mathscr{N}}, \|\cdot\|_{\mathrm{S}})\right)^{\frac{1}{2}} d\varepsilon \\
&\le \underbrace{\frac{4}{\hat{d}-1} \left(\max\left[2C_w, 8|D|C_\alpha \max\left\{C_\beta \mathrm{diag}(D\times D), 2\right\}\right] (LC_d^2)^{\frac{\hat{d}+2}{\hat{d}+1}} \rho\left\{(C_w+C_k)C_\sigma\right\}^L C_a\right)^{\frac{\hat{d}+1}{2}}}_{=:K} \alpha^{-\frac{\hat{d}-1}{2}}
\end{aligned}
$$

which implies that we conclude that with (C.2)

$$
\begin{aligned}
\mathfrak{R}_{\mathrm{S}}^n(\mathscr{F}_{\mathscr{N}}) &\le 4 \inf_{\alpha\ge 0} \left\{ \alpha + \underbrace{\frac{3K}{\sqrt{n}}}_{=:K'} \alpha^{-\frac{\hat{d}-1}{2}} \right\} \\
&= 4\left(\left(\frac{(\hat{d}-1)K'}{2}\right)^{\frac{2}{\hat{d}+1}} + K' \left(\frac{(\hat{d}-1)K'}{2}\right)^{\frac{2}{\hat{d}+1}\left(-\frac{\hat{d}-1}{2}\right)} \right) = \gamma L^{\frac{\hat{d}+2}{\hat{d}+1}} \left\{(C_w+C_k)C_\sigma\right\}^L \left(\frac{1}{n}\right)^{\frac{1}{\hat{d}+1}}
\end{aligned}
$$

where γ is the positive constant defined by

$$
\begin{aligned}
\gamma := 4 &\left\{ \left(\frac{\hat{d}-1}{2}\right)^{\frac{2}{\hat{d}+1}} + \left(\frac{\hat{d}-1}{2}\right)^{-\frac{\hat{d}-1}{\hat{d}+1}} \right\} \left(\frac{12}{\hat{d}-1}\right)^{\frac{2}{\hat{d}+1}} \\
&\times \max\left[2C_w, 8|D|C_\alpha \max\left\{C_\beta \mathrm{diag}(D\times D), 2\right\}\right] C_d^{\frac{2(\hat{d}+2)}{\hat{d}+1}} \rho C_a. \quad \square
\end{aligned} \tag{C.16}
$$

C.3. Proof of Theorem 8.5

Proof. The following argument is almost same with the proof of Theorem 8.4.

By employing Bartlett et al. [8, Lemma A.5] or Kakade and Tewari [59, Theorem 1.1], we have

$$
\mathfrak{R}_{\mathrm{S}}^n(\mathscr{F}_{\widetilde{\mathscr{N}}}) \le \inf_{\alpha\ge 0} \left\{ 4\alpha + \frac{12}{\sqrt{n}} \int_\alpha^\infty \left(\log N(\varepsilon, \mathscr{F}_{\widetilde{\mathscr{N}}}, \|\cdot\|_{\mathrm{S}})\right)^{\frac{1}{2}} d\varepsilon \right\} \tag{C.17}
$$

In the following, we will estimate the covering number $N(\varepsilon, \mathscr{F}_{\widetilde{\mathscr{N}}}, \|\cdot\|_{\mathrm{S}})$.

Let $f = \ell(\mathcal{G}(\cdot), \cdot)$ and $f' = \ell(\mathcal{G}'(\cdot), \cdot)$ where $\mathcal{G}, \mathcal{G}' \in \widetilde{\mathscr{N}}$. By (i) of Assumption 8.2, we calculate

$$
\left|f(a,u) - f'(a,u)\right| = \left|\ell(\mathcal{G}(a),u) - \ell(\mathcal{G}'(a),u)\right| \le \rho \left\|\mathcal{G}(a) - \mathcal{G}'(a)\right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})}. \tag{C.18}
$$

Denoting by

$$\mathcal{G}_\ell := (\mathbf{Z}_\ell \mathbf{I_d} + \mathbf{X}_\ell f_\ell) \circ (\mathbf{Z}_\ell \mathbf{I_d} + \mathbf{X}_\ell \sigma \circ \boldsymbol{\mathcal{K}}_\ell) \circ \cdots \circ (\mathbf{Z}_0 \mathbf{I_d} + \mathbf{X}_0 f_0) \circ (\mathbf{Z}_0 \mathbf{I_d} + \mathbf{X}_0 \sigma \circ \boldsymbol{\mathcal{K}}_0),$$

$$\mathcal{G}'_\ell := (\mathbf{Z}_\ell \mathbf{I_d} + \mathbf{X}_\ell f'_\ell) \circ (Z_\ell \mathbf{I_d} + \mathbf{X}_\ell \sigma \circ \boldsymbol{\mathcal{K}}'_\ell) \circ \cdots \circ (\mathbf{Z}_0 \mathbf{I_d} + \mathbf{X}_0 f'_0) \circ (Z_0 \mathbf{I_d} + \mathbf{X}_0 \sigma \circ \boldsymbol{\mathcal{K}}'_0),$$

the quantity $\|\mathcal{G}(a) - \mathcal{G}'(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})}$ is evaluated by

$$\begin{aligned}
&\|\mathcal{G}(a) - \mathcal{G}'(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})} = \|\mathcal{G}_L(a) - \mathcal{G}'_L(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_{L+1}})} \\
&= \Big\| (\mathbf{Z}_L \mathbf{I_d} + \mathbf{X}_L f_L) \circ (\mathbf{Z}_L \mathbf{I_d} + \mathbf{X}_L \sigma \circ \boldsymbol{\mathcal{K}}_L)(\mathcal{G}_{L-1}(a)) - (\mathbf{Z}_L \mathbf{I_d} + \mathbf{X}_L f_L) \circ (\mathbf{Z}_L \mathbf{I_d} + \mathbf{X}_L \sigma \circ \boldsymbol{\mathcal{K}}_L)(\mathcal{G}'_{L-1}(a)) \\
&\quad + (\mathbf{Z}_L \mathbf{I_d} + \mathbf{X}_L f_L) \circ (\mathbf{Z}_L \mathbf{I_d} + \mathbf{X}_L \sigma \circ \boldsymbol{\mathcal{K}}_L)(\mathcal{G}'_{L-1}(a)) - (\mathbf{Z}_L \mathbf{I_d} + \mathbf{X}_L f'_L) \circ (\mathbf{Z}_L \mathbf{I_d} + \mathbf{X}_L \sigma \circ \boldsymbol{\mathcal{K}}'_L)(\mathcal{G}'_{L-1}(a)) \Big\| \\
&\quad \text{Assumption 7.4}(i)(ii)(iv) \\
&\le (\mathbf{Z}_L + \mathbf{X}_L C_w^{M+1} C_\sigma^M)(\mathbf{Z}_L + \mathbf{X}_L C_k C_\sigma) \|\mathcal{G}_{L-1}(a) - \mathcal{G}'_{L-1}(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_L})} \\
&\quad + \Big((\mathbf{Z}_L + \mathbf{X}_L C_\sigma C_k) \mathbf{X}_L \|f_L - f'_L\|_{\mathrm{op}} + (\mathbf{Z}_L + \mathbf{X}_L C_w^{M+1} C_\sigma^M) \mathbf{X}_L \|\boldsymbol{\mathcal{K}}_L - \boldsymbol{\mathcal{K}}'_L\|_{\mathrm{op}} \Big) \|\mathcal{G}'_{L-1}(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_L})}.
\end{aligned} \tag{C.19}$$

Here, we have employed the following estimation:

$$\|f_L\|_{\mathrm{op}} = \|W_{L,M} \circ \sigma(W_{L,M-1}) \circ \cdots \circ \sigma(W_{L,1}) \circ \sigma(W_{L,0})\|_{\mathrm{op}} \le C_w^{M+1} C_\sigma^M. \tag{C.20}$$

By the same argument in (C.19)–(C.20), we evaluate

$$\begin{aligned}
&\|\mathcal{G}_{L-1}(a) - \mathcal{G}'_{L-1}(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_L})} \\
&\le (\mathbf{Z}_{L-1} + \mathbf{X}_{L-1} C_w^{M+1} C_\sigma^M)(\mathbf{Z}_{L-1} + \mathbf{X}_{L-1} C_k C_\sigma) \|\mathcal{G}_{L-2}(a) - \mathcal{G}'_{L-2}(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_{L-1}})} \\
&+ \Bigg((\mathbf{Z}_{L-1} + \mathbf{X}_{L-1} C_\sigma C_k) \mathbf{X}_{L-1} \|f_{L-1} - f'_{L-1}\|_{\mathrm{op}} \\
&\qquad + (\mathbf{Z}_{L-1} + \mathbf{X}_{L-1} C_w^{M+1} C_\sigma^M) \mathbf{X}_{L-1} \|\boldsymbol{\mathcal{K}}_{L-1} - \boldsymbol{\mathcal{K}}'_{L-1}\|_{\mathrm{op}} \Bigg) \|\mathcal{G}'_{L-2}(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_{L-1}})}.
\end{aligned} \tag{C.21}$$

By repeatedly evaluating $\|\mathcal{G}_\ell(a) - \mathcal{G}'_\ell(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_{\ell+1}})}$ $(\ell = L, L-1, ..., 0)$, we obtain

$$\begin{aligned}
&\|\mathcal{G}(a) - \mathcal{G}'(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})} \\
&\le C_a \prod_{\ell=0}^{L} (\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma) \\
&\times \sum_{\ell=0}^{L} \left(\frac{\mathbf{X}_\ell}{\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M} \|f_{L-1} - f'_{L-1}\|_{\mathrm{op}} + \frac{\mathbf{X}_\ell}{\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma} \|\boldsymbol{\mathcal{K}}_\ell - \boldsymbol{\mathcal{K}}'_\ell\|_{\mathrm{op}} \right). \\
&\le C_a \underbrace{\prod_{\ell=0}^{L} (\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma)}_{=: T_L} \\
&\times \sum_{\ell=0}^{L} \left(\underbrace{\frac{\mathbf{X}_\ell C_w^{M+1} C_\sigma^M}{\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M}}_{=: C_{w,\ell}} \sum_{m=0}^{M} \|W_{\ell,m} - W'_{\ell,m}\|_{\mathrm{op}} + \underbrace{\frac{\mathbf{X}_\ell}{\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma}}_{=: C_{k,\ell}} \|\boldsymbol{\mathcal{K}}_\ell - \boldsymbol{\mathcal{K}}'_\ell\|_{\mathrm{op}} \right).
\end{aligned} \tag{C.22}$$

Combining (C.18), (C.22), (C.10), and (C.11), the norm $\|f - f'\|_{\mathrm{S}}$ is estimated by

$$\begin{aligned}
&\|f - f'\|_{\mathrm{S}} = \left(\frac{1}{n} \sum_{i=1}^{n} |f(a_i, u_i) - f'(a_i, u_i)|^2 \right)^{\frac{1}{2}} \\
&\le \sum_{\ell=0}^{L} \Bigg[\sum_{m=0}^{M} \sum_{j=1}^{d^w_{\ell,m+1}} \sum_{i=1}^{d^w_{\ell,m}} \rho C_a T_L C_{w,\ell} C_w \left| \frac{w_{\ell,m,ij}}{C_w} - \frac{w'_{\ell,m,ij}}{C_w} \right| \\
&\qquad + \sum_{j=1}^{d^k_\ell} \sum_{i=1}^{d^k_\ell} \rho C_a T_L C_{k,\ell} |D| C_\alpha \left\| \frac{k_{\ell,ij}}{C_\alpha} - \frac{k'_{\ell,ij}}{C_\alpha} \right\|_{\mathbf{L}^\infty(D \times D;\mathbb{R})} \Bigg],
\end{aligned}$$

which implies that we have

$$\begin{aligned}
&N(\varepsilon, \mathscr{F}_{\widetilde{\mathscr{N}}}, \|\cdot\|_S) \\
&\leq \prod_{\ell=0}^{L} \prod_{j=1}^{d_{\ell+1}} \prod_{m=0}^{M} \prod_{j=1}^{d^w_{\ell,m+1}} \prod_{i=1}^{d^w_{\ell,m}} \prod_{j'=1}^{d^k_{\ell+1}} \prod_{i'=1}^{d^k_{\ell}} \\
&\times N\left(\frac{\varepsilon}{2\left(\sum_{\ell=0}^{L}\sum_{m=0}^{M} d^w_{\ell,m+1} d^w_{\ell,m} C_{w,\ell}\right) \rho C_a T_L C_w}, [-1,1], |\cdot| \right) \\
&\times N\left(\frac{\varepsilon}{2\left(\sum_{\ell=0}^{L} d^k_{\ell+1} d^k_{\ell} C_{k,\ell}\right) \rho C_a T_L |D| C_\alpha}, \|\cdot\|_{\mathbf{L}^\infty(D\times D;\mathbb{R})} \right).
\end{aligned} \tag{C.23}$$

By taking logarithmic functions in (C.23), we have

$$\begin{aligned}
&\log N(\varepsilon, \mathscr{F}_{\widetilde{\mathscr{N}}}, \|\cdot\|_S) \\
&\leq \left(\sum_{\ell=0}^{L}\sum_{m=0}^{M} d^w_{\ell,m+1} d^w_{\ell,m} d^k_{\ell+1} d^k_{\ell} \right) \\
&\times \Bigg\{ \underbrace{\log N\left(\frac{\varepsilon}{2\left(\sum_{\ell=0}^{L}\sum_{m=0}^{M} d^w_{\ell,m+1} d^w_{\ell,m} C_{w,\ell}\right) \rho C_a T_L C_w}, [-1,1], |\cdot| \right)}_{=:\widetilde{H}_w(\varepsilon)} \\
&+ \underbrace{\log N\left(\frac{\varepsilon}{2\left(\sum_{\ell=0}^{L} d^k_{\ell+1} d^k_{\ell} C_{k,\ell}\right) \rho C_a T_L |D| C_\alpha}, \|\cdot\|_{\mathbf{L}^\infty(D\times D;\mathbb{R})} \right)}_{=:\widetilde{H}_k(\varepsilon)} \Bigg\}.
\end{aligned} \tag{C.24}$$

By same arguments in (C.14) and (C.15) (using Wainwright [111, Example 5.3] and Gottlieb et al. [44, Lemma 4.2]), we can estimate $\widetilde{H}_w$ and $\widetilde{H}_k$, respectively as follows:

$$\widetilde{H}_w(\varepsilon) \leq \left(\frac{\widetilde{I}_w}{\varepsilon} \right)^{\hat{d}+1}, \quad \text{for } 0 < \varepsilon < 2\left(\sum_{\ell=0}^{L}\sum_{m=0}^{M} d^w_{\ell,m+1} d^w_{\ell,m} C_{w,\ell} \right) \rho C_a T_L C_w, \tag{C.25}$$

$$\widetilde{H}_k(\varepsilon) \leq \left(\frac{\widetilde{I}_k}{\varepsilon} \right)^{\hat{d}+1}, \quad \text{for } 0 < \varepsilon < 2\left(\sum_{\ell=0}^{L} d^k_{\ell+1} d^k_{\ell} C_{k,\ell} \right) \rho C_a T_L |D| C_\alpha, \tag{C.26}$$

where we denoted by

$$\begin{aligned}
\widetilde{I}_w &:= 2\left(\sum_{\ell=0}^{L}\sum_{m=0}^{M} d^w_{\ell,m+1} d^w_{\ell,m} C_{w,\ell} \right) \rho C_a T_L C_w, \\
\widetilde{I}_k &:= 8 \max\left\{ C_\beta \mathrm{diag}(D\times D), 2 \right\} \left(\sum_{\ell=0}^{L} d^k_{\ell+1} d^k_{\ell} C_{k,\ell} \right) \rho C_a T_L |D| C_\alpha.
\end{aligned}$$

By employing (C.24), (C.25), and (C.26), we calculate

$$\int_\alpha^\infty \left(\log N(\varepsilon, \mathscr{F}_{\widetilde{\mathscr{N}}}, \|\cdot\|_S) \right)^{\frac{1}{2}} d\varepsilon \leq \left(\sum_{\ell=0}^{L}\sum_{m=0}^{M} d^w_{\ell,m+1} d^w_{\ell,m} d^k_{\ell+1} d^k_{\ell} \right)^{\frac{1}{2}} \underbrace{\int_\alpha^\infty (\widetilde{H}_w(\varepsilon) + \widetilde{H}_k(\varepsilon))^{\frac{1}{2}} d\varepsilon}_{=:(*)}$$

$$
\begin{aligned}
(*) &\le \int_{\alpha}^{\infty} \widetilde{H}_w(\varepsilon)^{\frac{1}{2}} d\varepsilon + \int_{\alpha}^{\infty} \widetilde{H}_w(\varepsilon)^{\frac{1}{2}} d\varepsilon \\
&\le \left(\widetilde{I}_w^{\frac{\hat{d}+1}{2}} + \widetilde{I}_k^{\frac{\hat{d}+1}{2}} \right) \frac{2}{\hat{d}-1} \alpha^{-\frac{\hat{d}-1}{2}} \\
&\le \frac{4}{\hat{d}-1} \left(\max\left[2C_w, 8|D|C_\alpha \max\left\{ C_\beta \mathrm{diag}(D\times D), 2 \right\} \right] \rho M C_d^2 C_a T_L \right)^{\frac{\hat{d}+1}{2}} \\
&\times \left[\left(\sum_{\ell=0}^{L} C_{w,\ell} \right)^{\frac{\hat{d}+1}{2}} + \left(\sum_{\ell=0}^{L} C_{k,\ell} \right)^{\frac{\hat{d}+1}{2}} \right] \alpha^{-\frac{\hat{d}-1}{2}},
\end{aligned}
$$

that is, we have

$$
\int_{\alpha}^{\infty} \left(\log N(\varepsilon, \mathscr{F}_{\widetilde{\mathscr{N}}}, \|\cdot\|_{\mathrm{S}}) \right)^{\frac{1}{2}} d\varepsilon \le \widetilde{K} \alpha^{-\frac{\hat{d}-1}{2}}
$$

where

$$
\widetilde{K} := \frac{4C_d^2 M^{1/2} L^{1/2}}{\hat{d}-1} \left(\max\left[2C_w, 8|D|C_\alpha \max\left\{ C_\beta \mathrm{diag}(D\times D), 2 \right\} \right] \rho M C_d^2 C_a T_L \left(\sum_{\ell=0}^{L} C_{w,\ell} + C_{k,\ell} \right) \right)^{\frac{\hat{d}+1}{2}} \alpha^{-\frac{\hat{d}-1}{2}}
$$

which implies that we conclude that with (C.17)

$$
\begin{aligned}
\mathfrak{R}_{\mathrm{S}}^n(\mathscr{F}_{\widetilde{\mathscr{N}}}) &\le 4 \inf_{\alpha \ge 0} \left\{ \alpha + \underbrace{\frac{3\widetilde{K}}{\sqrt{n}} \alpha^{-\frac{\hat{d}-1}{2}}}_{=:\widetilde{K}'} \right\} \\
&= 4 \left(\left(\frac{(\hat{d}-1)\widetilde{K}'}{2} \right)^{\frac{2}{\hat{d}+1}} + \widetilde{K}' \left(\frac{(\hat{d}-1)\widetilde{K}'}{2} \right)^{\frac{2}{\hat{d}+1}\left(-\frac{\hat{d}-1}{2}\right)} \right) \\
&= \widetilde{\gamma} L^{\frac{1}{\hat{d}+1}} \left(\sum_{\ell=0}^{L} \frac{\mathbf{X}_\ell C_w^{M+1} C_\sigma^M}{\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M} + \frac{\mathbf{X}_\ell}{\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma} \right) \left[\prod_{\ell=0}^{L} (\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma) \right] \left(\frac{1}{n} \right)^{\frac{1}{\hat{d}+1}}
\end{aligned}
$$

where $\widetilde{\gamma}$ is the positive constant defined by

$$
\begin{aligned}
\widetilde{\gamma} := 4 &\left\{ \left(\frac{\hat{d}-1}{2} \right)^{\frac{2}{\hat{d}+1}} + \left(\frac{\hat{d}-1}{2} \right)^{-\frac{\hat{d}-1}{\hat{d}+1}} \right\} \left(\frac{12}{\hat{d}-1} \right)^{\frac{2}{\hat{d}+1}} \\
&\times \max\left[2C_w, 16|D|C_\alpha \max\left\{ C_\beta \mathrm{diag}(D\times D), 2 \right\} \right] (C_d^4 M)^{\frac{\hat{d}+2}{\hat{d}+1}} \rho C_a \quad \square
\end{aligned} \tag{C.27}
$$

C.4. Proof of Corollary 8.6

Proof. By using Assumption 7.3, we estimate for $\mathcal{G} \in \mathscr{N}$ and $a \in \mathrm{supp}(\mu_a)$,

$$
\begin{aligned}
\|\mathcal{G}(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})} &= \left\| (W_L + \boldsymbol{\mathcal{K}}_L) \circ \sigma(W_{L-1} + \boldsymbol{\mathcal{K}}_{L-1}) \circ \cdots \circ \sigma(W_0 + \boldsymbol{\mathcal{K}}_0)(a) \right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})} \\
&\le (C_w + C_k)^{L+1} C_\sigma^L C_a.
\end{aligned}
$$

Then, by applying Lemma 8.3 as $R = (C_w + C_k)^{L+1} C_\sigma^L C_a$, and combining with Theorem 8.4, we conclude that the inequality (38). $\square$

C.5. Proof of Corollary 8.7

Proof. By using Assumption 7.4, we estimate for $\mathcal{G} \in \widetilde{\mathscr{N}}$ and $a \in \mathrm{supp}(\mu_a)$,

$$\begin{aligned}
&\|\mathcal{G}(a)\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})} \\
&= \left\|(\mathbf{Z}_\ell \mathbf{I_d} + \mathbf{X}_\ell f_\ell)\circ(\mathbf{Z}_\ell \mathbf{I_d} + \mathbf{X}_\ell \sigma \circ \boldsymbol{\mathcal{K}}_\ell)\circ \cdots \circ (\mathbf{Z}_0 \mathbf{I_d} + \mathbf{X}_0 f_0)\circ(\mathbf{Z}_0 \mathbf{I_d} + \mathbf{X}_0 \sigma \circ \boldsymbol{\mathcal{K}}_0)(a)\right\|_{\mathrm{L}^2(D;\mathbb{R}^{d_u})} \\
&\le \left[\prod_{\ell=0}^{L} (\mathbf{Z}_L + \mathbf{X}_L C_w^{M+1} C_\sigma^M)(\mathbf{Z}_L + \mathbf{X}_L C_k C_\sigma)\right] C_a.
\end{aligned}$$

Then, by applying Lemma 8.3 as $R = \left[\prod_{\ell=0}^{L} (\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma)\right] C_a$ and combining with Theorem 8.5, we conclude that the inequality (39). □

Appendix D. Remark for Sections 8.3, 8.4, 8.5

Remark D.1. *In the implementation of NO,* $\boldsymbol{\mathcal{K}}_\ell$ is projected into a *finite-rank operator* by the chosen basis. For clarity's sake, let assume $\boldsymbol{\mathcal{K}}_\ell : \mathrm{L}^2(D) \to \mathrm{L}^2(D)$, i.e., domain and range are the same space, and $\mathrm{L}^2(D) = \mathrm{L}^2(D;\mathbb{R})$. Let $k_\ell \in \mathrm{L}^2(D \times D)$ be the kernel of $\boldsymbol{\mathcal{K}}_\ell$, and let $\{\phi_j\}_{j\in\mathbb{N}}$ be an orthonormal basis in $\mathrm{L}^2(D)$,[35] so $\{\phi_i \otimes \phi_j\}_{i,j\in\mathbb{N}}$ is an orthonormal basis of $\mathrm{L}^2(D \times D)$, and thus $k_\ell(x,y) = \sum_{j,k\ge 1} k_{\ell,jk}\, \phi_j(x) \otimes \phi_k(y)$, where $k_{\ell,jk} \in \mathbb{R}$, $k_{\ell,jk} = \langle k_\ell, \phi_j(x) \otimes \phi_k\rangle_{\mathrm{L}^2(D\times D)} = \langle \phi_j, \mathcal{K}\phi_k\rangle_{\mathrm{L}^2(D)}$. By choosing N-modes (first N basis), the kernel k_ℓ is approximated as $k_\ell^{(N)}(x,y) = \sum_{j,k\le N} k_{\ell,jk}\, \phi_j(x) \otimes \phi_k(y)$, and so

$$\|k_\ell^{(N)}\|^2_{\mathrm{L}^2(D\times D)} = \sum_{j,k=1}^{N} |k_{\ell,jk}|^2 \le \sum_{j,k=1}^{\infty} |k_{\ell,jk}|^2 = \|k\|^2_{\mathrm{L}^2(D\times D)}.$$

Hence, the implementable[36] kernel $k_\ell^{(N)}$ satisfies (ii) Assumption 7.3, and the Rademacher Complexity for (36) is also an upper-bound.

Remark D.2 *(Summary of generalization error bounds).*

(Bound for NO)

$$\lesssim L^{\frac{\hat{d}+2}{\hat{d}+1}} \{(C_w + C_k)C_\sigma\}^L \left(\frac{1}{n}\right)^{\frac{1}{\hat{d}+1}} + \{(C_w + C_k)C_\sigma\}^L \sqrt{\frac{2\delta}{n}}.$$

(Bound for sNO)

$$\lesssim L^{\frac{\hat{d}+2}{\hat{d}+1}} (C_w^{M+1} C_\sigma^{M+1} C_k)^L \left(\frac{1}{n}\right)^{\frac{1}{\hat{d}+1}} + (C_w^{M+1} C_\sigma^{M+1} C_k)^L \sqrt{\frac{2\delta}{n}}.$$

(Bound for $(s\mathrm{NO} + \varepsilon\mathrm{I})\upsilon 1$)

$$\lesssim L^{\frac{\hat{d}+2}{\hat{d}+1}} \left\{(1 + C_w^{M+1} C_\sigma^M)(1 + C_\sigma C_k)\right\}^L \left(\frac{1}{n}\right)^{\frac{1}{\hat{d}+1}} + \left\{(1 + C_w^{M+1} C_\sigma^M)(1 + C_\sigma C_k)\right\}^L \sqrt{\frac{2\delta}{n}}.$$

(Bound for $(s\mathrm{NO} + \varepsilon\mathrm{I})\upsilon 2$)

$$\begin{aligned}
&\lesssim L^{\frac{1}{\hat{d}+1}} \left(\sum_{\ell=0}^{L} \frac{\mathbf{X}_\ell C_w^{M+1} C_\sigma^M}{1 + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M} + \frac{\mathbf{X}_\ell}{1 + \mathbf{X}_\ell C_k C_\sigma}\right) \left[\prod_{\ell=0}^{L} (1 + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(1 + \mathbf{X}_\ell C_k C_\sigma)\right] \left(\frac{1}{n}\right)^{\frac{1}{\hat{d}+1}} \\
&+ \left[\prod_{\ell=0}^{L} (\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma)\right] \sqrt{\frac{2\delta}{n}}.
\end{aligned}$$

Here, $\lesssim$ implies that the left-hand side is bounded above by the right-hand side times a constant independent of n and L. Hence, Remark 8.2 can be observed.

[35] For FNO, the basis are the Fourier basis.

[36] In a computer.

Lemma D.1. *Let $\mathbf{Z}_\ell = 1$ and $\mathbf{X}_\ell$ be a Bernoulli RV with $\mathbf{P}\{\mathbf{X}_\ell = 1\} = p_\ell$, and $\mathbf{P}\{\mathbf{X}_\ell = 0\} = 1 - p_\ell$ for $p_\ell \in [0,1]$ in inequality (39). We assume that $p_\ell = x_\ell / L^{\frac{1}{d+1}}$ where $x_\ell \in [0,1]$ satisfies $\sum_{\ell=0}^{\infty} x_\ell < \infty$. Then,*

$$\mathbb{E}_{\boldsymbol{\mathcal{X}}}[\textit{RHS of (39)}] \lesssim \widehat{\mathcal{L}}_S(\mathcal{G}) + \left(\sum_{\ell=1}^{L} x_\ell\right) \prod_{\ell=0}^{L} \left[1 + (C_w^{M+1} C_\sigma^M + C_k C_\sigma + C_w^{M+1} C_k C_\sigma^{M+1}) x_\ell\right] \left(\frac{1}{n}\right)^{\frac{1}{d+1}} + \left(\rho \prod_{\ell=0}^{L} \left[1 + (C_w^{M+1} C_\sigma^M + C_k C_\sigma + C_w^{M+1} C_k C_\sigma^{M+1}) x_\ell\right] C_a + R_u\right) \sqrt{\frac{2\delta}{n}}.$$

Here, $\lesssim$ implies that the left-hand side is bounded above by the right-hand side times a constant independent of n and L.

We remark that the upper bound remain bounded as L tends to infinity because $\sum_{\ell=1}^{\infty} x_\ell < \infty$ and

$$\sum_{\ell=1}^{\infty} \left(1 + (C_w^{M+1} C_\sigma^M + C_k C_\sigma + C_w^{M+1} C_k C_\sigma^{M+1}) x_\ell\right) < \infty.$$

As result, infinite products also remain bounded.

Proof. First, we evaluate that

$$[\text{RHS of (39)}] \leq \widehat{\mathcal{L}}_S(\mathcal{G}) + 4\widetilde{\gamma}(C_w^{M+1} C_\sigma^M + 1) L^{\frac{1}{d+1}} \left(\sum_{\ell=1}^{L} \mathbf{X}_\ell\right) \left[\prod_{\ell=0}^{L} (1 + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(1 + \mathbf{X}_\ell C_k C_\sigma)\right] \left(\frac{1}{n}\right)^{\frac{1}{d+1}} + \left(\rho \left[\prod_{\ell=0}^{L} (1 + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(1 + \mathbf{X}_\ell C_k C_\sigma)\right] C_a + R_u\right) \sqrt{\frac{2\delta}{n}},$$

which implies that

$$\mathbb{E}_{\boldsymbol{\mathcal{X}}}[\text{RHS of (39)}] \lesssim \widehat{\mathcal{L}}_S(\mathcal{G}) + L^{\frac{1}{d+1}} \sum_{\ell=1}^{L} \mathbb{E}_{\mathbf{X}_\ell} \left[\mathbf{X}_\ell (1 + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(1 + \mathbf{X}_\ell C_k C_\sigma)\right] \times \mathbb{E}_{\boldsymbol{\mathcal{X}} \setminus \mathbf{X}_\ell} \left[\prod_{\ell'=0}^{L} (1 + \mathbf{X}_{\ell'} C_w^{M+1} C_\sigma^M)(1 + \mathbf{X}_{\ell'} C_k C_\sigma)\right] \left(\frac{1}{n}\right)^{\frac{1}{d+1}} + \left(\rho \mathbb{E}_{\boldsymbol{\mathcal{X}}} \left[\prod_{\ell=0}^{L} (\mathbf{Z}_\ell + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(\mathbf{Z}_\ell + \mathbf{X}_\ell C_k C_\sigma)\right] C_a + R_u\right) \sqrt{\frac{2\delta}{n}}.$$

Since we have

$$\mathbb{E}_{\mathbf{X}_\ell} \left[\mathbf{X}_\ell (1 + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(1 + \mathbf{X}_\ell C_k C_\sigma)\right] = (1 + C_w^{M+1} C_\sigma^M + C_k C_\sigma + C_w^{M+1} C_k C_\sigma^{M+1}) p_\ell,$$

and

$$\mathbb{E}_{\mathbf{X}_\ell} \left[(1 + \mathbf{X}_\ell C_w^{M+1} C_\sigma^M)(1 + \mathbf{X}_\ell C_k C_\sigma)\right] = 1 + (C_w^{M+1} C_\sigma^M + C_k C_\sigma + C_w^{M+1} C_k C_\sigma^{M+1}) p_\ell,$$

we conclude that by using $p_\ell = x_\ell / L^{\frac{1}{d+1}}$,

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{\mathcal{X}}}[\text{RHS of (39)}] \\
&\lesssim \widehat{\mathcal{L}}_S(\mathcal{G}) + L^{\frac{1}{d+1}} \left(\sum_{\ell=1}^{L} p_\ell\right) \prod_{\ell=0}^{L} \left[1 + (C_w^{M+1} C_\sigma^M + C_k C_\sigma + C_w^{M+1} C_k C_\sigma^{M+1}) p_\ell\right] \left(\frac{1}{n}\right)^{\frac{1}{d+1}} \\
&+ \left(\rho \prod_{\ell=0}^{L} \left[1 + (C_w^{M+1} C_\sigma^M + C_k C_\sigma + C_w^{M+1} C_k C_\sigma^{M+1}) p_\ell\right] C_a + R_u\right) \sqrt{\frac{2\delta}{n}} \\
&\lesssim \widehat{\mathcal{L}}_S(\mathcal{G}) + \left(\sum_{\ell=1}^{L} x_\ell\right) \prod_{\ell=0}^{L} \left[1 + \frac{(C_w^{M+1} C_\sigma^M + C_k C_\sigma + C_w^{M+1} C_k C_\sigma^{M+1})}{L^{\frac{1}{d+1}}} x_\ell\right] \left(\frac{1}{n}\right)^{\frac{1}{d+1}} \\
&+ \left(\rho \prod_{\ell=0}^{L} \left[1 + \frac{(C_w^{M+1} C_\sigma^M + C_k C_\sigma + C_w^{M+1} C_k C_\sigma^{M+1})}{L^{\frac{1}{d+1}}} x_\ell\right] C_a + R_u\right) \sqrt{\frac{2\delta}{n}} \quad \square
\end{aligned}$$

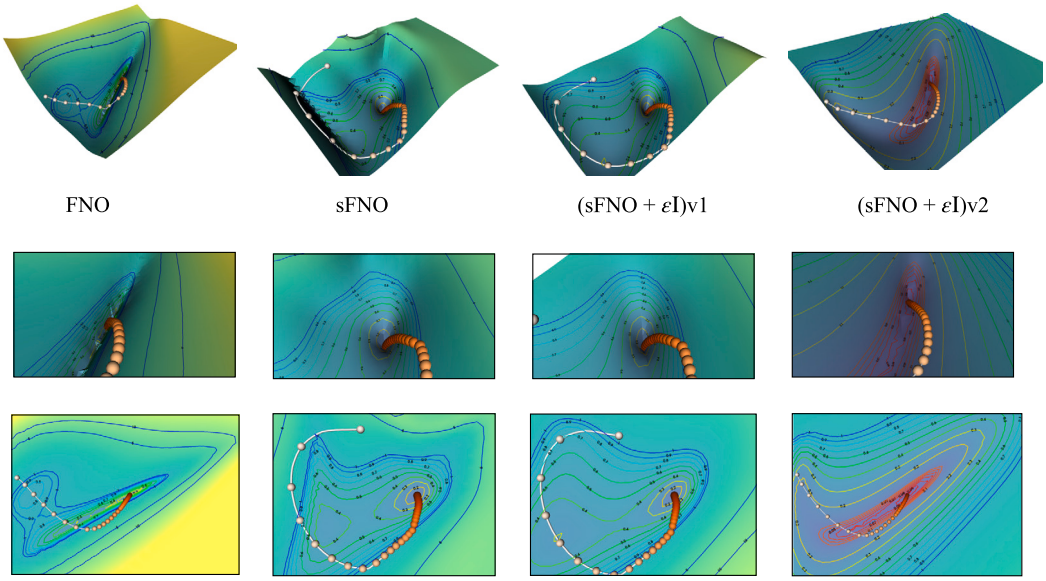


Fig. E.20. Visualization of the training landscapes associated with FNO, *s*FNO, *s*FNO + εI v1 and *s*FNO + εI v2.

Appendix E. Experiments

E.1. Loss landscape visualization

We include here additional views of the training loss landscape of the considered architectures that were created using the method discussed in Section 4.5. In particular, the images below offer a closer view of the landscape in the immediate vicinity of the found minimizer, to allow for a better comparison. In addition, a color-based planar view of the landscapes is provided for a better view of their respective topological features. (See Fig. E.20.)

E.2. Out-of-distribution

In this section, we present the wavefield reconstruction of the other families described in Section 5. The values of the parameters are established in Table 5, and the relative test loss error is presented in Tables 6 to 11. In our analysis, we selected three realizations from the previously trained neural networks. These networks were trained using a dataset at a frequency of 15 Hz and with the parameters of the random field generating the wave speed set as $\lambda = (1, 1)$ and a wave speed range of $[1500, 5000]$. Specifically, we chose the first three networks documented in Fig. 7.

To test the performance of these networks on a different random field, we kept the smoothness coefficient constant and varied the correlation range of the Whittle-Matérn field. The reconstructed wave fields are presented in Figs. E.21 to E.26. Please note that the imaginary part of the wave field is also recovered, but it is not shown in the figures.

OOD 1 In this set family, we keep the isotropic behavior of the original data, however we move the value to $\lambda_{\text{OOD1}} = (0.20, 0.20)$. The range is kept in $[1500, 5000]$. We see that this scenario is the *easier* for the networks. However, FNO still struggles to capture the correct wave propagation.

OOD 2 In this set family, we generate an anisotropic random field, different to the original trained data $\lambda_{\text{OOD2}} = (0.10, 0.20)$, however the range was kept similar than the original set.

OOD 3 In this set family, we generate an isotropic random field, different to the original trained data $\lambda_{\text{OOD3}} = (0.20, 0.20)$, however the range was moved to $[2000, 3500]$.

OOD 4 In this set family, we generate an anisotropic random field, different to the original trained data $\lambda_{\text{OOD4}} = (0.10, 0.20)$, however the range was kept to $[2000, 3500]$ the same as the original set.

OOD 5 In this set family, we generate an isotropic random field, different to the original trained data $\lambda_{\text{OOD5}} = (0.10, 0.30)$, however the range was also moved to $[2000, 6000]$ different than the original set.

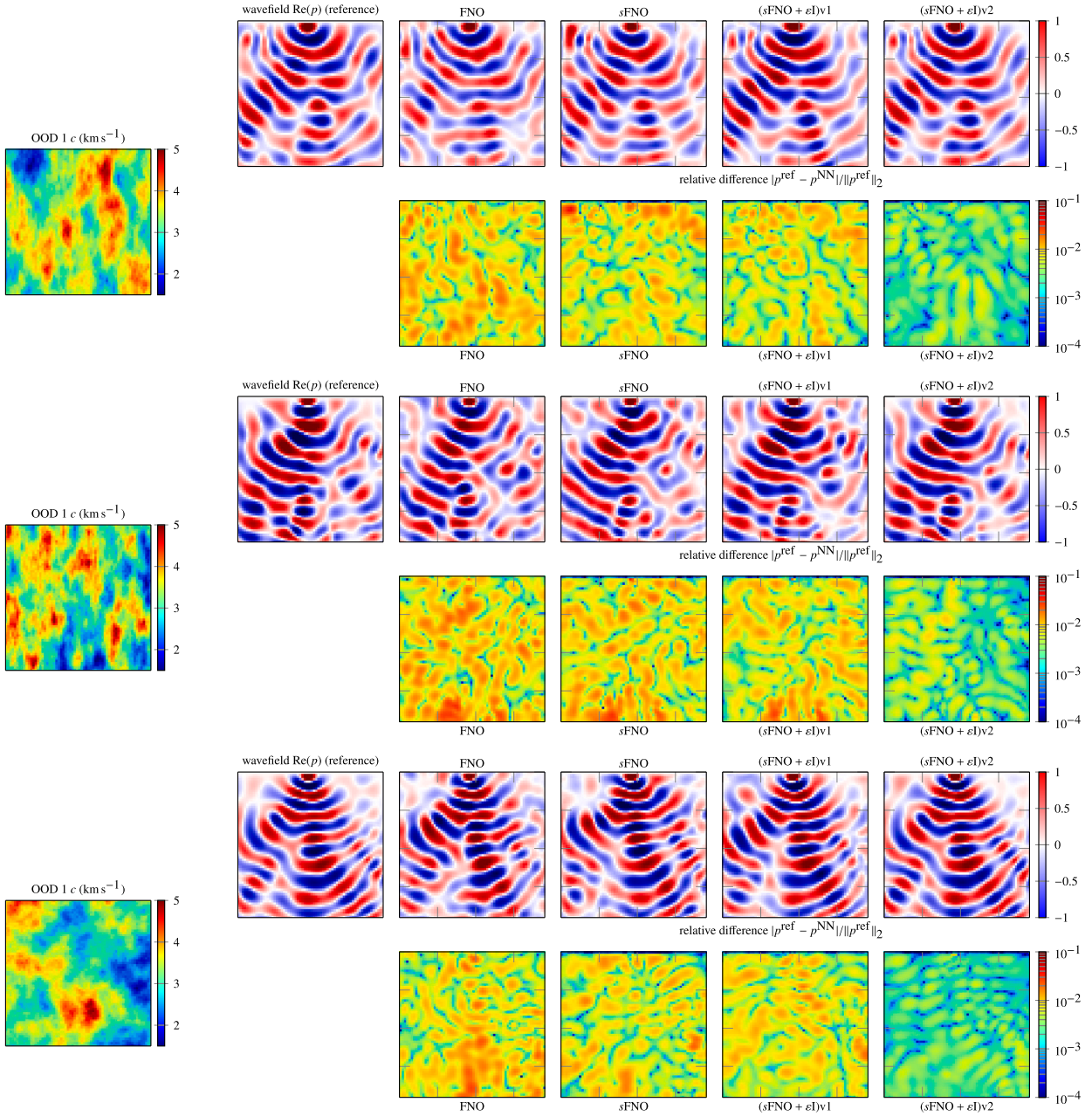


Fig. E.21. Pressure field reconstructed at 15 Hz trained with *isotropic Whittle–Matérn covariance* $\lambda = (1, 1)$, and wavespeed range of (1500, 5000) Equation (15) and tested with Table 6 $\lambda_{\text{OOD1}} = (0.20, 0.20)$, and wavespeed range of (1500, 5000) with the different architectures for multiple realizations of the new GRF *out-of distribution*, realizations of the wave speed. *Left column* shows independent GRF realization of the wave speed (see Equation (10)). *Second column* shows the real part of the pressure field solution to the wave PDE at frequency 15 Hz, obtained with software `hawen` [35], which we consider as the *reference solution*, *Other columns* show the approximated reconstructions using the different architectures. In each case, we show the real parts of the pressure fields, and the relative error with the reference solution on a logarithmic scale.

OOD 6 In this set family, we generate an anisotropic random field, *significantly* different to the original trained data $\lambda_{\text{OOD6}} = (0.25, 0.75)$, however the range was moved to [2000, 6000] different than the original set.

E.3. OOD of the velocity BP 2004

To assess the network's ability to handle wave speed that are significantly different from the input distribution (particularly those that deviate from Gaussian measures), we conducted additional tests using the trained networks on a scale version of the velocity model known as the "2004-BP velocity benchmark" [12]. The source was positioned similarly to the previous experiments,

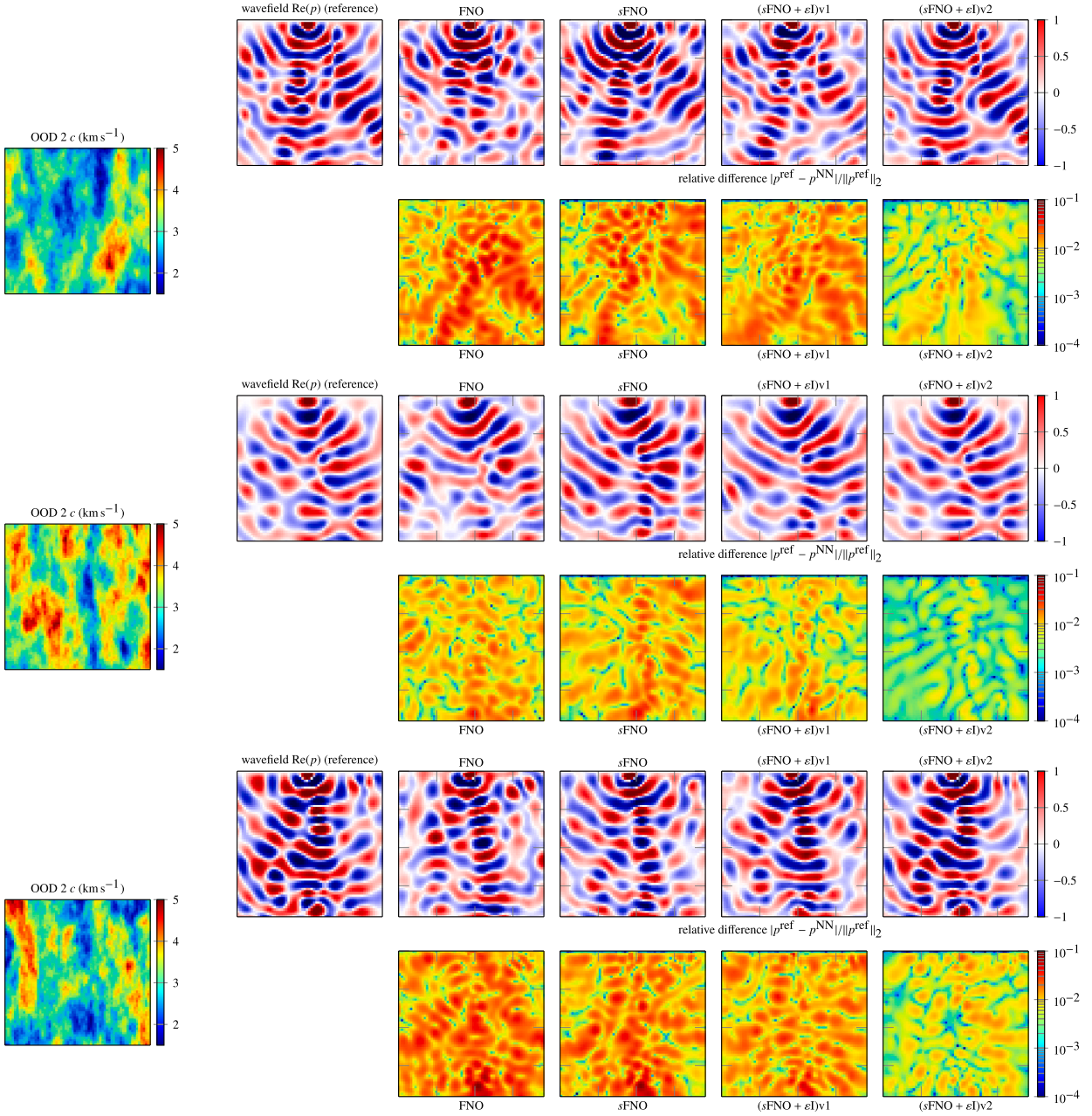


Fig. E.22. Pressure field reconstructed at 15 Hz trained with *isotropic Whittle–Matérn covariance* $\lambda = (1, 1)$, and wavespeed range of $(1500, 5000)$ Equation (15) and tested with Table 7 $\lambda_{\text{OOD2}} = (0.10, 0.20)$, and wavespeed range of $(1500, 5000)$ with the different architectures for multiple realizations of the new GRF *out-of distribution*, realizations of the wave speed. *Left column* shows independent GRF realization of the wave speed (see Equation (10)). *Second column* shows the real part of the pressure field solution to the wave PDE at frequency 15 Hz, obtained with software `hawen` [35], which we consider as the *reference solution*, *Other columns* show the approximated reconstructions using the different architectures. In each case, we show the real parts of the pressure fields, and the relative error with the reference solution on a logarithmic scale.

maintaining a frequency of 15 Hz, while adjusting the wavespeed's size to accommodate the capabilities of the GPU device. The generated approximations by each network are visualized in Fig. E.27.

E.4. Experiments at 7*,* 12 *and* 15 *Hz*

We consider two further datasets, lower frequency with a similar configuration as in Equation (12) at 12Hz, and an unrealistic case with the source beneath the surface, at 7 Hz, but we increase the size of the domain.

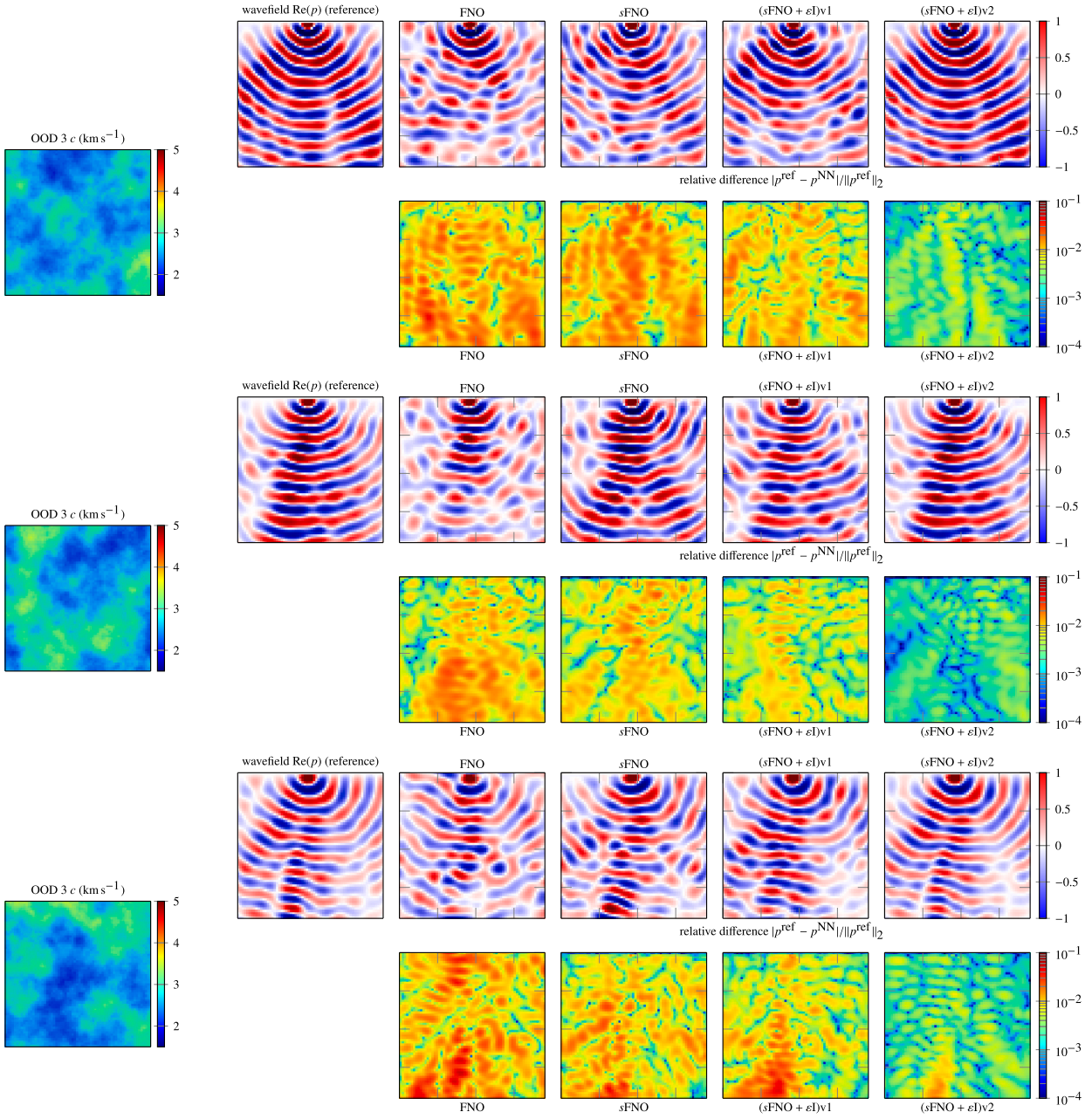


Fig. E.23. Pressure field at 15 Hz trained with *isotropic Whittle–Matérn covariance* $\lambda = (1, 1)$, and wavespeed range of (1500, 5000) Equation (15) and tested with Table 9 $\lambda_{\text{OOD3}} = (0.20, 0.20)$, and wavespeed range of (2000, 3500) with the different architectures for multiple realizations of the new GRF *out-of distribution*, realizations of the wave speed. *Left column* shows independent GRF realization of the wave speed (see Equation (10)). *Second column* shows the real part of the pressure field solution to the wave PDE at frequency 15 Hz, obtained with software `hawen` [35], which we consider as the *reference solution*, *Other columns* show the approximated reconstructions using the different architectures. In each case, we show the real parts of the pressure fields, and the relative error with the reference solution on a logarithmic scale.

Remark E.1. Similarly as in Section 4 *we deliberately avoid increasing the epochs of the training algorithm or the size of the training dataset to compensate the network.*

Experiments of 7 Hz (different configuration)

$$\text{Experiment 7 Hz} \begin{cases} \text{2D domain of size } 3.81 \times 3.81 \text{ km}^2 \\ 40\,000 \text{ GRF wave speeds generated, imposing } \ 1.5 \text{ km s}^{-1} \leq c(x) \leq 3 \text{ km s}^{-1} \\ \text{The data are } p \text{ that solve Equation (8) at frequency } \omega/(2\pi) = 7 \text{ Hz.} \end{cases} \tag{E.1}$$

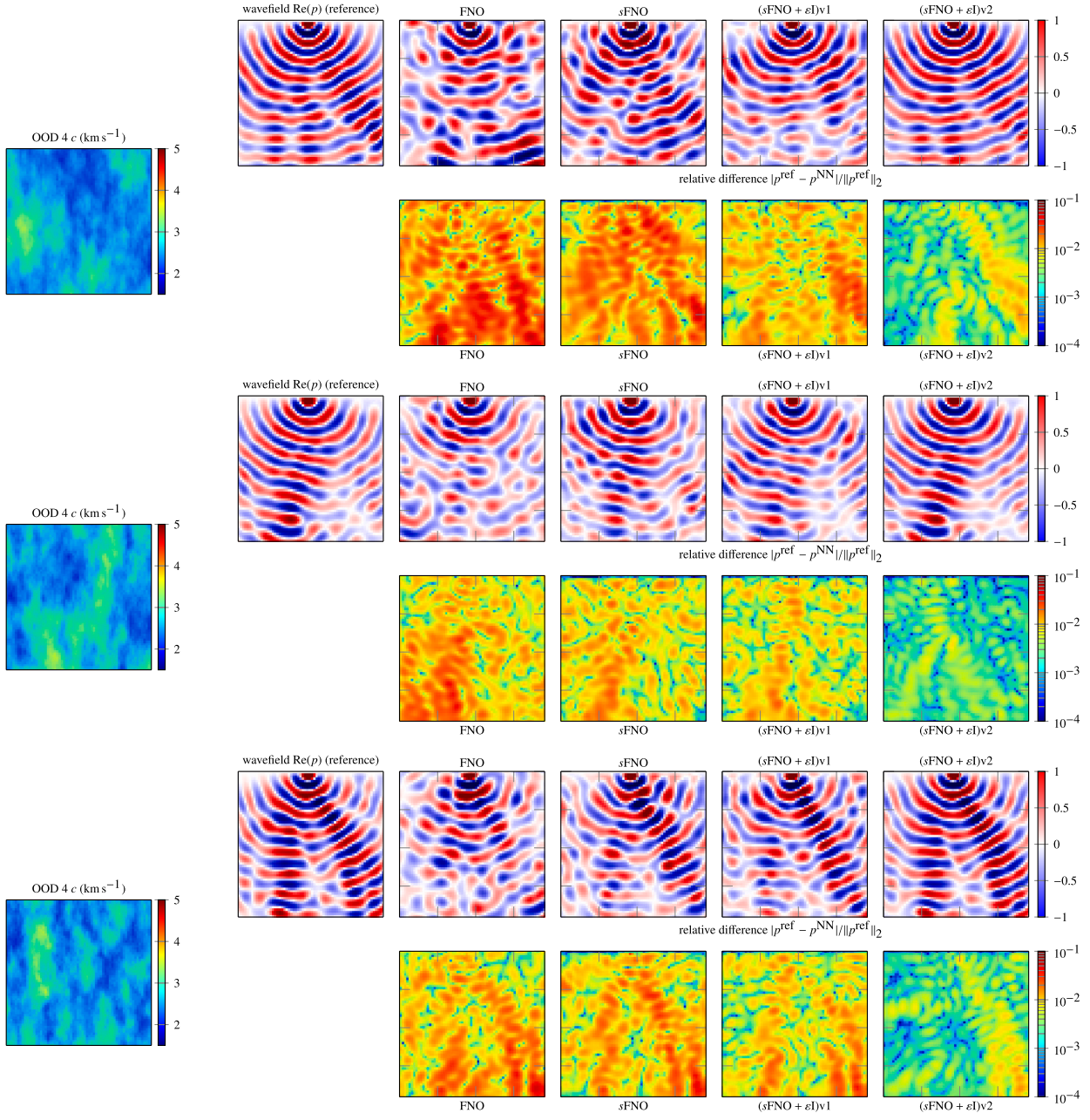


Fig. E.24. Pressure field reconstructed trained with *isotropic Whittle–Matérn covariance* $\lambda = (1, 1)$, and wavespeed range of $(1500, 5000)$ Equation (15) and tested with Table 9 $\lambda_{\text{OOD4}} = (0.10, 0.20)$, and wavespeed range of $(2000, 3500)$ with the different architectures for multiple realizations of the new GRF *out-of distribution*, realizations of the wave speed. *Left column* shows independent GRF realization of the wave speed (see Equation (10)). *Second column* shows the real part of the pressure field solution to the wave PDE at frequency 15 Hz, obtained with software `hawen` [35], which we consider as the *reference solution*, *Other columns* show the approximated reconstructions using the different architectures. In each case, we show the real parts of the pressure fields, and the relative error with the reference solution on a logarithmic scale.

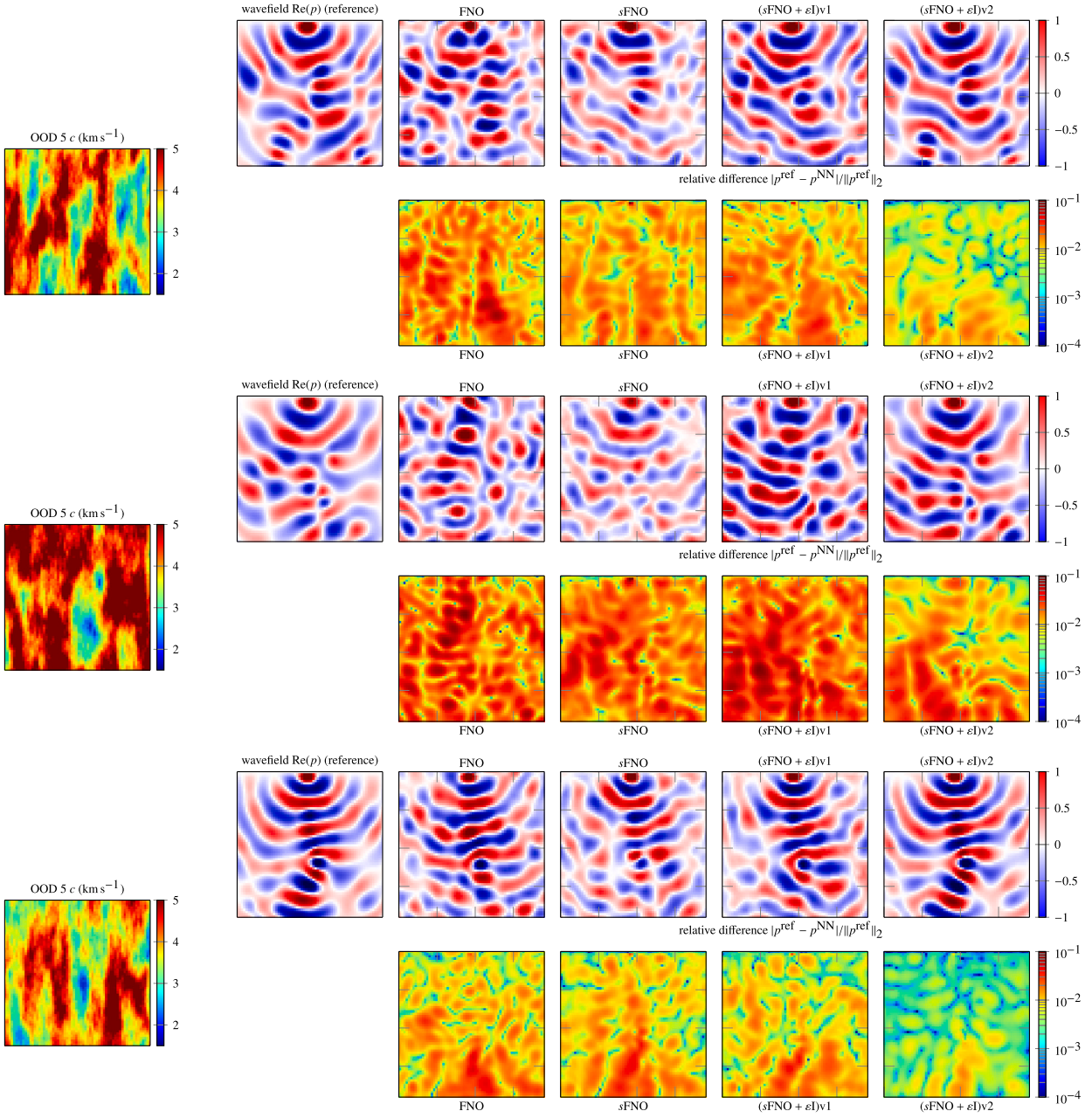


Fig. E.25. Pressure field reconstructed at 15 Hz trained with *isotropic Whittle–Matérn covariance* $\lambda = (1, 1)$, and wavespeed range of $(1500, 5000)$ Equation (15) and tested with Table 10 $\lambda_{\text{OOD5}} = (0.10, 0.30)$, and wavespeed range of $(2000, 6000)$ with the different architectures for multiple realizations of the new GRF *out-of distribution*, realizations of the wave speed. *Left column* shows independent GRF realization of the wave speed (see Equation (10)). *Second column* shows the real part of the pressure field solution to the wave PDE at frequency 15 Hz, obtained with software `hawen` [35], which we consider as the *reference solution*, *Other columns* show the approximated reconstructions using the different architectures. In each case, we show the real parts of the pressure fields, and the relative error with the reference solution on a logarithmic scale.

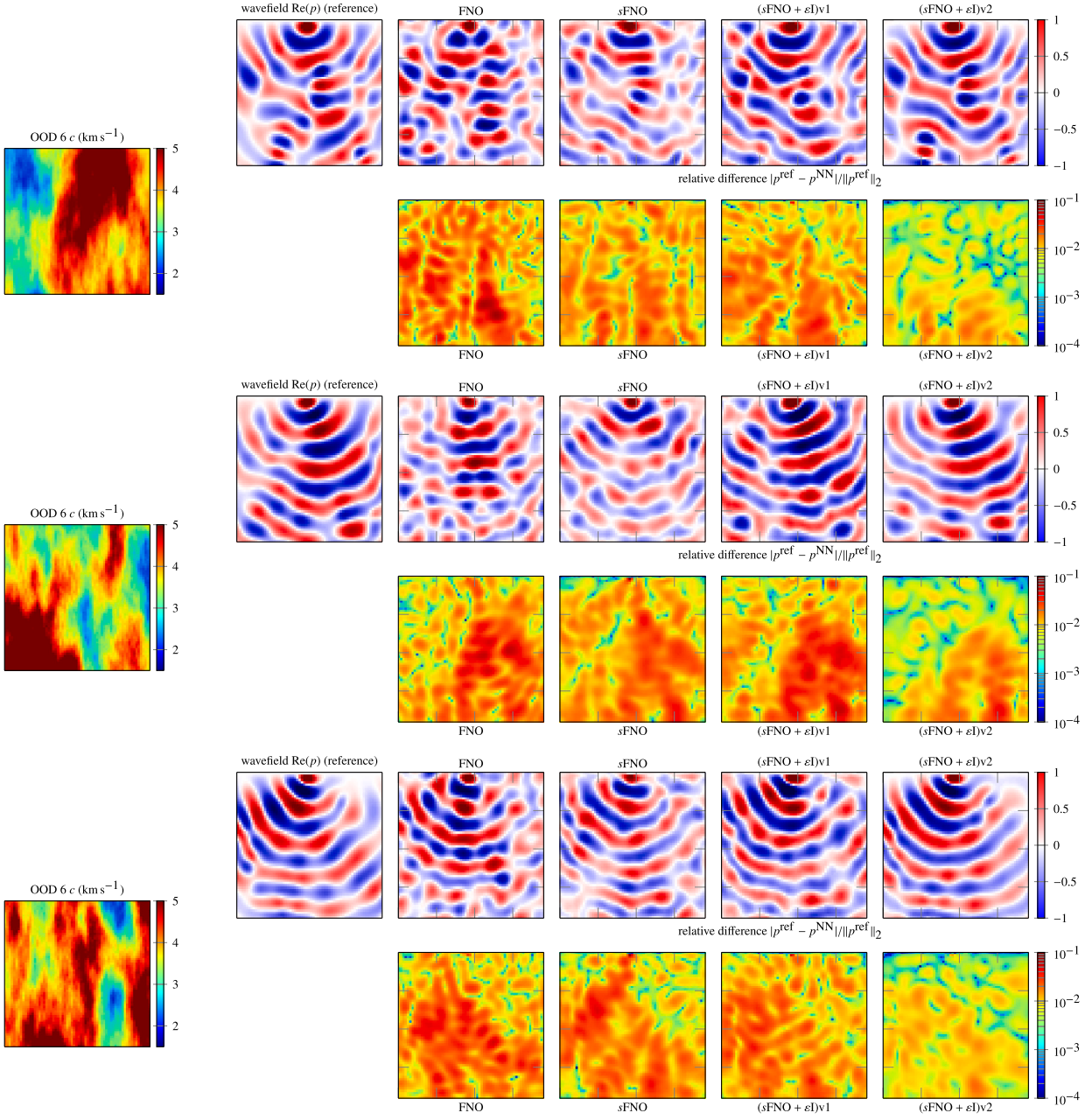


Fig. E.26. Pressure field reconstructed at 15 Hz trained with *isotropic Whittle–Matérn covariance* $\lambda = (1, 1)$, and wavespeed range of $(1500, 5000)$ Equation (15) and tested with Table 11 $\lambda_{\text{OOD6}} = (0.25, 0.75)$, and wavespeed range of $(2000, 6000)$ with the different architectures for multiple realizations of the new GRF *out-of distribution*, realizations of the wave speed. *Left column* shows independent GRF realization of the wave speed (see Equation (10)). *Second column* shows the real part of the pressure field solution to the wave PDE at frequency 15 Hz, obtained with software `hawen` [35], which we consider as the *reference solution*, *Other columns* show the approximated reconstructions using the different architectures. In each case, we show the real parts of the pressure fields, and the relative error with the reference solution on a logarithmic scale.

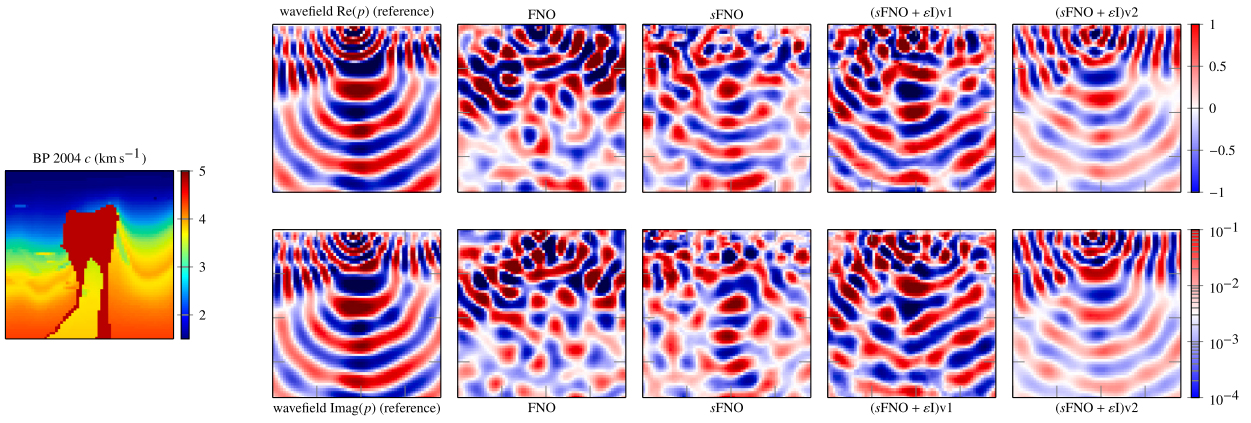
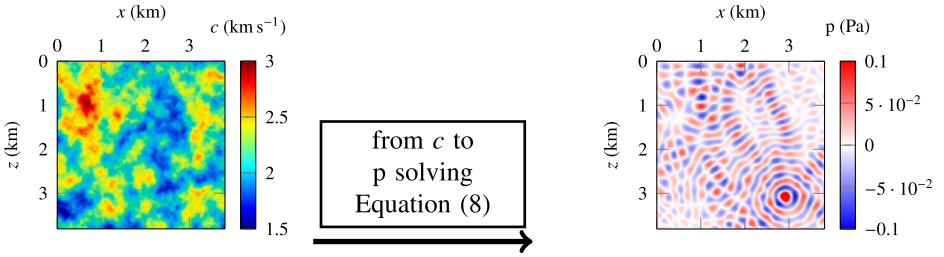


Fig. E.27. BP 2004 [12]. Using the networks trained in row 1 of Table 3.



(a) GRF realization of wave speed model c

(b) Real part of the pressure field at 7 Hz frequency.

Fig. E.28. Illustration of the full-wave dataset for Experiment 1 that considers a computational domain of size 3.81×3.81 km^2 with a source buried in the domain. The wave speed and pressure field are represented on a Cartesian grid of size 128×128 with a grid step of 30 m. The complete dataset corresponds to 40 000 couples made up of a wave speed model and associated acoustic wave.

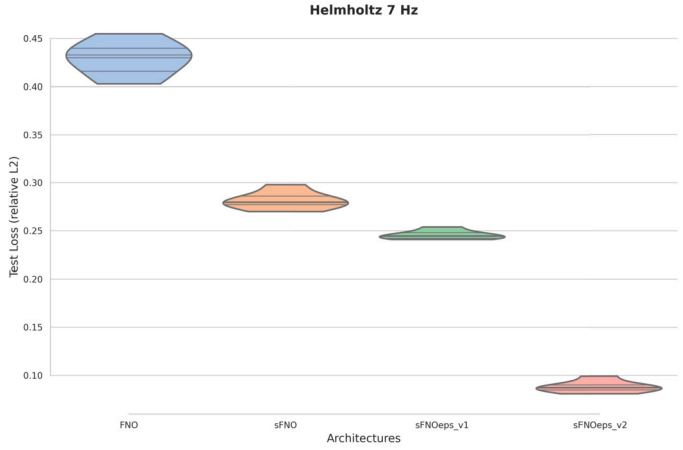


Fig. E.29. Comparison of test-loss for $\omega/(2\pi) = 7$ Hz. Each architecture is trained 9 times, the relative L^2-loss, $\|\mathcal{G}^{\text{ref}} - \mathcal{G}^{\text{approx}}\|_{\text{L}^2} / \|\mathcal{G}^{\text{ref}}\|_{\text{L}^2}$, on the test set is shown in the diagram.

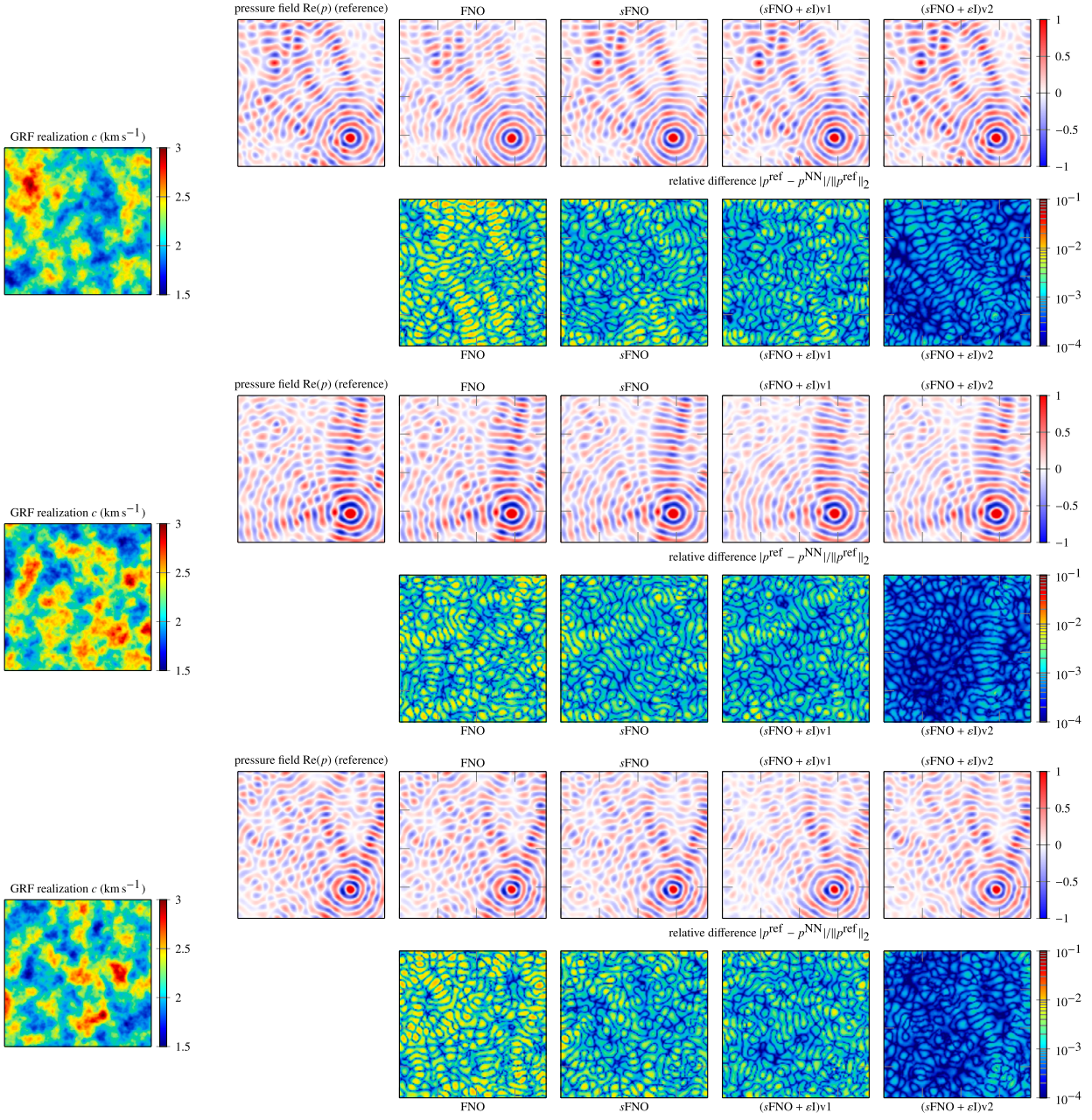


Fig. E.30. Pressure field reconstructed at frequency 7 Hz with the different architectures for three test-cases. *First column* shows independent GRF realization of the wave speed (see Equation (10)). *Second column* shows the solution of the wave PDE obtained with software `hawen` [35], which we consider as the *reference solution*, see Equation (10). *Other columns* show the approximated reconstruction using the different architectures: ***FNO***, see Kovachki et al. [66]; sequential structure (sFNO, see Section 2); and the solutions provided by $s\text{FNO} + \varepsilon\text{I}$, Section 2. In each case, we show the real part of the pressure field, and the relative error with the reference solution using a logarithmic scale.

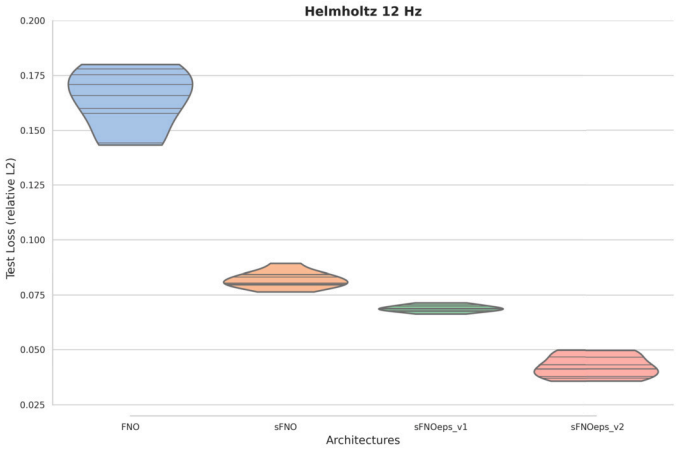


Fig. E.31. Comparison of test-loss for $\omega/(2\pi) = 12$ Hz. Each architecture is trained 9 times, the relative L^2-loss, $\|\mathcal{G}^{\text{ref}} - \mathcal{G}^{\text{approx}}\|_{L^2} / \|\mathcal{G}^{\text{ref}}\|_{L^2}$, on the test set is shown in the diagram.

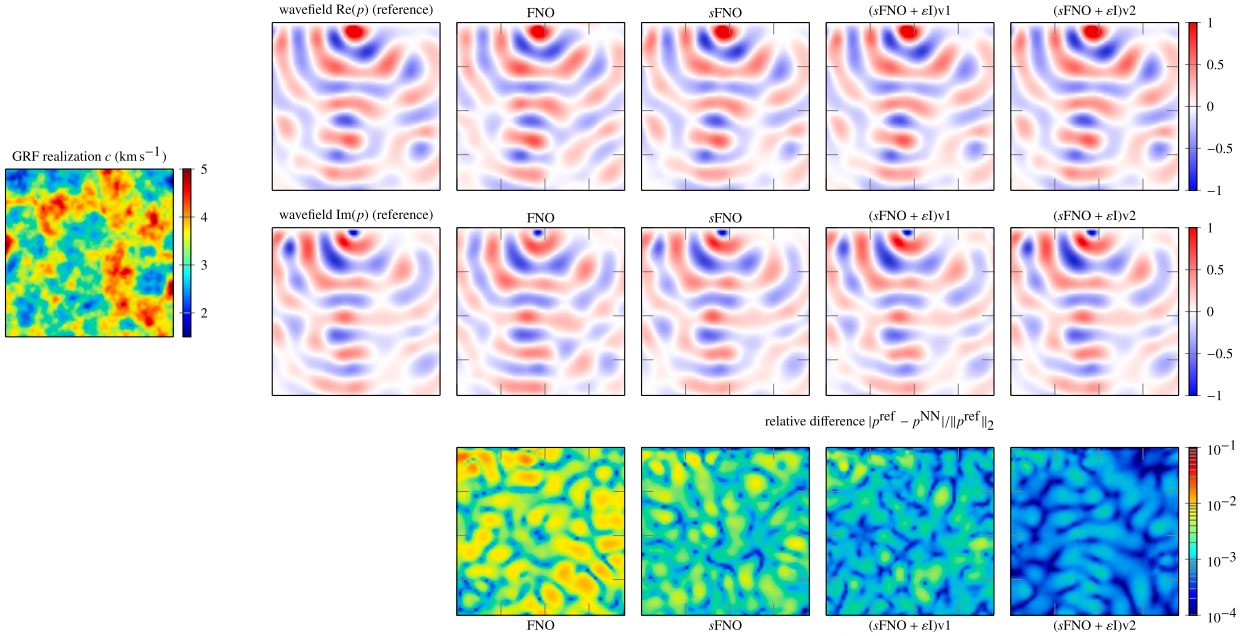


Fig. E.32. Pressure field reconstructed at frequency 12 Hz with the different architectures for two GRF realizations of the wave speed. *Left column* shows independent GRF realization of the wave speed (see Equation (10)). *Second column* shows the real and imaginary parts of the pressure field solution to the wave PDE at frequency 12 Hz, obtained with software `hawen` [35], which we consider as the *reference solution*, see Equation (10). *Other columns* show the approximated reconstructions using the different architectures: *FNO*, see Kovachki et al. [66]; sequential structure (*s*FNO, see Section 2); and the solutions provided by *s*FNO + εI, Section 2. In each case, we show the real and imaginary parts of the pressure fields, and the relative error with the reference solution on a logarithmic scale.

Both the wave speeds and the pressure field solution are represented on a Cartesian grid of size 128×128 pixels, that is, using a grid step of 30 m. We illustrate in Fig. E.28 a realization of the wave speed model and the corresponding pressure field. (See Figs. E.29 and E.30.)

E.5. Experiments at 12 Hz

$$\text{Experiment 2} \begin{cases} \text{2D domain of size } 1.27 \times 1.27 \text{ km}^2 \\ \text{40 000 GRF wave speeds generated, imposing } \ 1.5 \text{ km s}^{-1} \le c(x) \le 5 \text{ km s}^{-1} \\ \text{The data are } p \text{ that solve Equation (8) at frequency } \omega/(2\pi) = 12 \text{ Hz.} \end{cases} \tag{E.2}$$

See Figs. E.31 and E.32.

E.6. Wavefield reconstruction at 15 Hz

See Fig. E.33.

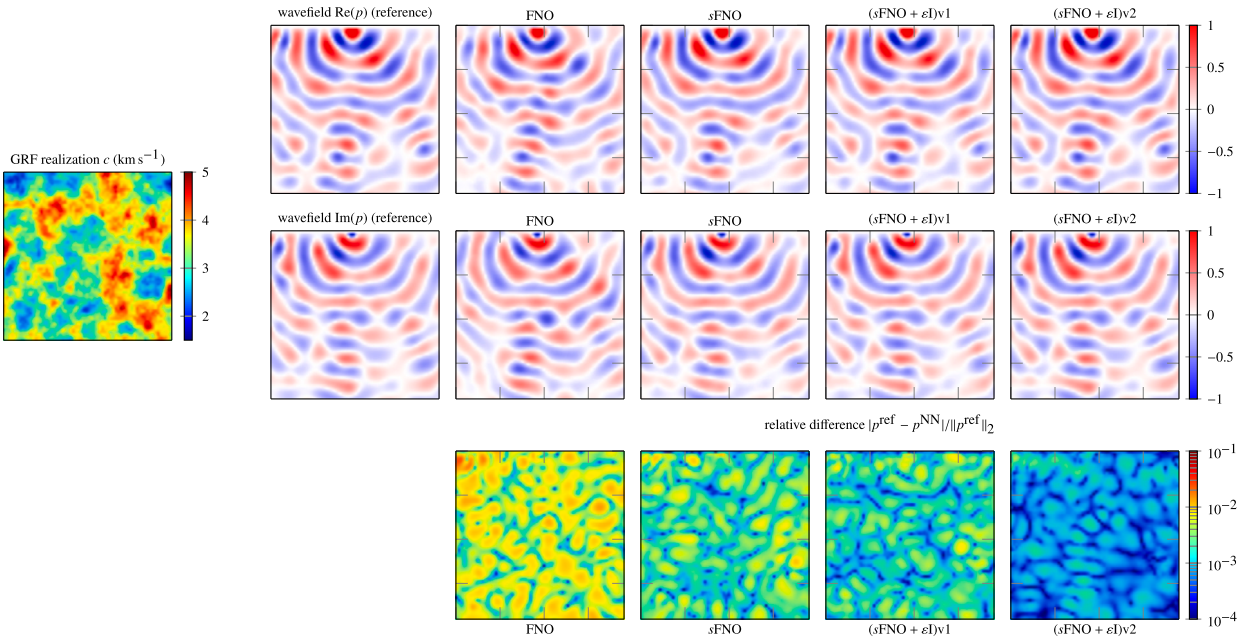


Fig. E.33. Pressure field reconstructed at frequency 15 Hz with the different architectures for two GRF realizations of the wave speed. *Left column* shows independent GRF realization of the wave speed (see Equation (10)). *Second column* shows the real and imaginary parts of the pressure field solution to the wave PDE at frequency 12 Hz, obtained with software `hawen` [35], which we consider as the *reference solution*, see Equation (10). *Other columns* show the approximated reconstructions using the different architectures: *FNO*, see Kovachki et al. [66]; sequential structure (sFNO, see Section 2); and the solutions provided by sFNO + εI, Section 2. In each case, we show the real and imaginary parts of the pressure fields, and the relative error with the reference solution on a logarithmic scale.

References

[1] Kweku Abraham, Richard Nickl, On statistical Calderón problems, Math. Stat. Learn. 2 (2) (2020) 165–216, https://doi.org/10.4171/MSL/14.

[2] Beatrice Acciaio, Anastasis Kratsios, Gudmund Pammer, Designing universal causal deep learning models: the geometric (hyper)transformer, in: Special Issue: Machine Learning in Finance, Math. Finance (2023) 1–65, https://doi.org/10.1111/mafi.12389, in press.

[3] Robert A. Adams, John JF Fournier, Sobolev Spaces, Elsevier, 2003.

[4] Luigi Ambrosio, Nicola Gigli, Giuseppe Savaré, Gradient Flows: in Metric Spaces and in the Space of Probability Measures, Springer Science & Business Media, 2005.

[5] George B. Arfken, Hans J. Weber, Mathematical Methods for Physicists, 1999.

[6] Emil Artin, The Gamma Function, Courier Dover Publications, 2015.

[7] Anatolii Borisovich Bakushinsky, M. Yu Kokurin, Iterative Methods for Approximate Solution of Inverse Problems, Springer Science & Business Media, vol. 577, 2005.

[8] Peter L. Bartlett, Dylan J. Foster, Matus J. Telgarsky, Spectrally-normalized margin bounds for neural networks, Adv. Neural Inf. Process. Syst. 30 (2017).

[9] Peter L. Bartlett, Andrea Montanari, Alexander Rakhlin, Deep learning: a statistical viewpoint, Acta Numer. 30 (2021) 87–201, https://doi.org/10.1017/S0962492921000027.

[10] Elena Beretta, Maarten V. De Hoop, Florian Faucher, Otmar Scherzer, Inverse boundary value problem for the Helmholtz equation: quantitative conditional Lipschitz stability estimates, SIAM J. Math. Anal. 48 (6) (2016) 3962–3983, https://doi.org/10.1137/15M1043856.

[11] Kaushik Bhattacharya, Bamdad Hosseini, Nikola B. Kovachki, Andrew M. Stuart, Model reduction and neural networks for parametric pdes, SMAI J. Comput. Math. 7 (2021) 121–157, https://doi.org/10.5802/smai-jcm.74.

[12] F.J. Billette, Sverre Brandsberg-Dahl, The 2004 bp velocity benchmark, in: 67th EAGE Conference & Exhibition, EAGE Publications BV, 2005, pp. cp–1.

[13] Vladimir Bogachev, Gaussian Measures, Mathematical Surveys and Monographs, American Mathematical Society, 2015.

[14] Emmanuel Boissard, Thibaut Le Gouic, On the mean speed of convergence of empirical and occupation measures in Wasserstein distance, Ann. Inst. Henri Poincaré Probab. Stat. 50 (2) (2014) 539–563, https://doi.org/10.1214/12-AIHP517.

[15] David Bolin, Kristin Kirchner, Mihály Kovács, Numerical solution of fractional elliptic stochastic pdes with spatial white noise, IMA J. Numer. Anal. 40 (2) (2020) 1051–1073, https://doi.org/10.1093/imanum/dry091.

[16] Frank Bowman, Introduction to Bessel Functions, Courier Corporation, 2012.

[17] Johannes Brandstetter, Rianne van den Berg, Max Welling, Jayesh K. Gupta, Clifford neural layers for pde modeling, arXiv preprint, arXiv:2209.04934, 2022.

[18] Qianying Cao, Somdatta Goswami, George Em Karniadakis, Lno: Laplace neural operator for solving differential equations, arXiv preprint, arXiv:2303.10528, 2023.

[19] Shuhao Cao, Choose a transformer: Fourier or Galerkin, Adv. Neural Inf. Process. Syst. 34 (2021) 24924–24940.

[20] Bernd Carl, Entropy numbers, s-numbers, and eigenvalue problems, J. Funct. Anal. 41 (3) (1981) 290–306, https://doi.org/10.1016/0022-1236(81)90076-8, ISSN 0022-1236.

[21] Neil K. Chada, Marco A. Iglesias, Lassi Roininen, Andrew M. Stuart, Parameterizations for ensemble Kalman inversion, Inverse Probl. 34 (5) (2018) 055009, https://doi.org/10.1088/1361-6420/aab6d9.
[22] Sonja G. Cox, Kristin Kirchner, Regularity and convergence analysis in Sobolev and Hölder spaces for generalized Whittle–matérn fields, Numer. Math. 146 (4) (2020) 819–873, https://doi.org/10.1007/s00211-020-01151-x.
[23] Giuseppe Da Prato, Jerzy Zabczyk, Stochastic Equations in Infinite Dimensions, Cambridge University Press, 2014.
[24] Masoumeh Dashti, Andrew M. Stuart, The Bayesian approach to inverse problems, in: Handbook of Uncertainty Quantification, Springer, 2017, pp. 311–428.
[25] Maarten V. de Hoop, Matti Lassas, Christopher A. Wong, Deep learning architectures for nonlinear operator functions and nonlinear inverse problems, Math. Stat. Learn. 4 (1) (2022) 1–86, https://doi.org/10.4171/MSL/28.
[26] Maarten V. de Hoop, Nikola B. Kovachki, Nicholas H. Nelsen, Andrew M. Stuart, Convergence rates for learning linear operators from noisy data, SIAM/ASA J. Uncertain. Quantificat. 11 (2) (2023) 480–513, https://doi.org/10.1137/21M1442942.
[27] Tim De Ryck, Siddhartha Mishra, Generic bounds on the approximation error for physics-informed (and) operator learning, Adv. Neural Inf. Process. Syst. 35 (2022) 10945–10958.
[28] Beichuan Deng, Yeonjong Shin, Lu Lu, Zhongqiang Zhang, George Em Karniadakis, Convergence rate of deeponets for learning operators arising from advection-diffusion equations, arXiv preprint, arXiv:2102.10621, 2021.
[29] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Bert: pre-training of deep bidirectional transformers for language understanding, arXiv preprint, arXiv:1810.04805, 2018.
[30] Claude R. Dietrich, Garry N. Newsam, Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix, SIAM J. Sci. Comput. 18 (4) (1997) 1088–1107, https://doi.org/10.1137/S1064827592240555.
[31] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al., An image is worth 16x16 words: transformers for image recognition at scale, arXiv preprint, arXiv:2010.11929, 2020.
[32] Björn Engquist, Andrew Majda, Absorbing boundary conditions for numerical simulation of waves, Proc. Natl. Acad. Sci. USA 74 (5) (1977) 1765–1766, https://doi.org/10.1073/pnas.74.5.1765.
[33] Yogi A. Erlangga, Advances in iterative methods and preconditioners for the Helmholtz equation, Arch. Comput. Methods Eng. 15 (2008) 37–66.
[34] Oliver G. Ernst, Martin J. Gander, Why it is difficult to solve Helmholtz problems with classical iterative methods, in: Numerical Analysis of Multiscale Problems, 2011, pp. 325–363.
[35] Florian Faucher, hawen: time-harmonic wave modeling and inversion using hybridizable discontinuous Galerkin discretization, J. Open Sour. Softw. 6 (57) (2021) 2699.
[36] Florian Faucher, Otmar Scherzer, Adjoint-state method for hybridizable discontinuous Galerkin discretization, application to the inverse acoustic wave problem, Comput. Methods Appl. Mech. Eng. 372 (2020) 113406, https://doi.org/10.1016/j.cma.2020.113406, ISSN 0045-7825.
[37] Florian Faucher, Giovanni Alessandrini, Hélène Barucq, Maarten V. de Hoop, Romina Gaburro, Eva Sincich, Full reciprocity-gap waveform inversion enabling sparse-source acquisition, Geophysics 85 (6) (2020) R461–R476, https://doi.org/10.1190/geo2019-0527.1.
[38] Luca Galimberti, Giulia Livieri, Anastasis Kratsios, Designing universal causal deep learning models: the case of infinite-dimensional dynamical systems from stochastic analysis, arXiv preprint, arXiv:2210.13300, 2022.
[39] Martin J. Gander, Hui Zhang, A class of iterative solvers for the Helmholtz equation: factorizations, sweeping preconditioners, source transfer, single layer potentials, polarized traces, and optimized Schwarz methods, SIAM Rev. 61 (1) (2019) 3–76, https://doi.org/10.1137/16M109781X.
[40] Subhashis Ghosal, Aad Van der Vaart, Fundamentals of Nonparametric Bayesian Inference, vol. 44, Cambridge University Press, 2017.
[41] Evarist Giné, Richard Nickl, Mathematical Foundations of Infinite-Dimensional Statistical Models, vol. 40, Cambridge University Press, 2015.
[42] Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 2016.
[43] Pulkit Gopalani, Sayar Karmakar, Anirbit Mukherjee, Capacity bounds for the deeponet method of solving differential equations, arXiv preprint, arXiv:2205.11359, 2022.
[44] Lee-Ad Gottlieb, Aryeh Kontorovich, Robert Krauthgamer, Adaptive metric dimensionality reduction, Theor. Comput. Sci. 620 (2016) 105–118, https://doi.org/10.1016/j.tcs.2015.10.040.
[45] Thomas J. Grady II, Rishi Khan, Mathias Louboutin, Ziyi Yin, Philipp A. Witte, Ranveer Chandra, Russell J. Hewett, Felix J. Herrmann, Towards large-scale learned solvers for parametric pdes with model-parallel Fourier neural operators, arXiv preprint, arXiv:2204.01205, 2022.
[46] Steven Guan, Ko-Tsung Hsu, Parag V. Chitnis, Fourier neural operator networks: a fast and general solver for the photoacoustic wave equation, arXiv preprint, arXiv:2108.09374, 2021.
[47] John Guibas, Morteza Mardani, Zongyi Li, Andrew Tao, Anima Anandkumar, Bryan Catanzaro, Adaptive Fourier neural operators: efficient token mixers for transformers, arXiv preprint, arXiv:2111.13587, 2021.
[48] Anupam Gupta, Robert Krauthgamer, James R. Lee, Bounded geometries, fractals, and low-distortion embeddings, in: 44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings, 2003, pp. 534–543.
[49] Martin Hairer, An introduction to stochastic pdes, arXiv preprint, arXiv:0907.4178, 2009.
[50] Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, Dawn Song, Pretrained transformers improve out-of-distribution robustness, arXiv preprint, arXiv:2004.06100, 2020.
[51] Jan S. Hesthaven, Stefano Ubbiali, Non-intrusive reduced order modeling of nonlinear problems using neural networks, J. Comput. Phys. 363 (2018) 55–78, https://doi.org/10.1016/j.jcp.2018.02.037.
[52] Jerry L. Hintze, Ray D. Nelson, Violin plots: a box plot-density trace synergism, Am. Stat. 52 (2) (1998) 181–184.
[53] Songyan Hou, Parnian Kassraie, Anastasis Kratsios, Jonas Rothfuss, Andreas Krause, Instance-dependent generalization bounds via optimal transport, J. Mach. Learn. Res. 24 (2023) 1–50.
[54] G. Huang, S. Crawley, R. Djebbi, J. Ramos-Martinez, N. Chemingui, Deep learning velocity model building using Fourier neural operators, in: 84th EAGE Annual Conference & Exhibition, vol. 2023, European Association of Geoscientists & Engineers, 2023, pp. 1–5, No. 1.
[55] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, Kilian Q. Weinberger, Deep networks with stochastic depth, in: European Conference on Computer Vision, Springer, 2016, pp. 646–661.
[56] Marco A. Iglesias, A regularizing iterative ensemble Kalman method for pde-constrained inverse problems, Inverse Probl. 32 (2) (2016) 025002.
[57] Marco A. Iglesias, Kody J.H. Law, Andrew M. Stuart, Ensemble Kalman methods for inverse problems, Inverse Probl. 29 (4) (2013) 045001, https://doi.org/10.1088/0266-5611/29/4/045001.
[58] Daniel Jakubovitz, Raja Giryes, Miguel R.D. Rodrigues, Generalization error in deep learning, in: Compressed Sensing and Its Applications, Springer, 2019, pp. 153–193.
[59] Sham Kakade, Ambuj Tewari, Dudley's theorem, fat shattering dimension, packing numbers, Lecture 15, Toyota Technological Institute at Chicago, 2008.
[60] Barbara Kaltenbacher, Andreas Neubauer, Otmar Scherzer, Iterative regularization methods for nonlinear ill-posed problems, in: Iterative Regularization Methods for Nonlinear Ill-Posed Problems, De Gruyter, 2008.
[61] George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, Liu Yang, Physics-informed machine learning, Nat. Rev. Phys. 3 (6) (2021) 422–440, https://doi.org/10.1038/s42254-021-00314-5.
[62] Taeyoung Kim, Myungjoo Kang, Bounding the Rademacher complexity of Fourier neural operator, Mach. Learn. 113 (5) (2024) 2467–2498, https://doi.org/10.1007/s10994-024-06533-y.

[63] Georgios Kissas, Jacob H. Seidman, Leonardo Ferreira Guilhoto, Victor M. Preciado, George J. Pappas, Paris Perdikaris, Learning operators with coupled attention, J. Mach. Learn. Res. 23 (215) (2022) 1–63.
[64] Aryeh Kontorovich, Iosif Pinelis, Exact Lower Bounds for the Agnostic Probably-Approximately-Correct (pac) Machine Learning Model, 2019.
[65] Nikola Kovachki, Samuel Lanthaler, Siddhartha Mishra, On universal approximation and error bounds for Fourier neural operators, J. Mach. Learn. Res. 22 (2021).
[66] Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, Anima Anandkumar, Neural operator: learning maps between function spaces, arXiv preprint, arXiv:2108.08481, 2021.
[67] J. Kuelbs, A strong convergence theorem for Banach space valued random variables, Ann. Probab. 4 (5) (1976) 744–771.
[68] James Kuelbs, Wenbo V. Li, Metric entropy and the small ball problem for Gaussian measures, J. Funct. Anal. 116 (1) (1993) 133–157, https://doi.org/10.1006/jfan.1993.1107.
[69] Prashant Kumar, Gaussian random fields with matern covariance parametrization, https://github.com/pks19/Gaussian-random-fields, 2019.
[70] Thorsten Kurth, Shashank Subramanian, Peter Harrington, Jaideep Pathak, Morteza Mardani, David Hall, Andrea Miele, Karthik Kashinath, Animashree Anandkumar, Fourcastnet: accelerating global high-resolution weather forecasting using adaptive Fourier neural operators, arXiv preprint, arXiv:2208.05419, 2022.
[71] Samuel Lanthaler, Siddhartha Mishra, George E. Karniadakis, Error estimates for deeponets: a deep learning framework in infinite dimensions, Trans. Math. Appl. 6 (1) (2022) tnac001.
[72] Samuel Lanthaler, Roberto Molinaro, Patrik Hadorn, Siddhartha Mishra, Nonlinear reconstruction for operator learning of pdes with discontinuities, arXiv preprint, arXiv:2210.01074, 2022.
[73] Samuel Lanthaler, Zongyi Li, Andrew M. Stuart, The nonlocal neural operator: universal approximation, arXiv preprint, arXiv:2304.13221, 2023.
[74] J. Antonio Lara B, Florian Faucher, Xavier Tricoche, Official repo of the data set: fine tuning neural operators, https://rice.box.com/s/haczq8oad4b5cvi8pf8cp01sz4f0vfey, 2023.
[75] J. Antonio Lara B, Florian Faucher, Xavier Tricoche, Official repo: fine tuning neural operators, https://github.com/JALB-epsilon/Fine-tuning-NOs, 2023.
[76] Michel Ledoux, Michel Talagrand, Probability in Banach Spaces: Isoperimetry and Processes, reprint of the 1991 edition, Classics in Mathematics, Springer-Verlag, Berlin, ISBN 978-3-642-20211-7, 2011.
[77] James Lee-Thorp, Joshua Ainslie, Ilya Eckstein, Santiago Ontanon, Fnet: mixing tokens with Fourier transforms, arXiv preprint, arXiv:2105.03824, 2021.
[78] Bian Li, Hanchen Wang, Xiu Yang, Youzuo Lin, Solving seismic wave equations on variable velocity models with Fourier neural operator, arXiv preprint, arXiv:2209.12340, 2022.
[79] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, Tom Goldstein, Visualizing the loss landscape of neural nets, Adv. Neural Inf. Process. Syst. 31 (2018).
[80] Wenbo V. Li, Werner Linde, Approximation, metric entropy and small ball estimates for Gaussian measures, Ann. Probab. 27 (3) (1999) 1556–1578, https://doi.org/10.1214/aop/1022677459.
[81] Zijie Li, Kazem Meidani, Amir Barati Farimani, Transformer for partial differential equations' operator learning, arXiv preprint, arXiv:2205.13671, 2022.
[82] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, Anima Anandkumar, Fourier neural operator for parametric partial differential equations, arXiv preprint, arXiv:2010.08895, 2020.
[83] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, Anima Anandkumar, Neural operator: graph kernel network for partial differential equations, arXiv preprint, arXiv:2003.03485, 2020.
[84] Yiheng Liu, Tianle Han, Siyuan Ma, Jiayue Zhang, Yuanyuan Yang, Jiaming Tian, Hao He, Antong Li, Mengshen He, Zhengliang Liu, et al., Summary of chatgpt/gpt-4 research and perspective towards the future of large language models, arXiv preprint, arXiv:2304.01852, 2023.
[85] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, Baining Guo, Swin transformer: hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 10012–10022.
[86] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, Saining Xie, A convnet for the 2020s, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 11976–11986.
[87] Gabriel J. Lord, Catherine E. Powell, Tony Shardlow, An Introduction to Computational Stochastic PDEs, vol. 50, Cambridge University Press, 2014.
[88] Ilya Loshchilov, Frank Hutter, Decoupled weight decay regularization, arXiv preprint, arXiv:1711.05101, 2017.
[89] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, George Em Karniadakis, Learning nonlinear operators via deeponet based on the universal approximation theorem of operators, Nat. Mach. Intell. 3 (3) (2021) 218–229, https://doi.org/10.1038/s42256-021-00302-5.
[90] Lu Lu, Xuhui Meng, Shengze Cai, Zhiping Mao, Somdatta Goswami, Zhongqiang Zhang, George Em Karniadakis, A comprehensive and fair comparison of two neural operators (with practical extensions) based on fair data, Comput. Methods Appl. Mech. Eng. 393 (2022) 114778, https://doi.org/10.1016/j.cma.2022.114778.
[91] Carlo Marcati, Christoph Schwab, Exponential convergence of deep operator networks for elliptic partial differential equations, SIAM J. Numer. Anal. 61 (3) (2023) 1513–1545, https://doi.org/10.1137/21M1465718.
[92] Paul A. Martin, Time-Domain Scattering, vol. 180, Cambridge University Press, 2021.
[93] Pedro Henrique Martins, Zita Marinho, André F.T. Martins, ∞-former: infinite memory transformer, arXiv preprint, arXiv:2109.00301, 2021.
[94] David M. Mason, Zhan Shi, Small deviations for some multi-parameter Gaussian processes, J. Theor. Probab. 14 (1) (2001) 213–239, https://doi.org/10.1023/A:1007833401562.
[95] Roberto Molinaro, Yunan Yang, Björn Engquist, Siddhartha Mishra, Neural inverse operators for solving pde inverse problems, arXiv preprint, arXiv:2301.11167, 2023.
[96] Gen Nakamura, Roland Potthast, Inverse Modeling, IOP Publishing, 2015.
[97] Richard Nickl, Sven Wang, On polynomial-time computation of high-dimensional posterior measures by Langevin-type algorithms, arXiv preprint, arXiv:2009.05298, 2020.
[98] Jaideep Pathak, Shashank Subramanian, Peter Harrington, Sanjeev Raja, Ashesh Chattopadhyay, Morteza Mardani, Thorsten Kurth, David Hall, Zongyi Li, Kamyar Azizzadenesheli, et al., Fourcastnet: a global data-driven high-resolution weather model using adaptive Fourier neural operators, arXiv preprint, arXiv:2202.11214, 2022.
[99] Maziar Raissi, Paris Perdikaris, George E. Karniadakis, Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. 378 (2019) 686–707, https://doi.org/10.1016/j.jcp.2018.10.045.
[100] Yongming Rao, Wenliang Zhao, Zheng Zhu, Jiwen Lu, Jie Zhou, Global filter networks for image classification, Adv. Neural Inf. Process. Syst. 34 (2021) 980–993.
[101] Filippo Santambrogio, Optimal transport for applied mathematicians, in: Calculus of Variations, PDEs, and Modeling, vol. 87, 2015, xxvii+353, https://doi.org/10.1007/978-3-319-20828-2.
[102] Shai Shalev-Shwartz, Shai Ben-David, Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press, 2014.
[103] Andrew M. Stuart, Inverse problems: a Bayesian perspective, Acta Numer. 19 (2010) 451–559, https://doi.org/10.1017/S0962492910000061.
[104] Michel Talagrand, Sharper bounds for Gaussian and empirical processes, Ann. Probab. (1994) 28–76.
[105] Ilya O. Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al., Mlp-mixer: an all-mlp architecture for vision, Adv. Neural Inf. Process. Syst. 34 (2021) 24261–24272.
[106] William F. Trench, Conditional convergence of infinite products, Am. Math. Mon. 106 (7) (1999) 646–651.
[107] Tapas Tripura, Souvik Chakraborty, Wavelet neural operator: a neural operator for parametric partial differential equations, arXiv preprint, arXiv:2205.02191, 2022.

[108] Lan V. Truong, On Rademacher complexity-based generalization bounds for deep learning, arXiv preprint, arXiv:2208.04284, 2022.
[109] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, Illia Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).
[110] Cédric Villani, et al., Optimal Transport: Old and New, vol. 338, Springer, 2009.
[111] Martin J. Wainwright, High-Dimensional Statistics: A Non-Asymptotic Viewpoint, Cambridge Series in Statistical and Probabilistic Mathematics, Cambridge University Press, 2019.
[112] Gege Wen, Zongyi Li, Kamyar Azizzadenesheli, Anima Anandkumar, Sally M. Benson, U-fno—an enhanced Fourier neural operator-based deep-learning model for multiphase flow, Adv. Water Resour. 163 (2022) 104180.
[113] Gege Wen, Zongyi Li, Qirui Long, Kamyar Azizzadenesheli, Anima Anandkumar, Sally M. Benson, Accelerating carbon capture and storage modeling using Fourier neural operators, arXiv preprint, arXiv:2210.17051, 2022.
[114] Ross Wightman, Stochastic depth implementation, https://github.com/huggingface/pytorch-image-models/blob/a6e8598aaf90261402f3e9e9a3f12eac81356e9d/timm/models/layers/drop.py#L140.
[115] Yan Yang, Angela F. Gao, Jorge C. Castellanos, Zachary E. Ross, Kamyar Azizzadenesheli, Robert W. Clayton, Seismic wave propagation and inversion with neural operators, Seism. Rec. 1 (3) (2021) 126–134, https://doi.org/10.1785/0320210026.
[116] Ziyi Yin, Ali Siahkoohi, Mathias Louboutin, Felix J. Herrmann, Learned coupled inversion for carbon sequestration monitoring and forecasting with Fourier neural operators, arXiv preprint, arXiv:2203.14396, 2022.
[117] Kôsaku Yoshida, Functional Analysis. Classics in Mathematics, Springer, 1980.
[118] Huaiqian You, Quinn Zhang, Colton J. Ross, Chung-Hao Lee, Yue Yu, Learning deep implicit Fourier neural operators (ifnos) with applications to heterogeneous material modeling, arXiv preprint, arXiv:2203.08205, 2022.
[119] Weihao Yu, Mi Luo, Pan Zhou, Chenyang Si, Yichen Zhou, Xinchao Wang, Jiashi Feng, Shuicheng Yan, Metaformer is actually what you need for vision, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 10819–10829.
[120] Andrey Zhmoginov, Mark Sandler, Maksym Vladymyrov, Hypertransformer: model generation for supervised and semi-supervised few-shot learning, in: International Conference on Machine Learning, PMLR, 2022, pp. 27075–27098.