Generalization Bounds and Statistical Guarantees for Multi-Task and Multiple Operator Learning with MNO Networks

Adrien Weihs[1] and Hayden Schaeffer[1]

[1]Department of Mathematics,
University of California Los Angeles,
Los Angeles, CA 90095, USA.

Abstract

Multiple operator learning concerns learning operator families $\{G[\alpha] : U \to V\}_{\alpha \in W}$ indexed by an operator descriptor α. Training data are collected hierarchically by sampling operator instances α, then input functions u per instance, and finally evaluation points x per input, yielding noisy observations of $G[\alpha][u](x)$. While recent work has developed expressive multi-task and multiple operator learning architectures and approximation-theoretic scaling laws, quantitative statistical generalization guarantees remain limited. We provide a covering-number-based generalization analysis for separable models, focusing on the Multiple Neural Operator (MNO) architecture: we first derive explicit metric-entropy bounds for hypothesis classes given by linear combinations of products of deep ReLU subnetworks, and then combine these complexity bounds with approximation guarantees for MNO to obtain an explicit approximation-estimation tradeoff for the expected test error on new (unseen) triples (α, u, x). The resulting bound makes the dependence on the hierarchical sampling budgets (n_α, n_u, n_x) transparent and yields an explicit learning-rate statement in the operator-sampling budget n_α, providing a sample-complexity characterization for generalization across operator instances. The structure and architecture can also be viewed as a general purpose solver or an example of a "small" PDE foundation model, where the triples are one form of multi-modality.

Keywords and phrases. Deep Neural Networks, Generalization Bounds, Neural Scaling Laws, Operator Learning, Multi-Operator Learning, Multi-Task Problems.
Mathematics Subject Classification. 41A99, 68T07

1 Introduction

Operator learning seeks to approximate maps between function spaces, typically of the form $u \mapsto G[u]$, where the input u is a function and the output is another function (see [35, 49] and references therein). In many applications, however, the object of interest is not a single operator but a family of related operators indexed by a parameter. This motivates the recent *multiple operator learning* problem (see [61, 62, 64] and references therein), which we formalize as learning a map

$$G : W \longrightarrow \{G[\alpha] : U \to V\}_{\alpha \in W},$$

where W, U, V are function spaces, $\alpha \in W$ encodes the operator instance, and for each α, the corresponding operator $G[\alpha]$ maps inputs $u \in U$ to outputs in V. We highlight three prototypical settings in which multiple operator learning is either intrinsic to the problem formulation or would benefits from this formulation. We refer to Section 2.1 for a detailed description of each example class.

1. *Parameterized integral operators.* A basic example is a family of integral operators

$$G[\alpha][u](x) = \int K_\alpha(x, y)u(y)\,\mathrm{d}y \tag{1}$$

where the kernel K_α depends on a parametric function α. The problem is inherently a multi-operator learning task, as each parameter α induces a different (related) operator.

2. *Solution operators of parameterized PDEs.* Many simulations or forward problems in the physical sciences are naturally expressed through PDE solution operators: for each parameter α encoding, for instance, coefficients, boundary data, geometry, or even the governing equation, the map $G[\alpha]$ takes an input u (e.g. a forcing term, initial condition, or source) to the corresponding solution $G[\alpha][u]$.

3. *Operator families indexed by symbolic or textual descriptions.* More broadly, the "operator index" α can encode a task specification: a symbolic form, a natural-language prompt, or a discrete task label. This viewpoint connects multiple operator learning to PDE foundation models [45, 61], where a shared representation is trained across many operator instances and queried on new tasks by conditioning on an explicit operator description. In such regimes, the ability to generalize in α is important since one aims to predict accurate operators that are not seen during training.

Neural networks are a particularly well-suited approximation class for multiple operator learning because they combine high expressive power with architectural flexibility. Deep networks can capture nonlinear dependence on both the operator index α and the input function u, and they can be instantiated in forms that encode relevant inductive biases. At the same time, choosing an effective architecture is subtle: the network must allocate capacity between encoding how the operator varies with α and representing the action $u \mapsto G[\alpha][u]$, while also coping with discretization and the effective dimensionality of $W \times U \times \Omega_V$. A number of principled architectures have been proposed recently to address these challenges. Among them, the Multiple Neural Operator (MNO) architecture introduced in [64] achieves strong empirical performance across diverse operator families and is accompanied by expressivity guarantees and explicit scaling laws, providing both practical effectiveness and theoretical guidance for design.

Beyond approximation capabilities, it is important to understand how such architectures generalize when trained from finite data. Generalization bounds quantify how accurately a learned predictor will perform on unseen inputs drawn from the same data-generating process. They typically decompose into an approximation term, capturing the best achievable error within the chosen hypothesis class $\mathcal{F}$, and an estimation term, which depends on the statistical complexity of that class together with the available sample size. In particular, when complexity is controlled via covering numbers (we write $\mathcal{N}(\eta, \mathcal{F}, \|\cdot\|)$ for the η-covering number of a function class $\mathcal{F}$ with respect to a norm $\|\cdot\|$), the resulting estimation term involves the metric entropy $\log \mathcal{N}(\eta, \mathcal{F}, \|\cdot\|)$ at scale η, combined with sample-size factors (e.g. $1/\sqrt{n}$ or $1/n$), reflecting how generalization improves as more data are collected. To the best of our knowledge, this work provides the first generalization bound of this kind for multiple operator learning or multi-task neural operators.

In our setting, the generalization error concerns how a learned model achieves small expected error on new, unseen triples $(\alpha, u, x) \sim \mu_\alpha \times \mu_u \times \mu_x$, thereby controlling transfer across operator instances, input functions, and evaluation points simultaneously. We obtain such a bound for the MNO architecture by combining an approximation term (controlling the best-in-class approximation error ε through the results in [64]) with covering-number complexity control of the induced ε-dependent hypothesis class, yielding a transparent approximation–estimation tradeoff that links target accuracy, architectural complexity, and the sampling budgets (n_α, n_u, n_x) (numbers of sampled operators, inputs per operator, and evaluation points, respectively). In addition, the bound formalizes two practical benefits of the multiple operator learning viewpoint: (i) *amortization across operator instances*, since a single conditional model can share representations across an operator family rather than training one model per α; and (ii) *hierarchical sampling guidance*, since the explicit dependence on (n_α, n_u, n_x) clarifies how accuracy improves when increasing operator variability, inputs per operator, or evaluation resolution.

1.1 Contributions

Our main contributions are as follows:

1. We derive **metric entropy bounds** for function classes given by linear combinations of products of three deep ReLU subnetworks, which is precisely the separable structure underlying MNO. In particular, Proposition 3.1 provides an estimate as a function of the architectural parameters of each subnetwork class: depth L_i, width p_i, sparsity budget K_i, and parameter magnitude κ_i, as well as the product-structure multiplicities P, H, N.

2. Under Lipschitz regularity of the map G, we prove an **explicit scaling law for the expected generalization error** of the MNO architecture; see Theorem 3.5. The bound is stated for test triples $(\alpha, u, x) \sim \mu_\alpha \times \mu_u \times \mu_x$ and makes explicit the dependence on the hierarchical sampling budgets (n_α, n_u, n_x), on a prescribed target accuracy $\varepsilon > 0$ (achieved by an explicit ε-dependent choice of the MNO hypothesis class), and on a covering scale $\eta > 0$ through the metric entropy $\log \mathcal{N}(\eta)$. Specifically, it takes the schematic form:

$$\mathbb{E}_{\alpha,u,x}\left[\text{test error}\right] \lesssim \varepsilon^2 + \eta + \frac{\eta}{\sqrt{n_\alpha n_u n_x}}\sqrt{\log(\mathcal{N}(\eta))} + \frac{1}{n_\alpha n_u n_x}\log(\mathcal{N}(\eta)) + \frac{1}{n_\alpha}\log(\mathcal{N}(\eta))$$

3. As a consequence of the scaling law, we derive an **explicit sample-complexity rate** in the operator-sampling budget n_α; see Corollary 3.8. In particular, by selecting $\varepsilon = \varepsilon(n_\alpha)$ and $\eta = \eta(n_\alpha)$ to balance the approximation and estimation terms in Theorem 3.5, we obtain the rate

$$\mathbb{E}[\text{test error}] = \mathcal{O}\left(\left(\frac{\log\log(n_\alpha)}{\log\log\log(n_\alpha)}\right)^{-2/d_W}\right),$$

with constants independent of n_α, n_u, n_x.

This perspective is closely connected to recent PDE foundation models, where a single conditional architecture is trained across a broad family of PDEs and queried via an explicit operator descriptor. Our results provide indirect theoretical support for such approaches by establishing generalization guarantees and sample-complexity bounds for separable deep architectures that can themselves be approximated by PDE foundation models [45, 61].

1.2 Informal Statement of the Main Results

For ease of presentation, we state an informal version of the main results (Theorem 3.5 and Corollary 3.8), highlighting the dependence of the expected test error on the operator sampling budget and on the complexity of the induced hypothesis class; all assumptions and precise results are specified in the formal statements in Section 3.

Theorem 1.1 (Generalization error for MNO)**.** *Let $G : W \mapsto \{G[\alpha] : U \mapsto W\}_{\alpha \in W}$ be a Lipschitz multiple operator map from the function space W into Lipschitz operators from U to V. Assume that we observe sampled noisy data:*

$$y_{\ell ij} := G[\alpha_\ell][u_{\ell i}](x_{\ell ij}) + \zeta_{\ell ij}, \qquad 1 \le \ell \le n_\alpha,\ 1 \le i \le n_u,\ 1 \le j \le n_x,$$

where $\zeta_{\ell ij}$ denotes observation noise. For every $\varepsilon > 0$, there exists a MNO:

$$\sum_{p=1}^{P}\sum_{k=1}^{H}\sum_{\ell=1}^{N} \theta_{pk\ell}\, l_p(\alpha)\, b_k(u)\, \tau_\ell(x),$$

trained on $\{y_{\ell ij}\}$, whose expected test error on unseen triples (α, u, x) satisfies:

$$\mathcal{O}\left(\varepsilon^2 + \frac{1}{n_\alpha}\varepsilon^{-\delta_1 \varepsilon^{-\delta_2 \varepsilon^{-d_W}}}\log(n_\alpha)\right).$$

Here $\theta_{\ell ij} \in \mathbb{R}$, $\delta_i > 0$, d_W denotes the dimension of the domain of functions in W and the ReLU subnetworks l_p, b_k, and τ_ℓ can be chosen with the following architectural scalings:

	# networks	width	depth	sparsity	parameter magnitude
l_p	$P \lesssim \varepsilon^{-\varepsilon^{-d_W}}$	$O(1)$	$\lesssim \varepsilon^{-d_W}$	$\lesssim \varepsilon^{-d_W}$	$\lesssim \varepsilon^{-\varepsilon^{-d_W}}$
b_k	$H \lesssim \varepsilon^{-\varepsilon^{-\varepsilon^{-d_W}}}$	$O(1)$	$\lesssim \varepsilon^{-\varepsilon^{-d_W}}$	$\lesssim \varepsilon^{-\varepsilon^{-d_W}}$	$\lesssim \varepsilon^{-\varepsilon^{-\varepsilon^{-d_W}}}$
τ_ℓ	$N \lesssim \varepsilon^{-\varepsilon^{-d_W}}$	$O(1)$	$\lesssim \varepsilon^{-d_W}$	$\lesssim \varepsilon^{-d_W}$	$\lesssim \varepsilon^{-\varepsilon^{-d_W}}$

Moreover, choosing $\varepsilon \asymp \left(\frac{\log\log(n_\alpha)}{\log\log\log(n_\alpha)}\right)^{-\frac{1}{d_W}}$ *yields the generalization bound:*

$$\mathcal{O}\left(\left(\frac{\log\log(n_\alpha)}{\log\log\log(n_\alpha)}\right)^{-2/d_W}\right).$$

1.3 Related Works and Literature Review

Neural operator architectures Operator learning aim to approximate maps that take functions as inputs and return functions as outputs [4, 11, 27, 40, 42, 54, 58, 73]. Neural networks have proven effective for learning such mappings in a range of scientific and engineering settings [20,22,23,31,72]. A common design pattern in neural operator models is to split the processing of functional and spatial inputs across interacting subnetworks and recombine them via additive or bilinear/tensor-style contractions [13, 14]; DeepONet [48] is a canonical example, with a branch network that encodes the input function and a trunk network that produces a coordinate-dependent learned output basis. This "separable expansion" viewpoint connects operator networks to low-rank approximation ideas, where complex mappings are expressed as sums of simpler, lower-dimensional factors [52]; neural operators can be viewed as nonlinear, data-adaptive analogues of such decompositions. In the multiple-operator learning setting, MNO [64] adopts an analogous separable structure by separating operator identity (task/parameter) from input-function dependence through interacting subnetworks, enabling shared representations across a family of operators. Many other designs have emerged, including Fourier Neural Operators [42] (motivated by spectral representations), Green's function-based approaches such as Deep Green Networks [6, 18], and graph-based variants (including multipole constructions) that exploit sparsity and multiscale structure to improve efficiency [2,41]. For additional architectures and broader perspectives, see the surveys and references in [19, 35].

Multi-task and multiple operator learning Motivations for learning families of operators are twofold. In some settings, the underlying problem is naturally specified as a collection of related operators (e.g., indexed by physical parameters, geometries, or boundary conditions). In others, jointly learning multiple operators is a strategy for improving data efficiency and generalization by sharing structure across tasks. A growing body of recent work proposes multi-operator learning frameworks along these lines [3,8,24,29,46,47,53,61,64–69,71]. Notably, [47,61] demonstrate that multi-operator models can transfer to tasks beyond those encountered during training.

At a high level, there are two common formulations. One may (i) train separate operator models independently, one per task/operator instance, or (ii) treat the target as a parameterized operator family $\{G[\alpha]\}$, where a discrete or continuous descriptor α encodes the operator identity. The first approach does not condition on any explicit operator descriptor and therefore can struggle when the operator family varies substantially; in particular, it offers limited leverage for generalization to unseen operators. The second approach augments operator learning with an explicit operator encoding [45, 47, 55, 61, 64, 66], incorporating side information such as the governing equation, symbolic representation, textual description, or task label alongside the input functions. In this way, the second approach focuses on multi-task learning and general solvers. Providing this additional context typically strengthens transfer and has emerged as a key ingredient in recent PDE foundation model works. Conditioning on operator information enables zero-shot generalization to new PDE tasks, as demonstrated in [61], and such approaches have shown promising performance on out-of-distribution problems without expensive retraining.

Theoretical analyses of approximation and statistical generalization A central theoretical requirement in operator learning is expressivity, where universal approximation results ensure that a given architecture can approximate broad classes of operators to arbitrary accuracy. Early foundational work developing operator network constructions and proving universal approximation for mappings between spaces of scalar-valued functions was established in [13, 14]. Subsequent analyses extended these guarantees to widely used architectures, including DeepONet [37, 44], the Fourier Neural Operator [32], and PCA-Net [4], among others. Further developments related to operator expressivity, discretization effects, and architectural refinements include [9, 10, 26, 28, 33, 70, 72].

Beyond approximation-theoretic guarantees, scaling laws aim to quantify how error depends on data size, model capacity, and computational budget. Establishing a theoretical foundation for such laws provides a route to principled generalization estimates and predicts how performance should improve as resources increase [30]. Empirically, [15] studies cost–accuracy trade-offs across neural operator architectures, highlighting how network size and sampling budgets affect approximation error. On the theoretical side, [44] derives scaling laws and complexity estimates for deep ReLU networks and DeepONet. Related analyses for DeepONet and variants appear in [17, 25, 36–38, 50, 51, 59, 60]. Generalization error bounds for DeepONet and related models are developed in [39, 43, 44], while sample-complexity results are established in [1, 21, 34]. For multi-task and multiple operator learning, empirical evidence can be found in [29, 62]. Universal approximation results and expressivity scaling laws for MNO are derived in [64]; in this work, we establish generalization error estimates for MNO.

The remainder of the paper is structured as follows. In Section 2, we extensively formalize the multi-task and multiple operator learning setting and the MNO architecture, and collect the mathematical background needed for our analysis. In Section 3, we present our main theoretical results. In Section 4, we provide detailed proofs. Finally, in Section 5, we conclude with a summary of our contributions and discuss directions for future work.

2 Background

This section is organized into three parts. We begin with a collection of illustrative examples that motivate the multiple operator learning viewpoint and clarify the distinct roles of the parametric function (operator descriptor), the input function, and the evaluation variable. In this context, we also recall the MNO architecture, which makes this separation explicit by modeling the dependencies on α, u, and x through distinct components. Next, we summarize the scaling-law results for the MNO architecture that underpin the approximation component of our generalization analysis. Finally, we recall the covering-number estimates for the neural network classes used in our construction, which provide the complexity bounds needed for the estimation part of the proof.

2.1 Multi-Task Problems, Multiple Operator Learning, and the MNO Architecture

We start by introducing a general and flexible network class used in all of our subsequent constructions.

Definition 2.1 (Feedforward ReLU network class)**.** *Let $q : \mathbb{R}^{d_1} \to \mathbb{R}$ be a feedforward ReLU network defined as*

$$q(x) = W_L \cdot \mathrm{ReLU}\left(W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) + \cdots + b_{L-1}\right) + b_L,$$

where W_ℓ are weight matrices, b_ℓ are bias vectors, and $\mathrm{ReLU}(a) = \max\{a, 0\}$ is applied element-wise.

We define the class of such feedforward networks with ReLU activations:

$$\mathcal{F}_{\mathrm{NN}}(d_1, d_2, L, p, K, \kappa, R) = \left\{ [q_1, q_2, \ldots, q_{d_2}]^\top \in \mathbb{R}^{d_2} \;\middle|\; \begin{array}{l} \text{each } q_k : \mathbb{R}^{d_1} \to \mathbb{R} \text{ has the above form with} \\ L \text{ layers, width bounded by } p, \\ \|q_k\|_{\mathrm{L}^\infty} \le R, \quad \|W_\ell\|_{\infty,\infty} \le \kappa, \quad \|b_\ell\|_\infty \le \kappa, \\ \sum_{\ell=1}^{L} \left(\|W_\ell\|_0 + \|b_\ell\|_0\right) \le K \end{array} \right\},$$

where

- $\|q\|_{\mathrm{L}^\infty} = \sup_{x \in \Omega} |q(x)|$,
- $\|W_\ell\|_{\infty,\infty} = \max_{i,j} |[W_\ell]_{ij}|$,
- $\|b_\ell\|_\infty = \max_i |[b_\ell]_i|$,
- $\|\cdot\|_0$ *denotes the number of nonzero elements.*

This network class consists of vector-valued functions with input dimension d_1, output dimension d_2, depth L, width at most p, at most K nonzero parameters, all bounded in magnitude by κ, and uniformly bounded output norm by R.

We recall our goal of approximating a multiple-operator map $G : W \to \{G[\alpha] : U \to V\}_{\alpha \in W}$, where W, U, V are function spaces (with underlying domains $\Omega_W, \Omega_U, \Omega_V$, respectively). The MNO architecture introduced in [64] provides an effective and structurally aligned model class for this problem.

Definition 2.2 (MNO Architecture)**.** *For fixed positive integers $P, H^{(p)}$, $1 \le p \le P$, we define a* MNO *as*

$$\mathrm{MNO}[\alpha][u](x) = \sum_{p=1}^{P} \sum_{k=1}^{H^{(p)}} l_p(\boldsymbol{\alpha}) b_{pk}(\boldsymbol{u}) \tau_{pk}(x)$$

for $\alpha \in W$, $u \in U$, and $x \in \Omega_V$, where l_p, b_{pk}, and τ_{pk} are neural networks in suitable classes $\mathcal{F}_{\mathrm{NN}}$, and $\boldsymbol{\alpha}$, $\boldsymbol{u}$ denote discretizations of α and u, respectively.

It is shown in [64, Remark 3.20] that MNO is a special case of the more general fully separable architecture

$$\sum_{p=1}^{P} \sum_{k=1}^{H} \sum_{\ell=1}^{N} \theta_{pk\ell}\, l_p(\boldsymbol{\alpha})\, b_k(\boldsymbol{u})\, \tau_\ell(x), \qquad \theta_{pk\ell} \in \mathbb{R}.$$

Our analysis is formulated for this fully separable class, since it is more convenient for approximation and covering-number estimates. The resulting bounds transfer to MNO by a standard re-indexing (equivalently, a rearrangement of subnetworks).

We next revisit the three representative classes of examples from the introduction, now in a form that makes their connection to the MNO architecture explicit. Collectively, these examples illustrate the following recurring themes:

- *Distinct roles of the inputs.* The operator descriptor α, the input function u, and the evaluation variable x play fundamentally different roles: α specifies the operator instance (task), u is the operand acted upon by that operator, and x is the query location at which the output is evaluated.
- *Shared structure and computational efficiency.* Many operator families exhibit reusable structure across α. Learning these families jointly allows one to represent common components once and reuse them across parameter regimes, amortizing training and reducing redundant approximation effort compared to training separate models per operator.
- *Generalization across operator instances.* The objective in multiple operator learning is not merely to fit finitely many operators, but to learn a map $\alpha \mapsto G[\alpha]$ that generalizes to new (possibly unseen) $\alpha \in W$, enabling transfer to new coefficients, boundary conditions, or related tasks without retraining.
- *Breadth of applications.* The same multiple operator learning viewpoint arises across diverse settings, including parameterized kernel operators, PDE solution operators, and task-conditioned operator families.

These principles directly motivate the MNO architecture. MNO is built to respect the separation of roles of α, u, and x through a separable structure, in which shared subnetworks l, b, and τ encode the dependence on the operator descriptor, the input function, and the evaluation variable, respectively. In particular, generalization across $\alpha \in W$ is a natural requirement in this formulation, since α enters the model as an explicit input and the learned predictor is defined for any admissible α within the specified domain.

Parametrized integral operators

Example 2.3 (Homogeneous kernels with parameter-dependent interaction radius). Assume that $\Omega_W = \Omega_V$ and let $\alpha : \Omega_V \to (0, \infty)$ represent a spatially varying interaction length scale. Consider kernels of the form

$$K_\alpha(x, y) = \frac{1}{\alpha(x)^d}\, \rho\!\left(\frac{|x - y|}{\alpha(x)}\right),$$

where $\rho : [0, \infty) \to \mathbb{R}$ is a prescribed radial profile. The associated operator is

$$G[\alpha][u](x) = \int_{\Omega_U} \frac{1}{\alpha(x)^d}\, \rho\!\left(\frac{|x - y|}{\alpha(x)}\right) u(y)\, \mathrm{d}y.$$

Such parameterized kernel operators are a standard building block in nonlocal models (e.g. [63]), and, under appropriate scalings of the kernel profile ρ, they can be used to approximate local differential operators [7].

Example 2.4 (Variable-order fractional kernel operators). Another important class of examples is provided by variable-order fractional kernels arising in nonlocal models and fractional calculus [16]. Assume $\Omega_W = \Omega_V$, and let $\alpha : \Omega_V \to (0, 1)$ be a spatially varying order function. Define

$$K_\alpha(x, y) = \frac{c_{d,\alpha(x)}}{|x - y|^{d+2\alpha(x)}},$$

where $c_{d,\alpha(x)}$ denotes a normalization constant depending on the spatial dimension d and the local fractional order $\alpha(x)$. The associated operator is

$$G[\alpha][u](x) = \int_{\Omega_U} \frac{c_{d,\alpha(x)}}{|x - y|^{d+2\alpha(x)}} u(y)\, \mathrm{d}y.$$

This example makes the role separation in multiple operator learning particularly transparent. Specifically, the functions α and u play fundamentally different roles, and therefore representing the operator as $G[\alpha, u]$, thereby treating (α, u) as a single concatenated input in a classical operator-learning formulation may obscure their structural distinction. The function α determines the kernel K_α and thereby defines the integration rule itself (namely how points interact, the weighting structure, and the relevant length scales). Changing α modifies the action under consideration and thus characterizes the task. In contrast, the function u represents the data being processed; it is the input integrated against the kernel, analogous to a signal or state on which the operator acts. If u varies while α remains fixed, the rule and therefore the task remains unchanged, and only the input within that task is varied. MNO and related multi-operator learning architectures directly mimic this hierarchical structure by explicitly separating these inputs, which models their respective roles in a mathematically consistent fashion.

Solution operator of parametrized PDEs

Example 2.5 (Green-kernel representation of a parameterized PDE solution operator). For broad classes of well-posed linear boundary-value problems, the solution operator admits the integral-kernel representation (1) where K_α is the Green's kernel associated with the parameterized differential operator.

For example, consider the boundary value problem:

$$-v''(x) = u(x), \qquad 0 < x < a, \qquad v(0) = 0, \qquad v(a) = 0,$$

where $a > 0$ and $u : (0, a) \to \mathbb{R}$ is a given source term. For every $u \in \mathrm{L}^2(0, a)$, classical elliptic regularity theory implies the existence of a unique weak solution:

$$v \in \mathrm{H}_0^1(0, a) \cap \mathrm{H}^2(0, a).$$

We denote the corresponding solution operator by:

$$G[a] : \mathrm{L}^2 \to \mathrm{H}_0^1(0, a) \cap \mathrm{H}^2(0, a)$$

which is the inverse of $-\dfrac{d^2}{dx^2}$ on $\mathrm{H}_0^1(0, a)$. The solution admits the Green representation

$$v(x) = G[a][u](x) = \int_0^a K_a(x, y)\, u(y)\, dy, \qquad 0 < x < a,$$

where the Dirichlet Green's kernel is given by

$$K_a(x, y) = \frac{1}{a} \begin{cases} x\,(a - y), & 0 \le x \le y \le a, \\ y\,(a - x), & 0 \le y < x \le a. \end{cases}$$

Equivalently, for $0 \le x, y \le a$, this can be written through ReLUs:

$$K_a(x, y) = \frac{x + y}{2} - \frac{\mathrm{ReLU}(x - y) + \mathrm{ReLU}(y - x)}{2} - \frac{1}{a} xy,$$

where $\mathrm{ReLU}(z) = \max\{0, z\}$. Extending the integral to the interval $[0, 1]$ using the Heaviside function H, we obtain:

$$\begin{aligned} G[a][u](x) &= \int_0^1 H(a-y)\, K_a(x,y)\, u(y)\, dy \\ &= \int_0^1 \ell^{(1)}(a,y)\, \tau^{(1)}(x,y)\, u(y)\, dy \;+\; \int_0^1 \ell^{(2)}(a,y)\, \tau^{(2)}(x,y)\, u(y)\, dy, \end{aligned}$$

where the functions are separated into:

$$\ell^{(1)}(a,y) = H(a-y), \qquad \ell^{(2)}(a,y) = -\frac{H(a-y)}{a},$$

and

$$\tau^{(1)}(x,y) = \frac{x+y}{2} - \frac{\mathrm{ReLU}(x-y) + \mathrm{ReLU}(y-x)}{2}, \qquad \tau^{(2)}(x,y) = xy.$$

This representation for $G[a][u]$ shows that the operator is in fact a finite sum of separable kernel components. Letting $\alpha : \Omega_V \to \mathbb{R}$ be the constant function a, a Monte Carlo quadrature with sampling nodes $Y_i \sim \mathrm{Unif}(0,1)$ yields:

$$G[\alpha][u](x) \approx \sum_{p=1}^{2} \sum_{i=1}^{N} \ell^{(p)}(\alpha, Y_i)\, b_i(u)\, \tau^{(p)}(x, Y_i),$$

where

$$\begin{aligned} \ell^{(1)}(\alpha, Y_i) &= \frac{H(\alpha - Y_i)}{N}, \qquad \ell^{(2)}(\alpha, Y_i) = -\frac{H(\alpha - Y_i)}{N\alpha}, \qquad b_i(u) = u(Y_i), \\ \tau^{(1)}(x, Y_i) &= \frac{x+Y_i}{2} - \frac{\mathrm{ReLU}(x-Y_i) + \mathrm{ReLU}(Y_i - x)}{2}, \qquad \tau^{(2)}(x, Y_i) = xY_i. \end{aligned}$$

The Monte Carlo approximation therefore yields the same structural form as the MNO ansatz, i.e., a finite sum of separable (low-rank) components where the nodes Y_i can be can be absorbed into the network parameters. The functions $\ell^{(p)}$ encode the dependence on the operator parameter α, the coefficients b_i encode the dependence on the input function u through point evaluations, and the functions $\tau^{(p)}$ encode the dependence on the output variable x. In this sense, the Green's solution provides an explicit, kernel-based realization of the separable operator structure. Similar decompositions arise for other linear PDE through their Green's formulations.

Example 2.6 (Nonlinear PDE solution operator with a shared semigroup structure). For nonlinear PDEs, the solution map $u \mapsto G[\alpha][u]$ is typically nonlinear and therefore cannot, in general, be represented by a single kernel acting linearly on u. We illustrate the idea of multi-operator approximations using a parameterized family of equations. Let $\alpha = (\sigma, \nu)$ be the model parameters, where

$$\sigma \in \{0, 1\} \quad \text{and} \quad \nu > 0,$$

and consider the PDE

$$\partial_t z + \sigma\, z\, \partial_x z = \nu\, \partial_{xx} z, \qquad z(0,x) = u(x), \qquad x \in \mathbb{R}.$$

When $\sigma = 0$, this reduces to the linear heat equation; when $\sigma = 1$, it is the viscous Burgers equation. We denote the corresponding solution operator by

$$u \to G[\alpha][u].$$

The linear heat equation setting occurs when $\alpha = (0, \nu)$, and the solution is given by the heat semigroup:

$$S_t^{\nu}[u](x) := \int_{\mathbb{R}} \Gamma_\nu(t, x-y)\, u(y)\, dy,$$

where

$$\Gamma_\nu(t,z) = \frac{1}{\sqrt{4\pi\nu t}} \exp\left(-\frac{z^2}{4\nu t}\right).$$

Thus, the heat operator is given by: $G[(0,\nu)][u] = S_t^\nu[u]$. For the Burgers' case ($\sigma = 1$), if $\alpha = (1,\nu)$, the Cole-Hopf transformation

$$\phi(t,x) = \exp\left(-\frac{1}{2\nu}\int_0^x z(t,\xi)\,d\xi\right)$$

reduces the equation to the heat equation

$$\partial_t\phi = \nu\,\partial_{xx}\phi,$$

and hence,

$$\phi(t,x) = \int_{\mathbb{R}} \Gamma_\nu(t,x-y)\,\exp\!\left(-\frac{1}{2\nu}\int_0^y u(\xi)\,d\xi\right)dy.$$

Transforming back yields

$$G[(1,\nu)][u](x) = -2\nu\,\partial_x \log\phi(t,x),$$

or equivalently,

$$G[(1,\nu)][u](x) = \frac{1}{t}\frac{\int_{\mathbb{R}}(x-y)\,\Gamma_\nu(t,x-y)\,E_u^\nu(y)\,dy}{\int_{\mathbb{R}}\Gamma_\nu(t,x-y)\,E_u^\nu(y)\,dy},$$

where

$$E_u^\nu(y) = \exp\!\left(-\frac{1}{2\nu}\int_0^y u(\xi)\,d\xi\right).$$

In both cases, the same linear heat semigroup $S_t^\nu[u](x)$ appears as a common sub-operator. The main difference between the two PDEs' solution operator lies only in nonlinear input/output transformations. Define

$$\mathcal{P}_{\sigma,\nu}[u] = \begin{cases} u, & \sigma = 0,\\ E_u^\nu, & \sigma = 1,\end{cases} \qquad \mathcal{T}_{\sigma,\nu}[\phi] = \begin{cases} \phi, & \sigma = 0,\\ -2\nu\,\partial_x\log\phi, & \sigma = 1.\end{cases}$$

Then both solution operators admit the unified representation

$$G[\alpha] = \mathcal{T}_{\sigma,\nu}\circ S_t^\nu \circ \mathcal{P}_{\sigma,\nu}, \qquad \alpha = (\sigma,\nu).$$

If generating the solutions to the two PDEs are modeled by independent by neural networks, as in the single operator learning case, one constructs

$$\mathrm{NN}_{(0,\nu)} \approx S_t^\nu, \qquad \mathrm{NN}_{(1,\nu)} \approx \mathcal{T}_{1,\nu}\circ S_t^\nu\circ\mathcal{P}_{1,\nu}.$$

In this case, the heat semigroup S_t^ν must effectively be learned twice. If $\mathcal{C}(S_t^\nu)$ denotes the approximation complexity (e.g., rank, width, or parameter count) required to represent S_t^ν, then the total complexity scales like:

$$2\,\mathcal{C}(S_t^\nu) + \mathcal{C}_{\mathrm{nonlinear}}.$$

By contrast, in a simultaneous (multi-operator) setting, the family:

$$u \to G[\alpha][u], \qquad \alpha = (\sigma,\nu),$$

is learned jointly. The common operator S_t^ν is approximated once, while only the lower-complexity transformations $\mathcal{P}_{\sigma,\nu}$ and $\mathcal{T}_{\sigma,\nu}$ depend on the equation type. The resulting complexity scales as

$$\mathcal{C}(S_t^\nu) + \mathcal{C}_{\mathrm{nonlinear}},$$

which is significantly smaller whenever the representation of the heat semigroup dominates the approximation cost. Thus, simultaneous operator learning leverages the shared linear propagator indexed by ν, whereas disjoint learning redundantly approximates the same semigroup structure for each equation type.

Operator families indexed by symbolic or textual descriptions

Example 2.7 (PROSE architecture). The family of networks defined by the PROSE architecture [45,47] learns nonlinear operators $G[\alpha][u](x)$ using a multimodal Transformer framework. The objective is to approximate an operator where α denotes auxiliary (e.g., symbolic, text, or parametric) information and u denotes a function represented through sampled observations.

In PROSE, the inputs α and u are first mapped into a shared latent space via separate MLP encoders, $\tilde{\alpha} = \Phi_\alpha(\alpha)$, $\tilde{u} = \Phi_u(u)$, where Φ_α and Φ_u are learned nonlinear embeddings. The encoded tokens are then concatenated to form $S = [\tilde{\alpha}, \tilde{u}] \in \mathbb{R}^{2\times d}$. A self-attention layer integrates information across modalities:

$$Y = \mathrm{SelfAttention}(S) = \mathrm{softmax}\left(\frac{(SW_Q^{(s)})(SW_K^{(s)})^\top}{\sqrt{d_k}}\right)(SW_V^{(s)}),$$

where $W_Q^{(s)}, W_K^{(s)}, W_V^{(s)}$ are learned projection matrices and the softmax is applied row-wise. This step produces fused latent representations $Y \in \mathbb{R}^{2\times d_v}$. The PROSE framework resembles an operator since the output function can be evaluate at query location x as follows. The query embedding $\tilde{x} = \Phi_x(x)$, is used in a cross-attention mechanism with keys and values derived from the processed inputs Y:

$$\mathrm{CrossAttention}(\tilde{x}, Y) = \mathrm{softmax}\left(\frac{(\tilde{x}W_Q^{(c)})(YW_K^{(c)})^\top}{\sqrt{d_k}}\right)(YW_V^{(c)}).$$

The resulting output is mapped through decoder Ψ (e.g., a simple MLP) to produce the scalar (or vector-valued) operator evaluation,

$$G[\alpha][u](x) = \Psi\left(\mathrm{CrossAttention}(\tilde{x}, Y)\right).$$

While the above description uses a single-head, in practice [45,47] a multi-head formulation is used. This generalizes the single-head formulation by introducing head-indexed projection matrices $\{W_{Q,h}, W_{K,h}, W_{V,h}\}_{h=1}^H$, computing attention independently across heads, and aggregating the resulting representations via concatenation followed by a learned output projection. Mathematically,

$$\mathrm{MultiHead}(S) = \mathrm{Concat}\big(\mathrm{head}_1(S), \ldots, \mathrm{head}_H(S)\big)W_O,$$

where for each head h,

$$\mathrm{head}_h(S) = \mathrm{softmax}\left(\frac{(SW_{Q,h})(SW_{K,h})^\top}{\sqrt{d_k}}\right)(SW_{V,h}),$$

and W_O denotes the output projection matrix. These architectures illustrate that generalization across operator instances α is naturally incorporated by treating α as an explicit input modality.

2.2 Scaling Laws for Multiple Operator Learning

In this section, we review the approximation-theoretic results underlying the MNO architecture, which provide the approximation term in Theorem 3.5 and motivate the ε-dependent instantiation of the network hypothesis class used throughout the proof of the latter. Specifically, we begin by recalling [64, Theorem 3.16], which establishes scaling laws for the expressivity of a general multiple operator architecture; the corresponding scaling laws for MNO follow as a special case.

Theorem 2.8 (Multiple Operator Scaling Laws)**.** *Let $d_W, d_U, d_V > 0$ be integers,*

$$\gamma_W, \gamma_U, \gamma_V, \beta_W, \beta_U, \beta_V, L_W, L_U, L_V, L_G, L_{\mathcal{G}} > 0 \qquad \textit{and} \qquad r_G, r_{\mathcal{G}} \geq 1$$

*and assume that $W(d_W, \gamma_W, L_W, \beta_W)$, $U(d_U, \gamma_U, L_U, \beta_U)$ and $V(d_V, \gamma_V, L_V, \beta_V)$ satisfy Assumption **S.**. Let G be a map such that*

$$G : \{\alpha : \Omega_W \mapsto \mathbb{R} \,|\, \|\alpha\|_{\mathrm{L}^\infty} \leq \beta_W\} \mapsto \mathcal{G} \qquad \textit{where}$$

$$\mathcal{G} = \Big\{ G[\alpha] \,|\, G[\alpha] : \{u : \Omega_U \mapsto \mathbb{R} \,|\, \|u\|_{\mathrm{L}^\infty} \le \beta_U\} \mapsto V \text{ and } \\ \|G[\alpha][u_1] - G[\alpha][u_2]\|_{\mathrm{L}^\infty(\Omega_V)} \le L_{\mathcal{G}} \|u_1 - u_2\|_{\mathrm{L}^{r_{\mathcal{G}}}(\Omega_U)} \Big\}$$

Furthermore, assume that G satisfies

$$\|G(\alpha_1) - G(\alpha_2)\|_{\mathrm{L}^\infty(\{u:\Omega_U \mapsto \mathbb{R} \,|\, \|u\|_{\mathrm{L}^\infty} \le \beta_U\} \times \Omega_V)} \le L_G \|\alpha_1 - \alpha_2\|_{\mathrm{L}^{r_G}(\Omega_W)}$$

for $\alpha_1, \alpha_2 \in \{\alpha : \Omega_W \mapsto \mathbb{R} \,|\, \|\alpha\|_{\mathrm{L}^\infty} \le \beta_W\}$.

There exists constants C depending on γ_V, L_V, C_δ depending on $L_{\mathcal{G}}, d_U, \gamma_U, r_{\mathcal{G}}, L_U$, C' depending on $\beta_U, L_{\mathcal{G}}, d_U, \gamma_U, r_{\mathcal{G}}$, C_ζ depending on $L_G, d_W, \gamma_W, r_G, L_W$ and C'' depending on $\beta_W, L_G, d_W, \gamma_W, r_G$ such that the following holds. For any $\varepsilon > 0$,

- *let $N = 2^{n_{c_W}+2} C \sqrt{d_V} (C'' \sqrt{n_{c_W}})^{n_{c_W}} \varepsilon^{-(n_{c_W}+1)}$ and consider the network class $\mathcal{F}_1 = \mathcal{F}_{\mathrm{NN}}(d_V, 1, L_1, p_1, K_1, \kappa_1, R_1)$ with parameters scaling as*

$$\begin{aligned} L_1 &= \mathcal{O}\left(d_V^2 \log d_V + d_V^2 (n_{c_W}+1) \log(\varepsilon^{-1}) + d_V^2 \log(2^{n_{c_W}+1} (C'' \sqrt{n_{c_W}})^{n_{c_W}}) + d_V^2 \log(2)\right), \\ p_1 &= \mathcal{O}(1), \\ K_1 &= \mathcal{O}\left(d_V^2 \log d_V + d_V^2 (n_{c_W}+1) \log(\varepsilon^{-1}) + d_V^2 \log(2^{n_{c_W}+1} (C'' \sqrt{n_{c_W}})^{n_{c_W}}) + d_V^2 \log(2)\right) \\ \kappa_1 &= \mathcal{O}(d_V^{d_V/2+1} \varepsilon^{-(d_V+1)(n_{c_W}+1)} \left[2^{n_{c_W}+2} (C'' \sqrt{n_{c_W}})^{n_{c_W}}\right]^{(d_V+1)}), \qquad R_1 = 1 \end{aligned}$$

  where the constants hidden in $\mathcal{O}$ depend on γ_V and L_V;

- *let $\{v_\ell\}_{\ell=1}^{N^{d_V}} \subset \Omega_V$ be a uniform grid with spacing $2\gamma_V / N$ along each dimension;*

- *let $\delta = \frac{C_\delta \varepsilon^{(1+d_V)(1+n_{c_W})}}{2^{d_V+n_{c_W}+2} (C\sqrt{d_V})^{d_V} (C'' \sqrt{n_{c_W}})^{n_{c_W}}}$ and let $\{c_m\}_{m=1}^{n_{c_U}} \subset \Omega_U$ be points so that $\{\mathcal{B}_\delta(c_m)\}_{m=1}^{n_{c_U}}$ is a cover of Ω_U for some n_{c_U};*

- *let $H = 2^{(d_V+1)(n_{c_W}+2)} C' \sqrt{n_{c_U}} (C\sqrt{d_V})^{d_V} (C'' \sqrt{n_{c_W}})^{n_{c_W}(d_V+1)} \varepsilon^{-(d_V+1)(1+n_{c_W})}$ and consider the network class $\mathcal{F}_2 = \mathcal{F}_{\mathrm{NN}}(n_{c_U}, 1, L_2, p_2, K_2, \kappa_2, R_2)$ with parameters scaling as*

$$\begin{aligned} L_2 &= \mathcal{O}\big(n_{c_U}^2 \log n_{c_U} + n_{c_U}^2 (d_V+1)(n_{c_W}+1) \log(\varepsilon^{-1}) + n_{c_U}^2 \log(2^{d_V+1} (C\sqrt{d_V})^{d_V}) \\ &\quad + n_{c_U}^2 (d_V+1) \log(2^{n_{c_W}+1} (C'' \sqrt{n_{c_W}})^{n_{c_W}}) + n_{c_U}^2 \log(2)\big), \quad p_2 = \mathcal{O}(1), \\ K_2 &= \mathcal{O}\big(n_{c_U}^2 \log n_{c_U} + n_{c_U}^2 (d_V+1)(n_{c_W}+1) \log(\varepsilon^{-1}) + n_{c_U}^2 \log(2^{d_V+1} (C\sqrt{d_V})^{d_V}) \\ &\quad + n_{c_U}^2 (d_V+1) \log(2^{n_{c_W}+1} (C'' \sqrt{n_{c_W}})^{n_{c_W}}) + n_{c_U}^2 \log(2)\big), \\ \kappa_2 &= \mathcal{O}(n_{c_U}^{n_{c_U}/2+1} \varepsilon^{-(d_V+1)(n_{c_U}+1)(n_{c_W}+1)} [2^{d_V+2} (C\sqrt{d_V})^{d_V}]^{n_{c_U}+1} \left[2^{d_V+1} (C\sqrt{d_V})^{d_V}\right]^{(d_V+1)(n_{c_U}+1)}), \\ R_2 &= 1 \end{aligned}$$

  where the constants hidden in $\mathcal{O}$ depend on $\beta_U, L_{\mathcal{G}}, d_U, \gamma_U, r_{\mathcal{G}}$;

- *let $\zeta = C_\zeta \varepsilon$ and let $\{y_m\}_{m=1}^{n_{c_W}} \subset \Omega_W$ be points so that $\{\mathcal{B}_\zeta(y_m)\}_{m=1}^{n_{c_W}}$ is a cover of Ω_W for some n_{c_W};*

- *let $P = 2C'' \sqrt{n_{c_W}} \varepsilon^{-1}$ and consider the network class $\mathcal{F}_3 = \mathcal{F}_{\mathrm{NN}}(n_{c_W}, 1, L_3, p_3, K_3, \kappa_3, R_3)$ with parameters scaling as*

$$\begin{aligned} L_3 &= \mathcal{O}\left(n_{c_W}^2 \log(n_{c_W}) + n_{c_W}^2 \log(\varepsilon^{-1}) + n_{c_W}^2 \log(2)\right), \quad p_3 = \mathcal{O}(1), \\ K_3 &= \mathcal{O}\left(n_{c_W}^2 \log n_{c_W} + n_{c_W}^2 \log(\varepsilon^{-1}) + n_{c_W}^2 \log(2)\right), \\ \kappa_3 &= \mathcal{O}(n_{c_W}^{n_{c_W}/2+1} 2^{n_{c_W}+1} \varepsilon^{-n_{c_W}-1}), \qquad R_3 = 1 \end{aligned}$$

  where the constants hidden in $\mathcal{O}$ depend on $\beta_W, L_G, d_W, \gamma_W, r_G$.

Then, there exists networks $\{\tau_\ell\}_{\ell=1}^{N^{d_V}} \subset \mathcal{F}_1$*, networks* $\{b_k\}_{k=1}^{H^{n_{c_U}}} \subset \mathcal{F}_2$*, networks* $\{l_p\}_{p=1}^{P} \subset \mathcal{F}_3$*, functions* $\{u_k\}_{k=1}^{H^{n_{c_U}}} \subset \{u : \Omega_U \mapsto \mathbb{R} \,|\, \|u\|_{\mathrm{L}^\infty} \leq \beta_U\}$ *and functions* $\{\alpha_p\}_{p=1}^{P} \subset \{\alpha : \Omega_W \mapsto \mathbb{R} \,|\, \|\alpha\|_{\mathrm{L}^\infty} \leq \beta_W\}$ *such that*

$$\sup_{\alpha\in W}\sup_{u\in U}\sup_{x\in\Omega_V}\left|G[\alpha][u](x) - \sum_{p=1}^{P^{n_{c_W}}}\sum_{k=1}^{H^{n_{c_U}}}\sum_{\ell=1}^{N^{d_V}} G[\alpha_p][u_k](v_\ell)l_p(\boldsymbol{\alpha})b_k(\mathbf{u})\tau_\ell(x)\right| \leq \varepsilon, \tag{2}$$

where $\boldsymbol{\alpha} = (\alpha(y_1), \alpha(y_2), ..., \alpha(y_{n_{c_W}}))^\top$ *is a discretization of* α *and* $\mathbf{u} = (u(c_1), u(c_2), ..., u(c_{n_{c_U}}))^\top$ *is a discretization of* u*.*

Next, for a function $f : \mathbb{R} \to \mathbb{R}$ and a constant $a \geq 0$, we define the *clipping operator* that constrains the range of f to the interval $[-a, a]$:

$$\mathrm{Clip}_a(f) \;=\; \min\{\max(f, -a),\, a\}.$$

This truncation can be implemented exactly by a two–layer ReLU network. One explicit realization is

$$\mathrm{Clip}_a(f) = -\mathrm{ReLU}\left(-\mathrm{ReLU}(f + a) + 2a\right) + a,$$

which expresses the clipping operation using only affine maps and ReLU activations. In particular, such a network is in the class $\mathcal{F}_{\mathrm{NN}}(1, 1, 2, 1, 6, 2a, a)$.

The next result incorporates the clipping operation into the general multiple operator architecture and shows that the resulting clipped network class admits analogous expressivity guarantees and scaling laws.

Corollary 2.9 (Clipped network scaling laws)**.** *Assume the same setting as in Theorem 2.8. Let* $\mathrm{NN}[\boldsymbol{\alpha}][\boldsymbol{u}](x)$ *be the network such that* (2) *holds for* $\varepsilon/2$*. Then,*

$$\sup_{\alpha\in W}\sup_{u\in U}\sup_{x\in\Omega_V}\left|G[\alpha][u](x) - \mathrm{Clip}_{\beta_V}\left(\mathrm{NN}[\boldsymbol{\alpha}][\boldsymbol{u}](x)\right)\right| \leq \varepsilon. \tag{3}$$

In view of Corollary 2.9, clipping can be incorporated without loss of approximation power (up to adjustment of the target accuracy). Accordingly, we henceforth take our hypothesis class to consist of clipped multiple operator networks. This boundedness property is a technical ingredient in the generalization analysis used in the proof of Theorem 3.5. We formalize the resulting model class in the following definition.

Definition 2.10 (Clipped multiple operator network class)**.** *Let* $\mathcal{F}_i$ *for* $1 \leq i \leq 3$ *be network classes defined in Definition 2.1. For* $a, I > 0$*,* $P, H, N \in \mathbb{N}$ *and fixed sampling points* $\{y_s\}_{s=1}^{n_{c_W}} \subset \Omega_W$ *and* $\{c_s\}_{s=1}^{n_{c_U}}$*, we define* $\mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N)$*, the set of* a*-clipped multiple operator networks, as*

$$\begin{aligned}&\mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N)\\ &= \left\{ \mathrm{Clip}_a\left(\sum_{p=1}^{P}\sum_{k=1}^{H}\sum_{\ell=1}^{N}\theta_{pk\ell}l_p(\boldsymbol{\alpha})b_k(\mathbf{u})\tau_\ell(x)\right) \mid \theta_{pk\ell} \in [-I, I],\, \tau_\ell \in \mathcal{F}_1,\, b_k \in \mathcal{F}_2,\, l_p \in \mathcal{F}_3 \right\}\end{aligned}$$

where $\boldsymbol{\alpha} = (\alpha(y_1), \alpha(y_2), ..., \alpha(y_{n_{c_W}}))^\top$ *is a discretization of* α *and* $\mathbf{u} = (u(c_1), u(c_2), ..., u(c_{n_{c_U}}))^\top$ *is a discretization of* u*.*

Remark 2.11 (Scaling laws for clipped multiple operator network classes). For $1 \leq i \leq 3$, let $\mathcal{F}_i(\varepsilon)$ denote the network classes in Theorem 2.8 such that (2) holds for ε. Corollary 2.9 implies that there exists a network in

$$\mathrm{Cl}_{\beta_V}(\beta_V, \mathcal{F}_1(\varepsilon/2), \mathcal{F}_2(\varepsilon/2), \mathcal{F}_3(\varepsilon/2), \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V})$$

such that (3) holds for ε. Indeed, the network in Corollary 2.9 is clipped at β_V and has coefficients $\theta_{pk\ell} = G[\alpha_p][u_k](v_\ell)$ which satisfy $|G[\alpha_p][u_k](v_\ell)| \leq \beta_V$ by assumption on G.

2.3 Covering Number of Neural Networks

In this section, we discuss covering numbers of the neural network classes defined in Definition 2.1.

Definition 2.12 (Covering Number)**.** *Let (X, d) be a metric space and let $\eta > 0$. A finite subset $\mathcal{C} \subset X$ is called a θ-cover of X if for every $x \in X$, there exists $c \in \mathcal{C}$ such that*

$$d(x, c) \leq \eta.$$

The covering number of X at scale θ with respect to the metric d is defined as

$$\mathcal{N}(\eta, X, d) := \min \{|\mathcal{C}| : \mathcal{C} \subset X \text{ is a } \eta\text{-cover of } X\}.$$

The next result is similar to [12, Lemma 7], [57, Lemma 3.2] or extensions in [56, Theorem 2.1].

Proposition 2.13 (Covering number of feedforward ReLU network class)**.** *Let $\mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R)$ be the network class defined in Definition 2.1 and suppose that $\kappa \geq 1$. Let $d(q_1, q_2)$ denote the maximum parameter discrepancy for $q_1, q_2 \in \mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R)$, that is*

$$d(q_1, q_2) = \max_{1 \leq \ell \leq L} \max\{\|W_\ell^{(1)} - W_\ell^{(2)}\|_{\infty,\infty}, \|b_\ell^{(1)} - b_\ell^{(2)}\|_\infty\}.$$

Then, the following identities hold:

1. *For any $q \in \mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R)$, the L^∞-norm of the output is bounded as follows:*

$$\|q\|_{\mathrm{L}^\infty} \leq \kappa^L (p+1)^{L-1} (p\|x\|_{\mathrm{L}^\infty} + 1); \tag{4}$$

2. *For any $q_1, q_2 \in \mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R)$, the L^∞-norm of the difference between the outputs is bounded as follows:*

$$\|q_1 - q_2\|_{\mathrm{L}^\infty} \leq L\kappa^{L-1} (p+1)^{L-1} (p\|x\|_{\mathrm{L}^\infty} + 1) d(q_1, q_2); \tag{5}$$

3. *The covering number of $\mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R)$ is bounded as follows:*

$$\mathcal{N}\left(\eta, \mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R), \|\cdot\|_{\mathrm{L}^\infty}\right) \leq \binom{L(p^2+p)}{K} \left(\left\lfloor \frac{L\kappa^L (p+1)^{L-1} (p\|x\|_{\mathrm{L}^\infty} + 1)}{\eta} \right\rfloor + 1 \right)^K.$$

3 Main Results

We start by introducing the standing assumptions on the underlying spaces utilized in the main results.

S. The space $U(d_U, \gamma_U, L_U, \beta_U)$ is a function set such that

(a) any function $u \in U$ is defined on $\Omega_U := [-\gamma_U, \gamma_U]^{d_U}$;

(b) for all functions $u \in U$ and $x, y \in \Omega_U$, we have

$$|u(x) - u(y)| \leq L_U |x - y|;$$

(c) for all functions $u \in U$, we have $\|u\|_{\mathrm{L}^\infty} \leq \beta_U$.

In the remainder of this section, we state our main results and discuss their interpretation. We conclude with a brief proof sketch highlighting the key ideas and the role of the intermediate technical lemmas.

3.1 Covering Numbers of Product Neural Network Classes

Our first result concerns the covering number of the clipped multiple operator network class from Definition 2.10. Since this class serves as the hypothesis class in Theorem 3.5, the estimation terms in the generalization bound involve $\log \mathcal{N}(\eta, \mathrm{Cl}_a, \|\cdot\|_{L^\infty(W\times U\times\Omega_V)})$ and therefore require an explicit covering-number estimate.

Proposition 3.1 (Covering number of the clipped multiple operator network class)**.** *Let*

$$\mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N)$$

be the clipped multiple operator class defined in Definition 2.10 where, for each NN *in that class, we assume the following input domains:* $\mathrm{NN} : W \times U \times \Omega_V \mapsto \mathbb{R}$*. Let*

$$F(L, p, K, \kappa, h) = \binom{L(p^2+p)}{K} \left(\left\lfloor \frac{2\kappa}{h} \right\rfloor + 1\right)^K.$$

Then, with $h = 2\eta/T$ *where*

$$\begin{aligned} T = P\cdot H\cdot N\cdot \Bigg[& IR_2R_3L_1\kappa_1^{L_1-1}(p_1+1)^{L_1-1}(p_1\|x\|_{\mathrm{L}^\infty}+1) \\ & + IR_1R_3L_2\kappa_2^{L_2-1}(p_2+1)^{L_2-1}(p_2\|\boldsymbol{u}\|_{\mathrm{L}^\infty}+1) \\ & + IR_1R_2L_3\kappa_3^{L_3-1}(p_3+1)^{L_3-1}(p_3\|\boldsymbol{\alpha}\|_{\mathrm{L}^\infty}+1) + R_1R_2R_3\Bigg], \end{aligned}$$

we obtain the following upper bound on the covering number:

$$\begin{aligned} & \mathcal{N}\left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right) \\ & \le \left[(\lfloor 2I/h\rfloor + 1)\, F(L_3, p_3, K_3, \kappa_3, h) F(L_2, p_2, K_2, \kappa_2, h) F(L_1, p_1, K_1, \kappa_1, h)\right]^{P\cdot H\cdot N}. \end{aligned}$$

If, in addition, for $d_W, d_U, d_V > 0$ *integers and* $\gamma_W, \gamma_U, \gamma_V, \beta_W, \beta_U, \beta_V, L_W, L_U, L_V > 0$*, the spaces* $W(d_W, \gamma_W, L_W, \beta_W)$*,* $U(d_U, \gamma_U, L_U, \beta_U)$ *and* $V(d_V, \gamma_V, L_V, \beta_V)$ *satisfy Assumption* ***S.****, then we can pick*

$$\begin{aligned} T = P\cdot H\cdot N\cdot \Bigg[& IR_2R_3L_1\kappa_1^{L_1-1}(p_1+1)^{L_1-1}(p_1\gamma_V+1) \\ & + IR_1R_3L_2\kappa_2^{L_2-1}(p_2+1)^{L_2-1}(p_2\beta_U+1) \\ & + IR_1R_2L_3\kappa_3^{L_3-1}(p_3+1)^{L_3-1}(p_3\beta_W+1) + R_1R_2R_3\Bigg]. \end{aligned}$$

In particular, the covering number is bounded uniformly in $\alpha \in W$*,* $u \in U$ *and* $x \in \Omega_V$*.*

We note that Proposition 3.1 yields a covering-number bound that depends explicitly on all architectural parameters of the component network classes $\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$ constituting the hypothesis class, including their depths L_i, widths p_i, sparsity budgets K_i, parameter magnitudes κ_i, and output bounds R_i, as well as the multiplicities P, H, N and coefficient magnitude I of the separable expansion.

3.2 Generalization Bounds

We now formalize the data-generating and sampling procedure used throughout the paper. In the multiple operator setting the training data are naturally collected in a hierarchical manner: we first sample operator instances α, then sample input functions u conditional on each α, and finally sample evaluation points x at which noisy observations of $G[\alpha][u]$ are recorded. The following definition makes this hierarchical training-set structure precise.

We recall that $X \sim \mathrm{subG}(\sigma^2)$, i.e. X is a mean 0 sub-Gaussian random variable with variance proxy σ^2 if $\mathbb{E}(e^{\lambda X}) \le e^{(\lambda^2\sigma^2)/2}$. In particular, if $X \sim \mathrm{subG}(\sigma^2)$, then $aX \sim \mathrm{subG}(a^2\sigma^2)$.

Definition 3.2 (Training set)**.** *Let $G : W \mapsto \{G[\alpha] : U \to V\}$ be a map. Let μ_α be a probability measure on W, μ_u a probability measure on U, and μ_x a probability measure on Ω_V. Given fixed sampling points $\{y_s\}_{s=1}^{n_{c_W}} \subset \Omega_W$ and $\{c_s\}_{s=1}^{n_{c_U}} \subset \Omega_U$, we define the training set:*

$$S_{G,\{y_s\},\{c_s\}} = \left\{ \boldsymbol{\alpha}_\ell, \left\{ \boldsymbol{u}_{\ell i}, \{(x_{\ell ij}, w_{\ell ij})\}_{j=1}^{n_x} \right\}_{i=1}^{n_u} \right\}_{\ell=1}^{n_\alpha}$$

where

- $\alpha_\ell \overset{\text{iid}}{\sim} \mu_\alpha$ *and* $\boldsymbol{\alpha}_\ell = \big(\alpha_\ell(y_1), \ldots, \alpha_\ell(y_{n_{c_W}})\big) \in \mathbb{R}^{n_{c_W}}$*;*
- $u_{\ell i} \overset{\text{iid}}{\sim} \mu_u$ *and* $\boldsymbol{u}_{\ell i} = \big(u_{\ell i}(c_1), \ldots, u_{\ell i}(c_{n_{c_U}})\big) \in \mathbb{R}^{n_{c_U}}$*;*
- $x_{\ell ij} \overset{\text{iid}}{\sim} \mu_x$ *drawn from* Ω_V*;*
- $w_{\ell ij} = G[\alpha_\ell][u_{\ell i}](x_{\ell ij}) + \zeta_{\ell ij}$*, where $\zeta_{\ell ij}$ are i.i.d. sub-Gaussian noise variables with mean 0 and variance proxy σ^2.*

All random variables are assumed independent across all indices. This structure is illustrated schematically in Figure 1.

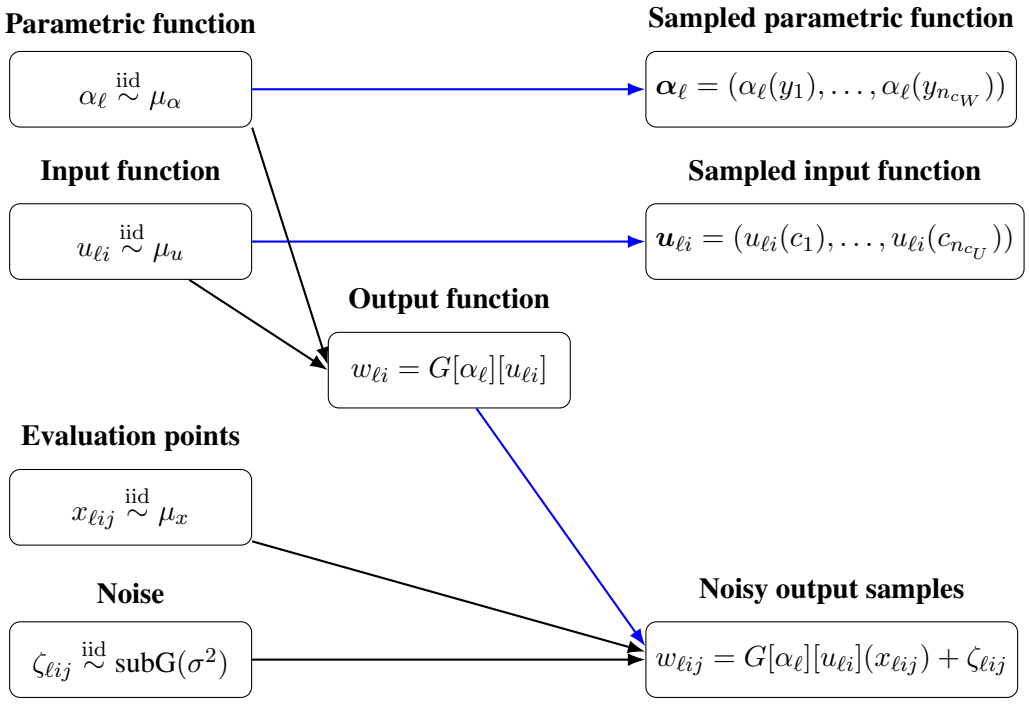

Figure 1: Schematic structure of the training dataset $S_{G,\{y_s\},\{c_s\}}$ defined in Definition 3.2 for multiple operator learning. supG(σ^2) denotes a sub-Gaussian distribution with variance proxy σ^2. Blue arrows indicate discretization steps; black arrows represent the flow of data.

Next, we define the trained operator given a dataset $S_{G,\{y_s\},\{c_s\}}$. In particular, the latter is a neural network chosen from the class $\mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\})$ and obtained by minimizing a L^2 empirical loss over the training set.

Definition 3.3 (Trained operator)**.** *Let $\mathcal{F}_i$ for $1 \leq i \leq 3$ be network classes defined in Definition 2.1. Let $G : W \mapsto \{G[\alpha] : U \to V\}$ be a map. Let μ_α be a probability measure on W, μ_u a probability measure on U, and μ_x a probability measure on Ω_V. Given fixed sampling points $\{y_s\}_{s=1}^{n_{c_W}} \subset \Omega_W$ and $\{c_s\}_{s=1}^{n_{c_U}} \subset \Omega_U$, let $S_{G,\{y_s\},\{c_s\}}$ be the training set defined in Definition 3.2. For $a, I > 0$, the trained a-clipped operator $G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}$ is defined as*

$$G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} = \underset{\mathrm{NN} \in \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\})}{\operatorname{argmin}} \frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \left(\mathrm{NN}[\boldsymbol{\alpha}_\ell][\boldsymbol{u}_{\ell i}](x_{\ell ij}) - w_{\ell ij}\right)^2$$

where $\mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\})$ is defined in Definition 2.10.

Subsequently, we introduce the expected generalization error of the learned operator. The following quantity measures the expected performance of $G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}$ by averaging over the randomness in the training set S and over unseen test inputs (α, u, x). It quantifies how well a model trained on one realization of the dataset generalizes to unseen data.

Definition 3.4 (Expected generalization error)**.** *Let $\mathcal{F}_i$ for $1 \leq i \leq 3$ be network classes defined in Definition 2.1. Let $G : W \mapsto \{G[\alpha] : U \to V\}$ be a map. Let μ_α be a probability measure on W, μ_u a probability measure on U, and μ_x a probability measure on Ω_V. Let $\{y_s\}_{s=1}^{n_{c_W}} \subset \Omega_W$ and $\{c_s\}_{s=1}^{n_{c_U}} \subset \Omega_U$ be fixed sampling points. We define the expected generalization error as*

$$\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \underbrace{\mathbb{E}_{\alpha\sim\mu_\alpha}\, \mathbb{E}_{u\sim\mu_u}\, \mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}}}_{\text{test sampling}} \underbrace{\left[\frac{1}{n_x}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}][\mathbf{u}](x_j) - G[\alpha][u](x_j)\right)^2\right]}_{\text{empirical approximation of the squared } \mathrm{L}^2(\mu_x) \text{ error}},$$

where $\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}$ denotes the expectation over the full training dataset $S_{G,\{y_s\},\{c_s\}}$ defined in Definition 3.2 (i.e. over all i.i.d. draws), $G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}$ is the a-clipped trained operator defined in Definition 3.3, $\boldsymbol{\alpha} = (\alpha(y_1), \alpha(y_2), ..., \alpha(y_{n_{c_W}}))^\top$ is a discretization of α and $\mathbf{u} = (u(c_1), u(c_2), ..., u(c_{n_{c_U}}))^\top$ is a discretization of u.

We now state our main scaling law bound for the expected generalization error. Fix a target approximation accuracy $\varepsilon > 0$, and instantiate the multiple operator network hypothesis class

$$\mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V})$$

using the ε-dependent architectural scalings prescribed by the approximation theory (so that the class contains an ε-accurate approximant to G). Let $\eta > 0$ denote the covering scale used to control the complexity of this class via $\log \mathcal{N}(\eta, \mathrm{Cl}_a, \|\cdot\|_{L^\infty(W\times U\times\Omega_V)})$. The following theorem combines these ingredients to yield an explicit approximation–estimation tradeoff linking target accuracy, architecture size, and the sampling budgets (n_α, n_u, n_x).

Theorem 3.5 (Scaling laws for the expected generalization error)**.** *Let $d_W, d_U, d_V > 0$ be integers,*

$$\gamma_W, \gamma_U, \gamma_V, \beta_W, \beta_U, \beta_V, L_W, L_U, L_V, L_G, L_{\mathcal{G}} > 0 \qquad \text{and} \qquad r_G, r_{\mathcal{G}} \geq 1$$

*and assume that $W(d_W, \gamma_W, L_W, \beta_W)$, $U(d_U, \gamma_U, L_U, \beta_U)$ and $V(d_V, \gamma_V, L_V, \beta_V)$ satisfy Assumption **S.**. Let G be a map such that*

$$\begin{aligned} &G : \{\alpha : \Omega_W \mapsto \mathbb{R} \,|\, \|\alpha\|_{\mathrm{L}^\infty} \leq \beta_W\} \mapsto \mathcal{G} \qquad \text{where} \\ &\mathcal{G} = \Big\{G[\alpha] \,|\, G[\alpha] : \{u : \Omega_U \mapsto \mathbb{R} \,|\, \|u\|_{\mathrm{L}^\infty} \leq \beta_U\} \mapsto V \text{ and} \\ &\qquad \|G[\alpha][u_1] - G[\alpha][u_2]\|_{\mathrm{L}^\infty(\Omega_V)} \leq L_{\mathcal{G}}\|u_1 - u_2\|_{\mathrm{L}^{r_{\mathcal{G}}}(\Omega_U)}\Big\} \end{aligned}$$

Furthermore, assume that G satisfies

$$\|G(\alpha_1) - G(\alpha_2)\|_{\mathrm{L}^\infty(\{u:\Omega_U\mapsto\mathbb{R}\,|\,\|u\|_{\mathrm{L}^\infty}\leq\beta_U\}\times\Omega_V)} \leq L_G\|\alpha_1 - \alpha_2\|_{\mathrm{L}^{r_G}(\Omega_W)}$$

for $\alpha_1, \alpha_2 \in \{\alpha : \Omega_W \mapsto \mathbb{R} \,|\, \|\alpha\|_{\mathrm{L}^\infty} \leq \beta_W\}$. There exists constants C depending on γ_V, L_V, C_δ depending on $L_{\mathcal{G}}, d_U, \gamma_U, r_{\mathcal{G}}, L_U$, C' depending on $\beta_U, L_{\mathcal{G}}, d_U, \gamma_U, r_{\mathcal{G}}$, C_ζ depending on $L_G, d_W, \gamma_W, r_G, L_W$ and C'' depending on $\beta_W, L_G, d_W, \gamma_W, r_G$ such that the following holds. For any $\varepsilon > 0$,

- *let $N = 2^{2n_{c_W}+3}C\sqrt{d_V}(C''\sqrt{n_{c_W}})^{n_{c_W}}\varepsilon^{-(n_{c_W}+1)}$ and consider the network class $\mathcal{F}_1 = \mathcal{F}_{\mathrm{NN}}(d_V, 1, L_1, p_1, K_1, \kappa_1, R_1)$ with parameters scaling as*

$$\begin{aligned} L_1 &= \mathcal{O}\left(d_V^2 \log d_V + d_V^2(n_{c_W}+1)\log(\varepsilon^{-1}) + d_V^2 \log(2^{n_{c_W}+1}(C''\sqrt{n_{c_W}})^{n_{c_W}})\right), \quad p_1 = \mathcal{O}(1), \\ K_1 &= \mathcal{O}\left(d_V^2 \log d_V + d_V^2(n_{c_W}+1)\log(\varepsilon^{-1}) + d_V^2 \log(2^{n_{c_W}+1}(C''\sqrt{n_{c_W}})^{n_{c_W}})\right) \\ \kappa_1 &= \mathcal{O}(2^{(d_V+1)(n_{c_W}+1)} d_V^{d_V/2+1}\varepsilon^{-(d_V+1)(n_{c_W}+1)}\left[2^{n_{c_W}+2}(C''\sqrt{n_{c_W}})^{n_{c_W}}\right]^{(d_V+1)}), \qquad R_1 = 1 \end{aligned}$$

  where the constants hidden in $\mathcal{O}$ depend on γ_V and L_V;

- *let* $\delta = \frac{C_\delta \varepsilon^{(1+d_V)(1+n_{c_W})}}{2^{2d_V+2n_{c_W}+3+d_V n_{c_W}} (C\sqrt{d_V})^{d_V} (C''\sqrt{n_{c_W}})^{n_{c_W}}}$ *and let* $\{c_m\}_{m=1}^{n_{c_U}} \subset \Omega_U$ *be points so that* $\{\mathcal{B}_\delta(c_m)\}_{m=1}^{n_{c_U}}$ *is a cover of* Ω_U *for some* n_{c_U}*;*
- *let* $H = 2^{3+2n_{c_W}+3d_V+2d_V n_{c_W}} C' \sqrt{n_{c_U}} (C\sqrt{d_V})^{d_V} (C''\sqrt{n_{c_W}})^{n_{c_W}(d_V+1)} \varepsilon^{-(d_V+1)(1+n_{c_W})}$ *and consider the network class* $\mathcal{F}_2 = \mathcal{F}_{\mathrm{NN}}(n_{c_U}, 1, L_2, p_2, K_2, \kappa_2, R_2)$ *with parameters scaling as*

$$
\begin{aligned}
L_2 &= \mathcal{O}\big(n_{c_U}^2 \log n_{c_U} + n_{c_U}^2 (d_V+1)(n_{c_W}+1)\log(\varepsilon^{-1}) + n_{c_U}^2 \log(2^{d_V+1}(C\sqrt{d_V})^{d_V}) \\
&\quad + n_{c_U}^2 (d_V+1)\log(2^{n_{c_W}+1}(C''\sqrt{n_{c_W}})^{n_{c_W}})\big), \quad p_2 = \mathcal{O}(1), \\
K_2 &= \mathcal{O}\big(n_{c_U}^2 \log n_{c_U} + n_{c_U}^2 (d_V+1)(n_{c_W}+1)\log(\varepsilon^{-1}) + n_{c_U}^2 \log(2^{d_V+1}(C\sqrt{d_V})^{d_V}) \\
&\quad + n_{c_U}^2 (d_V+1)\log(2^{n_{c_W}+1}(C''\sqrt{n_{c_W}})^{n_{c_W}})\big), \\
\kappa_2 &= \mathcal{O}(n_{c_U}^{n_{c_U}/2+1}[\varepsilon/2]^{-(d_V+1)(n_{c_U}+1)(n_{c_W}+1)}[2^{d_V+2}(C\sqrt{d_V})^{d_V}]^{n_{c_U}+1}\left[2^{d_V+1}(C\sqrt{d_V})^{d_V}\right]^{(d_V+1)(n_{c_U}+1)}), \\
R_2 &= 1
\end{aligned}
$$

  where the constants hidden in $\mathcal{O}$ *depend on* $\beta_U, L_{\mathcal{G}}, d_U, \gamma_U, r_{\mathcal{G}}$*;*
- *let* $\zeta = C_\zeta \varepsilon/2$ *and let* $\{y_m\}_{m=1}^{n_{c_W}} \subset \Omega_W$ *be points so that* $\{\mathcal{B}_\zeta(y_m)\}_{m=1}^{n_{c_W}}$ *is a cover of* Ω_W *for some* n_{c_W}*;*
- *let* $P = 4C''\sqrt{n_{c_W}}\varepsilon^{-1}$ *and consider the network class* $\mathcal{F}_3 = \mathcal{F}_{\mathrm{NN}}(n_{c_W}, 1, L_3, p_3, K_3, \kappa_3, R_3)$ *with parameters scaling as*

$$
\begin{aligned}
&L_3 = \mathcal{O}\left(n_{c_W}^2 \log(n_{c_W}) + n_{c_W}^2 \log(\varepsilon^{-1})\right), \quad p_3 = \mathcal{O}(1), \quad K_3 = \mathcal{O}\left(n_{c_W}^2 \log n_{c_W} + n_{c_W}^2 \log(\varepsilon^{-1})\right), \\
&\kappa_3 = \mathcal{O}(n_{c_W}^{n_{c_W}/2+1} 2^{n_{c_W}+1} \varepsilon^{-n_{c_W}-1} 2^{n_{c_W}+1}), \qquad R_3 = 1
\end{aligned}
$$

  where the constants hidden in $\mathcal{O}$ *depend on* $\beta_W, L_G, d_W, \gamma_W, r_G$*.*

Let $a = \beta_V$*,* $I \geq \beta_V$*,* $n_\alpha, n_u, n_x \in \mathbb{N}$*,* μ_α *a probability measure on* W*,* μ_u *a probability measure on* U*, and* μ_x *a probability measure on* Ω_V*. Consider the clipped network class*

$$
\mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}).
$$

For $\eta > 0$*, the expected generalization error is bounded as follows:*

$$
\begin{aligned}
&\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \mathbb{E}_{\alpha\sim\mu_\alpha} \mathbb{E}_{u\sim\mu_u} \mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}} \left[\frac{1}{n_x}\sum_{j=1}^{n_x} (G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}][\mathbf{u}](x_j) - G[\alpha][u](x_j))^2\right] \\
&\leq 4\varepsilon^2 + \eta(8\sigma+6) \\
&+ \frac{8\sigma\eta}{\sqrt{n_\alpha n_u n_x}} \sqrt{\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) + \log(2)} \\
&+ \frac{16\sigma^2}{n_\alpha n_u n_x}\left(\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) + \log(2)\right) \\
&+ \frac{112\beta_V^2}{3n_\alpha}\log\left(\mathcal{N}\left(\eta/(4\beta_V), \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right)
\end{aligned}
$$

where $\boldsymbol{\alpha} = (\alpha(y_1), \alpha(y_2), ..., \alpha(y_{n_{c_W}}))^\top$ *is a discretization of* α *and* $\mathbf{u} = (u(c_1), u(c_2), ..., u(c_{n_{c_U}}))^\top$ *is a discretization of* u*.*

In Theorem 3.5, the scaling laws depend on the best-in-class approximation error of the hypothesis class $\mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V})$, denoted by ε, where the architectural parameters of the component network classes $\mathcal{F}_i$ are instantiated as ε-dependent choices. The next corollary controls the resulting metric entropy of this ε-dependent hypothesis class explicitly as a function of ε.

Corollary 3.6 (Metric entropy bound under ε-dependent MNO scaling)**.** *Assume the same setting as in Theorem 3.5. Then,*

$$
\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right)
$$

$$\lesssim \varepsilon^{-\delta_1 \varepsilon^{-\delta_2 \varepsilon^{-d_W}}} \left(1 + \log(\eta^{-1})\right)$$

where $\delta_1 = d_U(1+d_V)\left(1+\frac{d_W}{2}\right) + d_W\frac{(d_V+1)}{2} + (d_V+1)$ *and* $\delta_2 = d_U(1+d_V)\left(1+\frac{d_W}{2}\right)$.

Remark 3.7 (Effective parameter scaling induced by the ε-dependent MNO construction). The ε-dependence of the metric entropy term in Corollary 3.6 is driven by the growth in (i) the number of subnetworks in the separable expansion and (ii) the admissible parameter magnitudes. In particular, it depends on the product-structure multiplicities through

$$\max\{N^{d_V},\, H^{n_{c_U}},\, P^{n_{c_W}}\} \qquad \text{and on the parameter bounds through} \qquad \max_{i\in\{1,2,3\}} \kappa_i.$$

As shown in [64, Remark 3.17], this growth is equivalent to the scaling of the total number of non-zero parameters $N_\#$ required by the resulting MNO-type architecture: specifically, we have

$$N_\# \;\lesssim\; \varepsilon^{-\gamma_1\, \varepsilon^{-\gamma_2\, \varepsilon^{-d_W}}},$$

for constants $\gamma_1, \gamma_2 > 0$ depending only d_W, d_U, d_V.

Theorem 3.5 yields a family of bounds parameterized by the approximation accuracy ε and covering scale η; using Corollary 3.6, in the next result, we select $\varepsilon = \varepsilon(n_\alpha)$ and $\eta = \eta(n_\alpha)$ to balance the approximation and estimation terms and thereby extract a single explicit learning rate as a function of the number of operator samples n_α.

Corollary 3.8 (Generalization rate in the number of sampled operators n_α)**.** *Assume the same setting as in Theorem 3.5. If we pick*

$$\varepsilon = \left(\frac{d_W}{2\delta_2}\frac{\log\log(n_\alpha)}{\log\log\log(n_\alpha)}\right)^{-\frac{1}{d_W}} \qquad \textit{and} \qquad \eta = 4\beta_V n_\alpha^{-1}$$

with $\delta_2 = d_U(1+d_V)\left(1+\frac{d_W}{2}\right)$*, then the expected generalization error scales as follows:*

$$\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\mathbb{E}_{\alpha\sim\mu_\alpha}\,\mathbb{E}_{u\sim\mu_u}\,\mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}}\left[\frac{1}{n_x}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}][\mathbf{u}](x_j) - G[\alpha][u](x_j)\right)^2\right]$$
$$= \mathcal{O}\left(\left(\frac{\log\log(n_\alpha)}{\log\log\log(n_\alpha)}\right)^{-\frac{2}{d_W}}\right)$$

where the constants hidden in $\mathcal{O}$ *are independent of* n_α, n_u, n_x.

3.3 Proof Sketch

We briefly summarize the main ideas behind the proofs of Theorem 3.5 and Corollaries 3.6, 3.8. The arguments follow the standard approximation–estimation paradigm, but the multiple operator setting (hierarchical sampling and data-dependent predictors) requires a careful organization of the stochastic terms.

Notation and risk decomposition. We recall that the learned operator network $\widehat{G}_S := G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}$ depends on the (random) training sample $S := S_{G,\{y_s\},\{c_s\}}$. For any candidate operator F (e.g. a network in our hypothesis class), define the population and empirical risks

$$\mathcal{R}(F) := \mathbb{E}_{\alpha\sim\mu_\alpha}\mathbb{E}_{u\sim\mu_u}\mathbb{E}_{x\sim\mu_x}\Big[\big(F[\alpha][u](x) - G[\alpha][u](x)\big)^2\Big],$$
$$\widehat{\mathcal{R}}_S(F) := \frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\Big(F[\alpha_\ell][u_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\Big)^2.$$

With this notation, the expected generalization error is

$$T_0 \;=\; \mathbb{E}_S\, \mathcal{R}(\widehat{G}_S).$$

Adding and subtracting the expected empirical risk of the learned predictor yields the decomposition

$$\begin{aligned} T_0 &= 2\,\mathbb{E}_S \widehat{\mathcal{R}}_S(\widehat{G}_S) \;+\; \Big(\mathbb{E}_S \mathcal{R}(\widehat{G}_S) - 2\,\mathbb{E}_S \widehat{\mathcal{R}}_S(\widehat{G}_S)\Big) \\ &=: T_1 + T_2, \end{aligned}$$

which matches (80) (the factor 2 is a bookkeeping constant).

Step 1: Control of T_1 (approximation + stochastic cross-term) The term $T_1 = 2\mathbb{E}_S \widehat{\mathcal{R}}_S(\widehat{G}_S)$ is an expected training-error quantity. Using the observation model $w = G[\alpha][u](x) + \zeta$ from Definition 3.2, we expand the square and split T_1 into (i) an approximation component and (ii) a noise-driven cross-term; see (81)–(89). The approximation component is controlled by the expressivity result for the clipped class: by Corollary 2.9 (together with Remark 2.11) there exists a network $\mathrm{NN} \in \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V})$ such that $\sup_{\alpha,u,x} |\,\mathrm{NN}[\alpha][u](x) - G[\alpha][u](x)| \le \varepsilon$, yielding a contribution of order ε^2 in (91). The remaining cross-term involves $(\widehat{G}_S - G)\zeta$ and cannot be averaged out directly because $\widehat{G}_S$ depends on S (and hence on ζ). We control this term by discretizing the relevant function class by an η-net and applying a moment generating function estimate together with a union bound, producing an estimation contribution involving $\log \mathcal{N}(\eta)$ and the sample sizes (n_α, n_u, n_x); see Lemma 4.1. This is the origin of the additive η term (net discretization) and the metric-entropy terms in the bound (92) for T_1.

Step 2: Control of T_2 (generalization gap via symmetrization and discretization) The term $T_2 = \mathbb{E}_S \mathcal{R}(\widehat{G}_S) - 2\mathbb{E}_S \widehat{\mathcal{R}}_S(\widehat{G}_S)$ is the generalization-gap contribution associated with the data-dependent predictor. To control it, we introduce an independent ghost sample S' (symmetrization), which rewrites population quantities as expectations of independent empirical averages and reduces the problem to bounding a supremum of a symmetrized (shifted) process over a shifted clipped class; see (48)–(53). We then discretize this shifted class by an η-net (constructed in Lemma 4.2), reducing the supremum to a maximum over finitely many functions plus an additive discretization error of order η; see (54)–(57). Finally, we bound the resulting maximum by a moment generating function estimate and a union bound over the finite cover, producing a term proportional to $\log \mathcal{N}(\eta)$ with explicit sample-size factors; see (69)–(79).

Step 3: Covering numbers and corollaries The proof of Theorem 3.5 is complete once the bounds for T_1 and T_2 from Steps 1–2 are combined, yielding the stated scaling-law estimate in terms of the metric entropy $\log \mathcal{N}(\eta)$. To obtain explicit consequences, we then invoke Proposition 3.1 to bound $\log \mathcal{N}(\eta)$ for the clipped hypothesis class, and subsequently choose the free parameters η and ε as functions of the sampling budgets to optimize/simplify the bound. This yields Corollary 3.6 (metric entropy as a function of ε) and Corollary 3.8 (an explicit generalization rate after balancing ε and η in terms of n_α).

4 Proofs

In this section, we present detailed proofs of our results.

4.1 Proofs of the Background Results

4.1.1 Scaling laws for clipped networks

Proof of Corollary 2.9. We note that

$$\begin{aligned} E &:= \sup_{\alpha\in W}\sup_{u\in U}\sup_{x\in\Omega_V} \big|G[\alpha][u](x) - \mathrm{Clip}_{\beta_V}\left(\mathrm{NN}[\alpha][u](x)\right)\big| \\ &\le \sup_{\alpha\in W}\sup_{u\in U}\sup_{x\in\Omega_V} |G[\alpha][u](x) - \mathrm{NN}[\alpha][u](x)| \end{aligned}$$

$$
\begin{aligned}
&+ \sup_{\alpha\in W}\sup_{u\in U}\sup_{x\in\Omega_V}\left|\mathrm{NN}[\alpha][u](x) - \mathrm{Clip}_{\beta_V}\left(\mathrm{NN}[\alpha][u](x)\right)\right| \\
&\leq \varepsilon + \sup_{\alpha\in W}\sup_{u\in U}\sup_{x\in\Omega_V}\left|\mathrm{NN}[\alpha][u](x) - \mathrm{Clip}_{\beta_V}\left(\mathrm{NN}[\alpha][u](x)\right)\right| \qquad (6)\\
&=: \frac{\varepsilon}{2} + T_1
\end{aligned}
$$

where we used (2) for (6).

For a fixed $\alpha \in W$, $u \in U$ and $x \in \Omega_V$, if $|\,\mathrm{NN}[\alpha][u](x)| \leq \beta_V$, then $T_1 = 0$ by definition and $E \leq \varepsilon/2$. Else, we first suppose that $\mathrm{NN}[\alpha][u](x) > \beta_V$. This implies that $\mathrm{Clip}_{\beta_V}\left(\mathrm{NN}[\alpha][u](x)\right) = \beta_V$ and therefore,

$$
\begin{aligned}
T_1 &= \mathrm{NN}[\alpha][u](x) - \beta_V \\
&\leq \mathrm{NN}[\alpha][u](x) - G[\alpha][u](x) \qquad (7)\\
&\leq |\,\mathrm{NN}[\alpha][u](x) - G[\alpha][u](x)| \\
&\leq \frac{\varepsilon}{2} \qquad (8)
\end{aligned}
$$

where we used the fact that $G[\alpha][u](x) \leq \beta_V$ by Assumption **S.** for (7) and (2) for (8). Similarly, if $\mathrm{NN}[\alpha][u](x) < -\beta_V$, then $\mathrm{Clip}_{\beta_V}\left(\mathrm{NN}[\alpha][u](x)\right) = -\beta_V$ and

$$
\begin{aligned}
T_1 &= -\beta_V - \mathrm{NN}[\alpha][u](x) \\
&\leq G[\alpha][u](x) - \mathrm{NN}[\alpha][u](x) \qquad (9)\\
&\leq |\,\mathrm{NN}[\alpha][u](x) - G[\alpha][u](x)| \\
&\leq \frac{\varepsilon}{2} \qquad (10)
\end{aligned}
$$

where we used the fact that $G[\alpha][u](x) \geq -\beta_V$ by Assumption **S.** for (9) and (2) for (10). Combining (8), (10) and (6), we conclude that $E \leq \varepsilon$.

□

4.1.2 Covering numbers for neural network classes

The following argument follows a standard discretization approach: we first establish uniform L^∞ bounds on network outputs and on output differences in terms of the architectural parameters, and then quantize the admissible parameter set to construct an explicit η-net and count its cardinality.

Proof of Proposition 2.13. 1. We start by establishing a bound for the L^∞-norm of the output of

$$q \in \mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R),$$

i.e. we provide a value of R as a function of all the other parameters. Specifically, we proceed by induction on the network depth to show that

$$\|q\|_{\mathrm{L}^\infty} \leq \kappa^k (p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty} + 1)$$

when $L = k$.

We recall the following bound for $A \in \mathbb{R}^{m\times n}$ and $x \in \mathbb{R}^n$:

$$\|Ax\|_{\mathrm{L}^\infty} = \max_{1\leq i\leq m}\left|\sum_{j=1}^{n}[A]_{ij}[x]_j\right| \leq n\|x\|_{\mathrm{L}^\infty}\|A\|_{\infty,\infty}. \qquad (11)$$

Base case: $L = 1$ The network we consider is $q(x) = W_1\mathrm{ReLU}(x) + b_1$. Then, we have:

$$
\begin{aligned}
\|q\|_{\mathrm{L}^\infty} &\leq \|W_1\mathrm{ReLU}(x)\|_{\mathrm{L}^\infty} + \|b_1\|_\infty \\
&\leq p\|W_1\|_{\infty,\infty}\|\mathrm{ReLU}(x)\|_{\mathrm{L}^\infty} + \|b_1\|_\infty \qquad (12)\\
&\leq p\kappa\|x\|_{\mathrm{L}^\infty} + \kappa \qquad (13)
\end{aligned}
$$

where we used (11) and the fact that the width of network (and hence the dimensions of the matrix W_1) is bounded by p for (12) as well as the facts that $\max\{\|W_1\|_{\infty,\infty}, \|b_1\|_\infty\} \leq \kappa$ and $\mathrm{ReLU}(x) \leq x$ for (13).

Induction step: $L = k+1$ Suppose that $\|q\|_{\mathrm{L}^\infty} \leq \kappa^k (p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty}+1)$ for $L=k$. For $L=k+1$, we write $q(x) = W_{L+1}\mathrm{ReLU}(\tilde{q}(x)) + b_{L+1}$ where $\tilde{q}$ is a feedforward ReLU network of depth k. We estimate as follows:

$$
\begin{aligned}
\|q\|_{\mathrm{L}^\infty} &\leq p\|W_{L+1}\|_\infty \|\mathrm{ReLU}(\tilde{q}(x))\|_{\mathrm{L}^\infty} + \|b_{L+1}\|_\infty && (14)\\
&\leq p\kappa\|\tilde{q}(x)\|_{\mathrm{L}^\infty} + \kappa && (15)\\
&\leq p\kappa\kappa^k(p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty}+1) + \kappa && (16)\\
&\leq p\kappa^{k+1}(p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty}+1) + \kappa^{k+1}(p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty}+1) && (17)\\
&= \kappa^{k+1}(p+1)^k(p\|x\|_{\mathrm{L}^\infty}+1)
\end{aligned}
$$

where we used (11) and the fact that the width of the network is bounded by p for (14), the facts that

$$\max\{\|W_1\|_{\infty,\infty}, \|b_1\|_\infty\} \leq \kappa$$

and $\mathrm{ReLU}(x) \leq x$ for (15), the induction hypothesis for (16) and the assumption that $\kappa \geq 1$ for (17).

2. Next, we proceed by induction on the network depth to bound the L^∞-norm of the difference between the output of two networks $q_1, q_2 \in \mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R)$. Specifically, we prove that

$$\|q_1 - q_2\|_{\mathrm{L}^\infty} \leq k\kappa^{k-1}(p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty}+1)d(q_1,q_2)$$

when $L=k$.

Base case: $L=1$ The networks we consider are $q_1(x) = W_1^{(1)}\mathrm{ReLU}(x) + b_1^{(1)}$ and $q_2(x) = W_1^{(2)}\mathrm{ReLU}(x) + b_1^{(2)}$. We have

$$
\begin{aligned}
\|q_1 - q_2\|_{\mathrm{L}^\infty} &\leq \|(W_1^{(1)} - W_1^{(2)})\mathrm{ReLU}(x)\|_{\mathrm{L}^\infty} + \|b_1^{(1)} - b_1^{(2)}\|_\infty \\
&\leq p\|W_1^{(1)} - W_1^{(2)}\|_{\infty,\infty}\|x\|_{\mathrm{L}^\infty} + d(q_1,q_2) && (18)\\
&\leq d(q_1,q_2)(p\|x\|_{\mathrm{L}^\infty}) + 1)
\end{aligned}
$$

where we used (11), the fact that the width of the two network is bounded by p and that $\mathrm{ReLU}(x) \leq x$ for (18).

Induction step: $L = k+1$ Suppose that $\|q_1 - q_2\|_{\mathrm{L}^\infty} \leq k\kappa^{k-1}(p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty}+1)d(q_1,q_2)$ for $L=k$. For $L=k+1$, we write $q_i(x) = W_{L+1}^{(i)}\mathrm{ReLU}\tilde{q}_i(x) + b_{L+1}^{(i)}$ where $\tilde{q}_i$ is a feedforward ReLU network of depth k. We estimate as follows:

$$
\begin{aligned}
&\|q_1 - q_2\|_{\mathrm{L}^\infty} \\
&\leq \|W_{L+1}^{(1)}\mathrm{ReLU}(\tilde{q}_1(x)) - W_{L+1}^{(2)}\mathrm{ReLU}(\tilde{q}_1(x)) - W_{L+1}^{(2)}\mathrm{ReLU}(\tilde{q}_2(x)) + W_{L+1}^{(2)}\mathrm{ReLU}(\tilde{q}_1(x))\|_{\mathrm{L}^\infty} \\
&+ \|b_{L+1}^{(1)} - b_{L+1}^{(2)}\|_\infty \\
&&(19)\\
&\leq p\|W_{L+1}^{(1)} - W_{L+1}\|_{\infty,\infty}\|\tilde{q}_1(x)\|_{\mathrm{L}^\infty} + p\|W_{L+1}^{(2)}\|_{\infty,\infty}\|\mathrm{ReLU}(\tilde{q}_1(x)) - \mathrm{ReLU}(\tilde{q}_2(x))\|_{\mathrm{L}^\infty} + d(q_1,q_2) \\
&&(20)\\
&\leq pd(q_1,q_2)\kappa^k(p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty}+1) + p\kappa\|\tilde{q}_1(x) - \tilde{q}_2(x)\|_{\mathrm{L}^\infty} + d(q_1,q_2) \\
&&(21)\\
&\leq pd(q_1,q_2)\kappa^k(p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty}+1) + pk\kappa^k(p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty}+1)d(q_1,q_2) + d(q_1,q_2) \\
&\leq pd(q_1,q_2)\kappa^k(p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty}+1) + pk\kappa^k(p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty}+1)d(q_1,q_2) \\
&&(22)\\
&+ (k+1)d(q_1,q_2)\kappa^k(p+1)^{k-1}(p\|x\|_{\mathrm{L}^\infty}+1)
\end{aligned}
$$

(23)
$= (k+1)\kappa^k (p+1)^k (p\|x\|_{\mathrm{L}^\infty} + 1) d(q_1, q_2)$

where we used (11) and the fact that the width of the two network is bounded by p for (19), (4) and the fact that ReLU is 1-Lipschitz for (20), the induction hypothesis for (21), the fact that $\kappa \geq 1$ for (22) and the identity $(k+1)(p+1) = pk + k + p + 1$ for (23).

3. Finally, we derive an upper bound on the covering number of $\mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R)$. For any $q \in \mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R)$, the weight matrices and biases satisfy $\max_{1 \leq \ell \leq L} \{\|W_\ell\|_{\infty,\infty}, \|b_\ell\|_\infty\} \leq \kappa$, so each parameter lies in the interval $[-\kappa, \kappa]$.

   For a fixed $h > 0$, we discretize the latter interval into $2\kappa/h$ subintervals of length h, yielding $\lfloor 2\kappa/h \rfloor + 1$ grid points. Then, for any parameter value $c \in [-\kappa, \kappa]$, there exists a grid point c^* such that $|c - c^*| \leq h/2$.

   Now, given any $q \in \mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R)$, let q^* denote the network where each nonzero parameter of q is replaced by its nearest grid point. By construction, this implies that $d(q, q^*) \leq h/2$, and it follows from (5) that

$$\|q - q^*\|_{\mathrm{L}^\infty} \leq L\, \kappa^{L-1} (p+1)^{L-1} \, (p\|x\|_{\mathrm{L}^\infty} + 1) \, \frac{h}{2}.$$

   Thus, setting $h = \frac{2\eta}{L\kappa^{L-1}(p+1)^{L-1}(p\|x\|_{\mathrm{L}^\infty}+1)}$, we conclude that the set of networks of the form

$$W_L \cdot \mathrm{ReLU}\left(W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) + \cdots + b_{L-1}\right) + b_L,$$

   i.e. a feedforward ReLU network with L layers and width p, whose nonzero parameters are constrained to grid points forms a η-cover of $\mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R)$ in the L^∞-norm.

   It remains to estimate the number of such networks. Since each $q \in \mathcal{F}_{\mathrm{NN}}(d_1, 1, L, p, K, \kappa, R)$ has at most K nonzero parameters, it suffices to consider networks with parameters restricted to grid points and at most K nonzero parameters.

   The total number of parameters of a feedforward ReLU network with L layers and width p, is at most $L(p^2 + p)$, where p^2 corresponds to weights and p to biases per layer. Therefore, the number of possible sparsity patterns is bounded by

$$\binom{L(p^2+p)}{K},$$

   and for each such pattern, there are $\lfloor 2\kappa/h \rfloor + 1$ choices per nonzero coordinate. Hence, the total number of distinct networks is at most

$$\binom{L(p^2+p)}{K} \left(\left\lfloor \frac{2\kappa}{h} \right\rfloor + 1\right)^K = \binom{L(p^2+p)}{K} \left(\left\lfloor \frac{L\kappa^L (p+1)^{L-1}(p\|x\|_{\mathrm{L}^\infty}+1)}{\eta} \right\rfloor + 1\right)^K.$$

□

4.2 Proof of the Main Results

4.2.1 Covering numbers for product network classes

The proof follows the same discretization strategy as Proposition 2.13, now applied to the separable multiple operator architecture, which is a linear combination of products of three subnetworks. In particular, in addition to quantizing the parameters of the subnetworks l_p, b_k, and τ_ℓ, we must also quantize the coefficient array $\{\theta_{pk\ell}\} \subset [-I, I]$ that weights the separable expansion.

Proof of Proposition 3.1. Let

$$\mathrm{NN}_1[\alpha][u](x) := \mathrm{Clip}_a \left(\sum_{p=1}^{P} \sum_{k=1}^{H} \sum_{\ell=1}^{N} \theta_{pk\ell} l_p(\boldsymbol{\alpha}) b_k(\mathbf{u}) \tau_\ell(x) \right)$$

and

$$\mathrm{NN}_2[\alpha][u](x) := \mathrm{Clip}_a\left(\sum_{p=1}^{P}\sum_{k=1}^{H}\sum_{\ell=1}^{N}\tilde{\theta}_{pk\ell}\tilde{l}_p(\boldsymbol{\alpha})\tilde{b}_k(\mathbf{u})\tilde{\tau}_\ell(x)\right)$$

be two neural networks in

$$\mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N).$$

We proceed as in Proposition 2.13 and start by estimating as follows:

$$\begin{aligned}
&\|\,\mathrm{NN}_1[\alpha][u](x) - \mathrm{NN}_2[\alpha][u](x)\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\\
&\leq \sum_{p=1}^{P}\sum_{k=1}^{H}\sum_{\ell=1}^{N}\|\theta_{pk\ell}l_p(\boldsymbol{\alpha})b_k(\mathbf{u})\tau_\ell(x) - \tilde{\theta}_{pk\ell}\tilde{l}_p(\boldsymbol{\alpha})\tilde{b}_k(\mathbf{u})\tilde{\tau}_\ell(x)\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\\
&\leq \sum_{p=1}^{P}\sum_{k=1}^{H}\sum_{\ell=1}^{N}\Big[|\theta_{pk\ell}|\cdot\|l_p(\boldsymbol{\alpha})b_k(\mathbf{u})\tau_\ell(x) - \tilde{l}_p(\boldsymbol{\alpha})\tilde{b}_k(\mathbf{u})\tilde{\tau}_\ell(x)\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\\
&+ |\theta_{pk\ell} - \tilde{\theta}_{pk\ell}|\cdot\|\tilde{l}_p(\boldsymbol{\alpha})\tilde{b}_k(\mathbf{u})\tilde{\tau}_\ell(x)\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\Big]\\
&\leq \sum_{p=1}^{P}\sum_{k=1}^{H}\sum_{\ell=1}^{N}\Big[|\theta_{pk\ell}|\cdot\Big(\|l_p(\boldsymbol{\alpha})\|_{\mathrm{L}^\infty(W)}\cdot\|b_k(\boldsymbol{u})\tau_\ell(x) - \tilde{b}_k(\boldsymbol{u})\tilde{\tau}_\ell(x)\|_{\mathrm{L}^\infty(U\times\Omega_V)}\\
&+ \|l_p(\boldsymbol{\alpha}) - \tilde{l}_p(\boldsymbol{\alpha})\|_{\mathrm{L}^\infty(W)}\cdot\|\tilde{b}_k(\mathbf{u})\tilde{\tau}_\ell(x)\|_{\mathrm{L}^\infty(U\times\Omega_V)}\Big) + |\theta_{pk\ell} - \tilde{\theta}_{pk\ell}|\cdot\|\tilde{l}_p(\boldsymbol{\alpha})\tilde{b}_k(\mathbf{u})\tilde{\tau}_\ell(x)\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\Big]\\
&\leq \sum_{p=1}^{P}\sum_{k=1}^{H}\sum_{\ell=1}^{N}\Big[|\theta_{pk\ell}|\cdot\Big(\|l_p(\boldsymbol{\alpha})\|_{\mathrm{L}^\infty(W)}\cdot\Big[\|b_k(\boldsymbol{u})\|_{\mathrm{L}^\infty(U)}\cdot\|\tau_\ell(x) - \tilde{\tau}_\ell(x)\|_{\mathrm{L}^\infty(\Omega_V)}\\
&+ \|b_k(\boldsymbol{u}) - \tilde{b}_k(\boldsymbol{u})\|_{\mathrm{L}^\infty(U)}\cdot\|\tilde{\tau}_\ell(x)\|_{\mathrm{L}^\infty(\Omega_V)}\Big] + \|l_p(\boldsymbol{\alpha}) - \tilde{l}_p(\boldsymbol{\alpha})\|_{\mathrm{L}^\infty(W)}\|\tilde{b}_k(\mathbf{u})\tilde{\tau}_\ell(x)\|_{\mathrm{L}^\infty(U\times\Omega_V)}\Big)\\
&+ |\theta_{pk\ell} - \tilde{\theta}_{pk\ell}|\cdot\|\tilde{l}_p(\boldsymbol{\alpha})\tilde{b}_k(\mathbf{u})\tilde{\tau}_\ell(x)\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\Big]\\
&\leq \sum_{p=1}^{P}\sum_{k=1}^{H}\sum_{\ell=1}^{N}\Big[I\Big(R_3\Big[R_2\|\tau_\ell(x) - \tilde{\tau}_\ell(x)\|_{\mathrm{L}^\infty(\Omega_V)} + \|b_k(\boldsymbol{u}) - \tilde{b}_k(\boldsymbol{u})\|_{\mathrm{L}^\infty(U)}R_1\Big]
\end{aligned}$$

(25)

$$\begin{aligned}
&+ \|l_p(\boldsymbol{\alpha}) - \tilde{l}_p(\boldsymbol{\alpha})\|_{\mathrm{L}^\infty(W)}R_2R_1\Big) + |\theta_{pk\ell} - \tilde{\theta}_{pk\ell}|\cdot R_3R_2R_1\Big]\\
&\leq \sum_{p=1}^{P}\sum_{k=1}^{H}\sum_{\ell=1}^{N}\Big[I\Big(R_3\Big[R_2L_1\kappa_1^{L_1-1}(p_1+1)^{L_1-1}(p_1\|x\|_{\mathrm{L}^\infty}+1)d(\tau_\ell,\tilde{\tau}_\ell)\\
&+ R_1L_2\kappa_2^{L_2-1}(p_2+1)^{L_2-1}(p_2\|\boldsymbol{u}\|_{\mathrm{L}^\infty}+1)d(b_k,\tilde{b}_k)\Big]
\end{aligned}$$

(26)

$$\begin{aligned}
&+ R_1R_2L_3\kappa_3^{L_3-1}(p_3+1)^{L_3-1}(p_3\|\boldsymbol{\alpha}\|_{\mathrm{L}^\infty}+1)d(l_p,\tilde{l}_p)\Big) + |\theta_{pk\ell} - \tilde{\theta}_{pk\ell}|\cdot R_3R_2R_1\Big]\\
&= \sum_{p=1}^{P}\sum_{k=1}^{H}\sum_{\ell=1}^{N}\Big[IR_2R_3L_1\kappa_1^{L_1-1}(p_1+1)^{L_1-1}(p_1\|x\|_{\mathrm{L}^\infty}+1)d(\tau_\ell,\tilde{\tau}_\ell)\\
&+ IR_1R_3L_2\kappa_2^{L_2-1}(p_2+1)^{L_2-1}(p_2\|\boldsymbol{u}\|_{\mathrm{L}^\infty}+1)d(b_k,\tilde{b}_k)
\end{aligned}$$

(27)

$$+ IR_1R_2L_3\kappa_3^{L_3-1}(p_3+1)^{L_3-1}(p_3\|\boldsymbol{\alpha}\|_{\mathrm{L}^\infty}+1)d(l_p,\tilde{l}_p) + R_1R_2R_3|\theta_{pk\ell}-\tilde{\theta}_{pk\ell}|\Bigg]$$

where we used the fact that the clipping operator Clip is 1-Lipschitz for (24), Definitions 2.1 and 2.10 for (25) as well as Proposition 2.13 for (26).

By the definition of $\mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P,H,N)$, we have $\theta_{pk\ell},\tilde{\theta}_{pk\ell}\in[-I,I]$ and each parameter of l_p, b_k and τ_ℓ is contained in $[-\kappa_3,\kappa_3]$, $[-\kappa_2,\kappa_2]$ and $[-\kappa_1,\kappa_1]$, respectively. For a fixed $h>0$, we discretize the latter intervals into $2I/h$, $2\kappa_1/h$, $2\kappa_2/h$ and $2\kappa_3/h$ subintervals of length h, yielding $\lfloor 2I/h\rfloor+1$, $\lfloor 2\kappa_1/h\rfloor+1$, $\lfloor 2\kappa_2/h\rfloor+1$ and $\lfloor 2\kappa_3/h\rfloor+1$ grid points. Then, for any coefficient $\theta_{pk\ell}$ or parameter value c of l_p, b_k and τ_ℓ, there exists grid points θ^* or c^* such that $|\theta_{pk\ell}-\theta^*|\leq h/2$ or $|c-c^*|\leq h/2$.

Now, for any $\mathrm{NN}\in\mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P,H,N)$, let NN^* denote its grid-constrained approximation. Specifically,

$$\mathrm{NN}^*[\alpha][u](x)=\mathrm{Clip}_a\left(\sum_{p=1}^{P}\sum_{k=1}^{H}\sum_{\ell=1}^{N}\theta^*_{pk\ell}l_p^*(\boldsymbol{\alpha})b_k^*(\mathbf{u})\tau_\ell^*(x)\right).$$

where $\theta^*_{pk\ell}$ is the nearest grid point to $\theta_{pk\ell}$ and l_p^*, b_k^*, and τ_ℓ^* are the grid-constrained versions of l_p, b_k, and τ_ℓ, obtained by replacing each nonzero parameter with its nearest grid point. By construction, this implies that $|\theta_{pk\ell}-\theta^*_{pk\ell}|\leq h/2$, $d(l_p,l_p^*)\leq h/2$, $d(b_k,b_k^*)\leq h/2$ and $d(\tau_\ell,\tau_\ell^*)\leq h/2$. Inserting this into (27), this yields

$$\begin{aligned}
&\|\,\mathrm{NN}[\alpha][u](x)-\mathrm{NN}^*[\alpha][u](x)\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\\
&\leq\frac{P\cdot H\cdot N\cdot h}{2}\Bigg[IR_2R_3L_1\kappa_1^{L_1-1}(p_1+1)^{L_1-1}(p_1\|x\|_{\mathrm{L}^\infty}+1)\\
&+IR_1R_3L_2\kappa_2^{L_2-1}(p_2+1)^{L_2-1}(p_2\|\boldsymbol{u}\|_{\mathrm{L}^\infty}+1)\\
&+IR_1R_2L_3\kappa_3^{L_3-1}(p_3+1)^{L_3-1}(p_3\|\boldsymbol{\alpha}\|_{\mathrm{L}^\infty}+1)+R_1R_2R_3\Bigg]\\
&=:\frac{h}{2}T
\end{aligned}\tag{28}$$

By picking $h=\frac{2\eta}{T}$, we conclude that the set of the networks of the form

$$\mathrm{Clip}_a\left(\sum_{p=1}^{P}\sum_{k=1}^{H}\sum_{\ell=1}^{N}\theta_{pk\ell}l_p(\boldsymbol{\alpha})b_k(\mathbf{u})\tau_\ell(x)\right)$$

whose nonzero coefficients and parameters are constrained to grid points form a η-cover of

$$\mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P,H,N).$$

For each tuple (p,k,ℓ), since l_p, b_k and τ_ℓ have at most K_3, K_2 and K_1 nonzero parameters, it suffices to consider networks restricted to grid points and at most K_3, K_2 and K_1 nonzero parameters. As argued in Proposition 2.13, for each tuple (p,k,ℓ), there therefore are

- $F(L_3,p_3,K_3,\kappa_3,h)$ possible grid-constrained networks l_p^*,
- $F(L_2,p_2,K_2,\kappa_2,h)$ possible grid-constrained networks b_k^*
- and $F(L_1,p_1,K_1,\kappa_1,h)$ possible grid-constrained networks τ_ℓ^*.

Furthermore, there are $\lfloor 2I/h\rfloor+1$ choices for $\theta^*_{pk\ell}$ and thus, for each tuple (p,k,ℓ), this yield a total of

$$(\lfloor 2I/h\rfloor+1)\,F(L_3,p_3,K_3,\kappa_3,h)F(L_2,p_2,K_2,\kappa_2,h)F(L_1,p_1,K_1,h)$$

grid-constrained networks. Since for each of the $P \cdot H \cdot N$ tuples (p, k, ℓ), the associated grid-constrained networks can be selected independently, we conclude that

$$\begin{aligned}&\mathcal{N}\left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\\ &\leq \left[\left(\lfloor 2I/h\rfloor + 1\right) F(L_3, p_3, K_3, \kappa_3, h) F(L_2, p_2, K_2, \kappa_2, h) F(L_1, p_1, K_1, \kappa_1, h)\right]^{P\cdot H\cdot N}.\end{aligned}$$

For the second claim of the Proposition, we continue from (28) and estimate it using Assumption **S.**:

$$\begin{aligned}&\|\,\mathrm{NN}[\alpha][u](x) - \mathrm{NN}^*[\alpha][u](x)\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\\ &\leq \frac{P\cdot H\cdot N\cdot h}{2}\Bigg[IR_2R_3L_1\kappa_1^{L_1-1}(p_1+1)^{L_1-1}(p_1\|x\|_{\mathrm{L}^\infty}+1)\\ &+ IR_1R_3L_2\kappa_2^{L_2-1}(p_2+1)^{L_2-1}(p_2\|\boldsymbol{u}\|_{\mathrm{L}^\infty}+1)\\ &+ IR_1R_2L_3\kappa_3^{L_3-1}(p_3+1)^{L_3-1}(p_3\|\boldsymbol{\alpha}\|_{\mathrm{L}^\infty}+1) + R_1R_2R_3\Bigg]\\ &\leq \frac{P\cdot H\cdot N\cdot h}{2}\Bigg[IR_2R_3L_1\kappa_1^{L_1-1}(p_1+1)^{L_1-1}(p_1\gamma_V+1)\\ &+ IR_1R_3L_2\kappa_2^{L_2-1}(p_2+1)^{L_2-1}(p_2\beta_U+1)\\ &+ IR_1R_2L_3\kappa_3^{L_3-1}(p_3+1)^{L_3-1}(p_3\beta_W+1) + R_1R_2R_3\Bigg].\end{aligned}$$

The rest of the proof is analogous to the first part.

□

4.2.2 Generalization bounds

We start this section with several intermediate results used in the proof of Theorem 3.5. Specifically, the next result bounds a quantity appearing in the estimation of the expected training-error.

Lemma 4.1 (Noise-error cross term bound)**.** *Assume the same setting as in Theorem 3.5. Then,*

$$\begin{aligned}&\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)\zeta_{\ell ij}\right]\\ &\leq \eta\sigma + \frac{\sigma}{\sqrt{n_\alpha n_u n_x}}\left(\eta + \sqrt{\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} - G)^2\right]}\right)\\ &\times\sqrt{\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) + \log(2)}\end{aligned}$$

where $\mathrm{Emp}(\mathrm{NN})^2 = \frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\mathrm{NN}[\boldsymbol{\alpha}_\ell][\boldsymbol{u}_{\ell i}][x_{\ell ij}]^2$.

Proof. Let

$$\mathrm{Cvr} = \{\mathcal{NN}_k^*\}_{k=1}^{\mathcal{N}\left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)}$$

be the η-covering constructed in the proof of Proposition 3.1. We have that

$$G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} \in \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{cW}}, H^{n_{cU}}, N^{d_V})$$

and there therefore exists $G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} \in \mathrm{Cvr}$ such that

$$\|G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} - G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \leq \eta. \tag{29}$$

Step 1: Decomposition We first decompose our quantity of interest as follows:

$$T_1 := \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)\zeta_{\ell ij}\right]$$

$$= \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij})\right)\zeta_{\ell ij}\right]$$

$$+ \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)\zeta_{\ell ij}\right]$$

(30)

$$=: T_2 + T_3.$$

We first upper bound T_2 as follows:

$$T_2^2 \leq \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\left(\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij})\right.\right.\right.$$

(31)
$$\left.\left.\left. - G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij})\right)\zeta_{\ell ij}\right)^2\right]$$

$$\leq \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\left(\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij})\right.\right.\right.$$

(32)
$$\left.\left.\left. - G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij})\right)^2\right)\left(\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\zeta_{\ell ij}^2\right)\right]$$

(33)
$$\leq \eta^2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\zeta_{\ell ij}^2\right]$$

(34)
$$\leq \eta^2\sigma^2$$

where we used Jensen's inequality for (31), Cauchy-Schwarz on the inner sum for (32), (29) for (33) and Definition 3.2 for (34).

Step 2: Moment generating function estimation We now turn to the estimation of T_3. In particular, we want to express the latter through moments of sub-Gaussian random variables. We first recall the squared empirical evaluation of a network

$$\mathrm{Emp}(\mathrm{NN})^2 = \frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\mathrm{NN}[\boldsymbol{\alpha}_\ell][\boldsymbol{u}_{\ell i}][x_{\ell ij}]^2.$$

With an abuse of notation, for our map G, we also write

$$\mathrm{Emp}(G)^2 = \frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}G[\alpha_\ell][u_{\ell i}][x_{\ell ij}]^2.$$

Then,

$$\mathrm{Emp}(G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} - G)$$
$$= \mathrm{Emp}(G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} - G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} + G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} - G)$$
$$\leq \left[\frac{2}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\boldsymbol{u}_{\ell i}][x_{\ell ij}] - G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\boldsymbol{u}_{\ell i}][x_{\ell ij}]\right)^2\right.$$

$$+\frac{2}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\boldsymbol{u}_{\ell i}][x_{\ell ij}]-G[\alpha_\ell][u_{\ell i}][x_{\ell ij}]\right)^2\Bigg]^{1/2}$$

$$\leq \left[2\eta^2+2\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}-G)^2\right]^{1/2} \tag{35}$$

$$\leq \sqrt{2}\left[\eta+\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}-G)\right) \tag{36}$$

where we used (29) for (35). Next, we define

$$z_k=\frac{1}{\sqrt{n_\alpha n_u n_x}\mathrm{Emp}(\mathrm{NN}_k^*-G)}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(\mathrm{NN}_k^*[\boldsymbol{\alpha}_\ell][\boldsymbol{u}_{\ell i}][x_{\ell ij}]-G[\alpha_\ell][u_{\ell i}][x_{\ell ij}]\right)\zeta_{\ell ij}$$

for $1\leq k\leq \mathcal{N}\left(\eta,\mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P,H,N),\|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)$ and note that,

$$z_k\mid\{\alpha_\ell,\{u_{\ell i},\{x_{\ell ij}\}\}\}\sim \mathrm{subG}\left(\sigma^2\frac{\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(\mathrm{NN}_k^*[\boldsymbol{\alpha}_\ell][\boldsymbol{u}_{\ell i}][x_{\ell ij}]-G[\alpha_\ell][u_{\ell i}][x_{\ell ij}]\right)^2}{\mathrm{Emp}(\mathrm{NN}_k^*-G)^2 n_\alpha n_u n_x}\right) \tag{37}$$

$$\sim \mathrm{subG}(\sigma^2)$$

where we used Definition 3.2 and [5, p.24] for (37). We estimate as follows:

$$T_3=\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\Bigg[\frac{\mathrm{Emp}(G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}-G)}{n_\alpha n_u n_x}$$

$$\times\frac{\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij})-G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)\zeta_{\ell ij}}{\mathrm{Emp}(G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}-G)}\Bigg]$$

$$\leq\sqrt{2}\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\Bigg[\frac{\eta+\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}-G)}{\sqrt{n_\alpha n_u n_x}}$$

$$\times\frac{\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij})-G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)\zeta_{\ell ij}}{\sqrt{n_\alpha n_u n_x}\mathrm{Emp}(G^*_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}-G)}\Bigg] \tag{38}$$

$$\leq\sqrt{2}\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{\eta+\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}-G)}{\sqrt{n_\alpha n_u n_x}}\max_k|z_k|\right]$$

$$\leq\frac{\sqrt{2}}{\sqrt{n_\alpha n_u n_x}}\sqrt{\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[(\eta+\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}-G))^2\right]}\sqrt{\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\max_k|z_k^2|\right]} \tag{39}$$

$$\leq\frac{2}{\sqrt{n_\alpha n_u n_x}}\sqrt{\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\eta^2+\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}-G)^2\right]}$$

$$\times\sqrt{\mathbb{E}_{\{\alpha_\ell,\{u_{\ell i},\{x_{\ell ij}\}\}\}}\left[\mathbb{E}_{\{\zeta_{\ell ij}\}}\left[\max_k|z_k^2|\mid\{\alpha_\ell,\{u_{\ell i},\{x_{\ell ij}\}\}\}\right]\right]} \tag{40}$$

$$\leq\frac{2}{\sqrt{n_\alpha n_u n_x}}\left(\eta+\sqrt{\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}-G)^2\right]}\right)$$

$$\times\sqrt{\mathbb{E}_{\{\alpha_\ell,\{u_{\ell i},\{x_{\ell ij}\}\}\}}\left[\mathbb{E}_{\{\zeta_{\ell ij}\}}\left[\max_k|z_k^2|\mid\{\alpha_\ell,\{u_{\ell i},\{x_{\ell ij}\}\}\}\right]\right]} \tag{41}$$

$$=:\frac{2}{\sqrt{n_\alpha n_u n_x}}\left(\eta+\sqrt{\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}-G)^2\right]}\right)\sqrt{\mathbb{E}_{\{\alpha_\ell,\{u_{\ell i},\{x_{\ell ij}\}\}\}}\left[T_4\right]} \tag{42}$$

where we used (36) for (38), Cauchy-Schwarz for (39), we split the expectation

$$\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}=\mathbb{E}_{\left\{\alpha_\ell,\left\{u_{\ell i},\left\{(x_{\ell ij},\zeta_{\ell ij})\right\}_j\right\}_i\right\}_\ell}=\mathbb{E}_{\left\{\alpha_\ell,\left\{u_{\ell i},\left\{x_{\ell ij}\right\}_j\right\}_i\right\}_\ell,\{\zeta_{\ell ij}\}_{\ell ij}}$$

using the law of iterated expectations $E_{X,Y}[f(X,Y)]=E_Y[E_X[f(X,Y)\mid Y]]$ for (40) and the inequality $\sqrt{a^2+b}\leq a+\sqrt{b}$ for (41).

Our aim is now show that $T_4 \leq C$ for $C > 0$ independent of all other random variables. This will follow from standard bounds on moment generating functions of the sub-Gaussian variables $z_k \mid \{\alpha_\ell, \{u_{\ell i}, \{x_{\ell ij}\}\}\}$. For $2/\sigma^2 \leq t \leq 4/\sigma^2$, we proceed as follows:

$$
\begin{aligned}
T_4 &= \frac{1}{t} \log \left(\exp \left(\mathbb{E}_{\{\zeta_{\ell ij}\}} \left[\max_k t|z_k^2| \mid \{\alpha_\ell, \{u_{\ell i}, \{x_{\ell ij}\}\}\} \right] \right) \right) \\
&\leq \frac{1}{t} \log \left(\mathbb{E}_{\{\zeta_{\ell ij}\}} \left[\max_k e^{tz_k^2} \mid \{\alpha_\ell, \{u_{\ell i}, \{x_{\ell ij}\}\}\} \right] \right) \qquad (43) \\
&\leq \frac{1}{t} \log \left(\mathbb{E}_{\{\zeta_{\ell ij}\}} \left[\sum_{k=1}^{\mathcal{N}\left(\eta, \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P,H,N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)} e^{tz_k^2} \mid \{\alpha_\ell, \{u_{\ell i}, \{x_{\ell ij}\}\}\} \right] \right) \\
&\leq \frac{1}{t} \log \left(\mathcal{N} \left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \right) \right) \\
&\quad + \frac{1}{t} \log \left(\mathbb{E}_{\{\zeta_{\ell ij}\}} \left[e^{tz_1^2} \mid \{\alpha_\ell, \{u_{\ell i}, \{x_{\ell ij}\}\}\} \right] \right) \qquad (44) \\
&\leq \frac{1}{t} \log \left(\mathcal{N} \left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \right) \right) + \frac{1}{t}\log(2) \qquad (45)
\end{aligned}
$$

where we used Jensen's inequality for (43), the fact that z_k is identically distributed for all k by (37) for (44) and the following equivalence from [5, p. 26] for (45): $X \sim \mathrm{subG}(v)$ if and only if $\mathbb{E}\left[e^{tX^2}\right] \leq 2$ for all $2/v \leq t \leq 4/v$. Picking $t = 4/\sigma^2$, we obtain that

$$
T_4 \leq \frac{\sigma^2}{4} \log \left(\mathcal{N} \left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \right) \right) + \frac{\sigma^2}{4}\log(2)
$$

which is non-random, since uniform in $\alpha \in W$, $u \in U$ and $x \in \Omega_V$ by the second claim of Proposition 3.1. Therefore, substituting the latter expression in (42) yields:

$$
\begin{aligned}
T_3 &\leq \frac{\sigma}{\sqrt{n_\alpha n_u n_x}} \left(\eta + \sqrt{\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} - G)^2 \right]} \right) \\
&\quad \times \sqrt{\log \left(\mathcal{N} \left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \right) \right) + \log(2)}.
\end{aligned}
$$

Continuing from (30) and using (34), we conclude that

$$
\begin{aligned}
T_1 &\leq \eta\sigma + \frac{\sigma}{\sqrt{n_\alpha n_u n_x}} \left(\eta + \sqrt{\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} - G)^2 \right]} \right) \\
&\quad \times \sqrt{\log \left(\mathcal{N} \left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \right) \right) + \log(2)}. \qquad \square
\end{aligned}
$$

We next establish the covering number of a shifted network class used for bounding the generalization-gap.

Lemma 4.2 (Covering number of the shifted clipped multiple operator class)**.** *Let*

$$
\mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N)
$$

be the clipped multiple operator class defined in Definition 2.10 where, for each NN *in that class, we assume the following input domains:* $\mathrm{NN} : W \times U \times \Omega_V \mapsto \mathbb{R}$*. Let* $G : W \mapsto \{G[\alpha] : U \mapsto V\}$ *be a map such that* $\|G\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \leq a$ *and define the shifted clipped multiple operator class*

$$
\begin{aligned}
&\mathrm{SCl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N) \\
&= \left\{ (\mathrm{NN}[\boldsymbol{\alpha}][\boldsymbol{u}](x) - G[\alpha][u](x))^2 \mid \mathrm{NN} \in \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N) \right\}.
\end{aligned}
$$

Then,

$$
\{(\mathcal{N}\mathcal{N}_k^* - G)^2\}_{k=1}^{\mathcal{N}\left(\eta/(4a), \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P,H,N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)}
$$

is a η-cover of $\mathrm{SCl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N)$ *where*

$$\{\mathcal{NN}_k^*\}_{k=1}^{\mathcal{N}\left(\eta/(4a), \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P,H,N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)}$$

is the $\eta/(4a)$-covering constructed in the proof of Proposition 3.1. In particular, we also have that

$$\begin{aligned}&\mathcal{N}\left(\eta, \mathrm{SCl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\\ &\leq \mathcal{N}\left(\eta/(4a), \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\end{aligned}$$

and

$$\|(\mathcal{NN}_k^* - G)^2\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \leq 4a^2$$

for all $1 \leq k \leq \mathcal{N}\left(\eta/(4a), \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)$.

Proof. Let $g = (\mathrm{NN} - G)^2$ and $\tilde{g} = (\widetilde{\mathrm{NN}} - G)^2$ be two networks in $\mathrm{SCl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N)$. Then, we have:

$$\begin{aligned}\|g - \tilde{g}\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} &= \left\|\left(\mathrm{NN} - \widetilde{\mathrm{NN}}\right)\left(\mathrm{NN} + \widetilde{\mathrm{NN}} - 2G\right)\right\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} &(46)\\ &\leq \left\|\mathrm{NN} - \widetilde{\mathrm{NN}}\right\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \left\|\mathrm{NN} + \widetilde{\mathrm{NN}} - 2G\right\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\\ &\leq 4a \left\|\mathrm{NN} - \widetilde{\mathrm{NN}}\right\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} &(47)\end{aligned}$$

where we used the identity $(b-c)^2 - (d-c)^2 = (b-d)(b+d-2c)$ for (46) as well as the facts that $\mathrm{NN}, \widetilde{\mathrm{NN}} \in \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N)$ and $\|G\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \leq a$ for (47).

Let $\{\mathcal{NN}_k^*\}_{k=1}^{\mathcal{N}\left(\eta/(4a), \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P,H,N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)}$ be the $\eta/(4a)$-covering constructed in the proof of Proposition 3.1. Equation (47) implies that

$$\{(\mathcal{NN}_k^* - G)^2\}_{k=1}^{\mathcal{N}\left(\eta/(4a), \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P,H,N), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)}$$

forms a η-covering of $\mathrm{SCl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P, H, N)$. In particular, by the construction in Proposition 3.1, we know that $\|\mathrm{NN}_k^*\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \leq a$ for all k and therefore,

$$\|(\mathcal{NN}_k^* - G)^2\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \leq 4a^2. \qquad \square$$

Lemma 4.3 (Bound on generalization gap/estimation part)**.** *Assume the same setting as in Theorem 3.5. Then,*

$$\begin{aligned}&\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \mathbb{E}_{\alpha\sim\mu_\alpha} \mathbb{E}_{u\sim\mu_u} \mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}} \left[\frac{1}{n_x}\sum_{j=1}^{n_x} \left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}][\mathbf{u}](x_j) - G[\alpha][u](x_j)\right)^2\right]\\ &- 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} \left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)^2\right]\\ &\leq 6\eta + \frac{112\beta_V^2}{3n_\alpha}\log\left(\mathcal{N}\left(\eta, \mathrm{SCl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right)\end{aligned}$$

where $\mathrm{SCl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V})$ *is defined in Lemma 4.2.*

Proof. For ease of notation, we define

$$\hat{g}[\alpha][u][x] = \left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}][\mathbf{u}](x) - G[\alpha][u](x)\right)^2.$$

Step 1: Symmetrization We estimate as follows:

$$
\begin{aligned}
T_1 &:= \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \mathbb{E}_{\alpha\sim\mu_\alpha} \mathbb{E}_{u\sim\mu_u} \mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}} \left[\frac{1}{n_x} \sum_{j=1}^{n_x} \left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}][\mathbf{u}](x_j) - G[\alpha][u](x_j) \right)^2 \right] \\
&- 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij}) \right)^2 \right] \\
&= \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \mathbb{E}_{\alpha\sim\mu_\alpha} \mathbb{E}_{u\sim\mu_u} \mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}} \left[\frac{1}{n_x} \sum_{j=1}^{n_x} \hat{g}[\alpha][u](x_j) \right] \\
&- 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \hat{g}[\alpha_\ell][u_{\ell i}](x_{\ell ij}) \right] \\
&= 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\mathbb{E}_{\alpha\sim\mu_\alpha} \mathbb{E}_{u\sim\mu_u} \mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}} \left[\frac{1}{n_x} \sum_{j=1}^{n_x} \hat{g}[\alpha][u](x_j) \right] \right. \\
&\left. - \frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \hat{g}[\alpha_\ell][u_{\ell i}](x_{\ell ij}) - \frac{1}{2} \mathbb{E}_{\alpha\sim\mu_\alpha} \mathbb{E}_{u\sim\mu_u} \mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}} \left[\frac{1}{n_x} \sum_{j=1}^{n_x} \hat{g}[\alpha][u](x_j) \right] \right] \\
&\leq 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\mathbb{E}_{\alpha\sim\mu_\alpha} \mathbb{E}_{u\sim\mu_u} \mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}} \left[\frac{1}{n_x} \sum_{j=1}^{n_x} \hat{g}[\alpha][u](x_j) \right] \right.
\end{aligned}
$$

(48)

$$
\left. - \frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \hat{g}[\alpha_\ell][u_{\ell i}](x_{\ell ij}) - \frac{1}{8\beta_V^2} \mathbb{E}_{\alpha\sim\mu_\alpha} \mathbb{E}_{u\sim\mu_u} \mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}} \left[\frac{1}{n_x} \sum_{j=1}^{n_x} \hat{g}^2[\alpha][u](x_j) \right] \right]
$$

where we used the fact that

$$
G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}][\mathbf{u}](x) \in \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{cW}}, H^{n_{cU}}, N^{d_V})
$$

and Assumption **S.** on V imply that

$$
\|\hat{g}[\alpha][u](x)\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \leq 4\beta_V^2, \tag{49}
$$

thus $\hat{g}^2/(4\beta_V^2) \leq \hat{g}\cdot\hat{g}/(4\beta_V^2) \leq \hat{g}$, for (48).

Let S' be an independent copy of S. We introduce the latter to symmetrize (48). In particular,

$$
\begin{aligned}
\mathbb{E}_{S'_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \hat{g}[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) \right] &= \frac{1}{n_\alpha n_u} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \mathbb{E}_{S'_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_x} \sum_{j=1}^{n_x} \hat{g}[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) \right] \\
&= \mathbb{E}_{\alpha\sim\mu_\alpha} \mathbb{E}_{u\sim\mu_u} \mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}} \left[\frac{1}{n_x} \sum_{j=1}^{n_x} \hat{g}[\alpha][u](x_j) \right] \qquad (50)
\end{aligned}
$$

where we used Definition 3.2 for (50). Inserting (50) in (48), we obtain

$$
\begin{aligned}
T_1 \leq 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} &\left[\mathbb{E}_{S'_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \hat{g}[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) \right] \right. \\
&\left. - \frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \hat{g}[\alpha_\ell][u_{\ell i}](x_{\ell ij}) - \frac{1}{8\beta_V^2} \mathbb{E}_{S'_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \hat{g}^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) \right] \right]
\end{aligned}
$$

$$\leq 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\sup_{g\in\mathrm{SCl}}\left(\mathbb{E}_{S'_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} g[\alpha'_\ell][u'_{\ell i}])(x'_{\ell ij})\right]\right.\right.$$

(51)

$$\left.\left.-\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} g[\alpha_\ell][u_{\ell i}](x_{\ell ij})-\frac{1}{8\beta_V^2}\mathbb{E}_{S'_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} g^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})\right]\right)\right]$$

$$=2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\sup_{g\in\mathrm{SCl}}\left(\mathbb{E}_{S'_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} g[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})-g[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right]\right.\right.$$

$$\left.\left.-\frac{1}{16\beta_V^2}\mathbb{E}_{S'_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} g^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})\right]\right)\right]$$

(52)

$$\left.\left.-\frac{1}{16\beta_V^2}\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} g^2[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right]\right)\right]$$

$$=2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\sup_{g\in\mathrm{SCl}}\left(\mathbb{E}_{S'_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} g[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})-g[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right]\right.\right.$$

$$\left.\left.-\frac{1}{16\beta_V^2}\mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} g^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})+g^2[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right]\right)\right]$$

$$=2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\sup_{g\in\mathrm{SCl}}\left(\mathbb{E}_{S'_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} g[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})-g[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right.\right.\right.$$

$$\left.\left.\left.-\frac{1}{16\beta_V^2}\mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} g^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})+g^2[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right]\right)\right)\right]$$

$$\leq 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}}\left[\sup_{g\in\mathrm{SCl}}\left(\frac{1}{n_\alpha}\sum_{\ell=1}^{n_\alpha}\left(\frac{1}{n_u n_x}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} g[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})-g[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right.\right.\right.$$

(53)

$$\left.\left.\left.-\frac{1}{16\beta_V^2}\mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_u n_x}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x} g^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})+g^2[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right]\right)\right)\right]$$

where we used the facts that $G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}][\mathbf{u}](x) \in \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V})$ and $\mathrm{SCl}=\mathrm{SCl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V})$ is the shifted clipped multiple operator network class defined in Lemma 4.2 in (51), the fact that S' be an independent copy of S for (52) and Jensen's inequality for (53).

Step 2: Moment generating function estimation Let

$$\mathrm{Cvr}=\{g_k^*\}_{k=1}^{\mathcal{N}\left(\eta,\mathrm{SCl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}),\|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)}$$

be the η-cover of SCl constructed in Lemma 4.2. Then, for every $g\in\mathrm{SCl}$, there exists $g^*\in\mathrm{Cvr}$ such that $\|g-g^*\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\leq\eta$. Using this, we estimate as follows:

$$\begin{aligned}g[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})-g[\alpha_\ell][u_{\ell i}](x_{\ell ij}) &= g[\alpha'_\ell][u'_{\ell i}][x'_{\ell ij}]-g^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})+g^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})-g^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})\\ &\quad+g^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})-g[\alpha_\ell][u_{\ell i}](x_{\ell ij})\\ &\leq 2\|g-g^*\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}+g^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})-g^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})\end{aligned}$$

$$\leq 2\eta + g^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g^*[\alpha_\ell][u_{\ell i}](x_{\ell ij}) \tag{54}$$

and

$$\begin{aligned}
&g^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) + g^2[\alpha_\ell][u_{\ell i}](x_{\ell ij}) \\
&= g^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - (g^*)^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) + g^2[\alpha_\ell][u_{\ell i}](x_{\ell ij}) - (g^*)^2[\alpha_\ell][u_{\ell i}](x_{\ell ij}) \\
&+ (g^*)^2[\alpha_\ell][u_{\ell i}](x_{\ell ij}) + (g^*)^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) \\
&\geq (g^*)^2[\alpha_\ell][u_{\ell i}](x_{\ell ij}) + (g^*)^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) \\
&- |g[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})| \cdot |g[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) + g^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})| \\
&- |g[\alpha_\ell][u_{\ell i}](x_{\ell ij}) - g^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})| \cdot |g[\alpha_\ell][u_{\ell i}](x_{\ell ij}) + g^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})| \qquad (55) \\
&\geq (g^*)^2[\alpha_\ell][u_{\ell i}](x_{\ell ij}) + (g^*)^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - 2\eta \left(\|g\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} + \|g^*\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \right) \\
&\geq (g^*)^2[\alpha_\ell][u_{\ell i}](x_{\ell ij}) + (g^*)^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - 16\eta\beta_V^2 \qquad (56)
\end{aligned}$$

where we used the inequality $(a-b)(a+b) + |a-b| \cdot |a+b| = a^2 - b^2 + |a-b| \cdot |a+b| \geq 0$ twice for (55) as well as (49) and Lemma 4.2 for (56). Inserting (54) and (56) into (53), we continue estimating as follows:

$$\begin{aligned}
T_1 &\leq 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}} \Bigg[3\eta + \sup_{g\in\mathrm{SCl}} \Bigg(\frac{1}{n_\alpha} \sum_{\ell=1}^{n_\alpha} \Bigg(\frac{1}{n_u n_x} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} g^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g^*[\alpha_\ell][u_{\ell i}](x_{\ell ij}) \\
&- \frac{1}{16\beta_V^2} \mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_u n_x} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} (g^*)^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) + (g^*)^2[\alpha_\ell][u_{\ell i}](x_{\ell ij}) \right] \Bigg) \Bigg) \Bigg] \\
&= 6\eta + 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}} \Bigg[\max_k \Bigg(\frac{1}{n_\alpha} \sum_{\ell=1}^{n_\alpha} \Bigg(\frac{1}{n_u n_x} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij}) \\
&- \frac{1}{16\beta_V^2} \mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_u n_x} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} (g_k^*)^2[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) + (g_k^*)^2[\alpha_\ell][u_{\ell i}](x_{\ell ij}) \right] \Bigg) \Bigg) \Bigg] \qquad (57) \\
&=: 6\eta + T_2.
\end{aligned}$$

For ease of notation, we write $\mathbb{E}_{S,S'}$ for $\mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}}$ and define

$$\begin{aligned}
r_k(\alpha'_\ell, \{u'_{\ell i}\}_{i=1}^{n_u}, \{x'_{\ell ij}\}_{i,j=1}^{n_u,n_x}, \alpha_\ell, \{u_{\ell i}\}_{i=1}^{n_u}, \{x_{\ell ij}\}_{i,j=1}^{n_u,n_x}) &= \frac{1}{n_u n_x} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij}) \right) \\
&=: r_{k\ell}
\end{aligned}$$

and note that, since S and S' are identical copies,

$$\mathbb{E}_{S,S'}\left[r_{k\ell}\right] = 0. \tag{58}$$

For fixed k and ℓ, we also define

$$Z_{ij}^{(k\ell)} := g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij}),$$

such that $r_{k\ell} = \frac{1}{n_u n_x} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} Z_{ij}^{(k\ell)}$. By (58), we have

$$\begin{aligned}
\mathrm{Var}(r_{k\ell}) &= \mathbb{E}_{S,S'}[r_{k\ell}^2] \\
&= \mathbb{E}_{S,S'} \left[\frac{1}{(n_u n_x)^2} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \sum_{s=1}^{n_u} \sum_{q=1}^{n_x} Z_{ij}^{(k\ell)} Z_{sq}^{(k\ell)} \right] \\
&= \mathbb{E}_{S,S'} \left[\frac{1}{(n_u n_x)^2} \sum_{(i,j)=(s,q)} \left(Z_{ij}^{(k\ell)} \right)^2 \right] + \mathbb{E}_{S,S'} \left[\frac{1}{(n_u n_x)^2} \sum_{(i,j)\neq(s,q)} Z_{ij}^{(k\ell)} Z_{sq}^{(k\ell)} \right]
\end{aligned}$$

$$= \mathbb{E}_{S,S'}\left[\frac{1}{(n_u n_x)^2}\sum_{(i,j)=(s,q)}\left(Z_{ij}^{(k\ell)}\right)^2\right] + \mathbb{E}_{S,S'}\left[\frac{1}{(n_u n_x)^2}\sum_{i=s,\, j\neq q} Z_{ij}^{(k\ell)} Z_{iq}^{(k\ell)}\right]$$

$$+ \mathbb{E}_{S,S'}\left[\frac{1}{(n_u n_x)^2}\sum_{i\neq s} Z_{ij}^{(k\ell)} Z_{sq}^{(k\ell)}\right]. \tag{59}$$

For $j \neq q$, we estimate as follows:

$$\mathbb{E}_{S,S'}\left[Z_{ij}^{(k\ell)} Z_{iq}^{(k\ell)}\right] = \mathbb{E}_{S,S'}\left[\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell iq}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell iq})\right)\right]$$

$$= \mathbb{E}_{\alpha_\ell,\alpha'_\ell \overset{\text{iid}}{\sim}\mu_\alpha,\, u_{\ell i},u'_{\ell i}\overset{\text{iid}}{\sim}\mu_u}\left[\mathbb{E}_{x_{\ell ij},x'_{\ell ij},x_{\ell iq},x'_{\ell iq}\overset{\text{iid}}{\sim}\mu_x}\left[\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)\right.\right.$$

$$\left.\left.\times\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell iq}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell iq})\right) \mid \alpha_\ell,\alpha'_\ell,u_{\ell i},u'_{\ell i}\right]\right]$$

$$= \mathbb{E}_{\alpha_\ell,\alpha'_\ell \overset{\text{iid}}{\sim}\mu_\alpha,\, u_{\ell i},u'_{\ell i}\overset{\text{iid}}{\sim}\mu_u}\left[\mathbb{E}_{x_{\ell ij},x'_{\ell ij}\overset{\text{iid}}{\sim}\mu_x}\left[\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right) \mid \alpha_\ell,\alpha'_\ell,u_{\ell i},u'_{\ell i}\right]\right.$$

$$\left.\times\,\mathbb{E}_{x_{\ell iq},x'_{\ell iq}\overset{\text{iid}}{\sim}\mu_x}\left[\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell iq}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell iq})\right) \mid \alpha_\ell,\alpha'_\ell,u_{\ell i},u'_{\ell i}\right]\right] \tag{60}$$

$$= \mathbb{E}_{\alpha_\ell,\alpha'_\ell \overset{\text{iid}}{\sim}\mu_\alpha,\, u_{\ell i},u'_{\ell i}\overset{\text{iid}}{\sim}\mu_u}\left[\mathbb{E}_{x_{\ell ij},x'_{\ell ij}\overset{\text{iid}}{\sim}\mu_x}\left[\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right) \mid \alpha_\ell,\alpha'_\ell,u_{\ell i},u'_{\ell i}\right]^2\right] \tag{61}$$

$$\leq \mathbb{E}_{\alpha_\ell,\alpha'_\ell \overset{\text{iid}}{\sim}\mu_\alpha,\, u_{\ell i},u'_{\ell i}\overset{\text{iid}}{\sim}\mu_u}\left[\mathbb{E}_{x_{\ell ij},x'_{\ell ij}\overset{\text{iid}}{\sim}\mu_x}\left[\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)^2 \mid \alpha_\ell,\alpha'_\ell,u_{\ell i},u'_{\ell i}\right]\right] \tag{62}$$

$$= \mathbb{E}_{S,S'}\left[\left(Z_{ij}^{(k\ell)}\right)^2\right] \tag{63}$$

where we used the conditional independence of $x_{\ell ij}, x'_{\ell ij}$ and $x_{\ell iq}, x'_{\ell iq}$ for (60), the fact that $x_{\ell ij}, x'_{\ell ij}$ and $x_{\ell iq}, x'_{\ell iq}$ are identically distributed for (61) and Jensen's inequality for (62). Similarly, for $i \neq s$, we have

$$\mathbb{E}_{S,S'}\left[Z_{ij}^{(k\ell)} Z_{sq}^{(k\ell)}\right] = \mathbb{E}_{\alpha_\ell,\alpha'_\ell\overset{\text{iid}}{\sim}\mu_\alpha}\left[\mathbb{E}_{u_{\ell i},u'_{\ell i},u_{\ell s},u'_{\ell s}\overset{\text{iid}}{\sim}\mu_u,x_{\ell ij},x'_{\ell ij},x_{\ell sq},x'_{\ell sq}\overset{\text{iid}}{\sim}\mu_x}\left[\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)\right.\right.$$

$$\left.\left.\times\left(g_k^*[\alpha'_\ell][u'_{\ell s}](x'_{\ell sq}) - g_k^*[\alpha_\ell][u_{\ell s}](x_{\ell sq})\right) \mid \alpha_\ell,\alpha'_\ell\right]\right]$$

$$= \mathbb{E}_{\alpha_\ell,\alpha'_\ell\overset{\text{iid}}{\sim}\mu_\alpha}\left[\mathbb{E}_{u_{\ell i},u'_{\ell i},u_{\ell s},u'_{\ell s}\overset{\text{iid}}{\sim}\mu_u,x_{\ell ij},x'_{\ell ij},x_{\ell sq},x'_{\ell sq}\overset{\text{iid}}{\sim}\mu_x}\left[\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right) \mid \alpha_\ell,\alpha'_\ell\right]\right.$$

$$\left.\times\,\mathbb{E}_{u_{\ell i},u'_{\ell i},u_{\ell s},u'_{\ell s}\overset{\text{iid}}{\sim}\mu_u,x_{\ell ij},x'_{\ell ij},x_{\ell sq},x'_{\ell sq}\overset{\text{iid}}{\sim}\mu_x}\left[\left(g_k^*[\alpha'_\ell][u'_{\ell s}](x'_{\ell sq}) - g_k^*[\alpha_\ell][u_{\ell s}](x_{\ell sq})\right) \mid \alpha_\ell,\alpha'_\ell\right]\right] \tag{64}$$

$$= \mathbb{E}_{\alpha_\ell,\alpha'_\ell\overset{\text{iid}}{\sim}\mu_\alpha}\left[\mathbb{E}_{u_{\ell i},u'_{\ell i},u_{\ell s},u'_{\ell s}\overset{\text{iid}}{\sim}\mu_u,x_{\ell ij},x'_{\ell ij},x_{\ell sq},x'_{\ell sq}\overset{\text{iid}}{\sim}\mu_x}\left[\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right) \mid \alpha_\ell,\alpha'_\ell\right]^2\right] \tag{65}$$

$$\leq \mathbb{E}_{\alpha_\ell,\alpha'_\ell\overset{\text{iid}}{\sim}\mu_\alpha}\left[\mathbb{E}_{u_{\ell i},u'_{\ell i},u_{\ell s},u'_{\ell s}\overset{\text{iid}}{\sim}\mu_u,x_{\ell ij},x'_{\ell ij},x_{\ell sq},x'_{\ell sq}\overset{\text{iid}}{\sim}\mu_x}\left[\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij}) - g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)^2 \mid \alpha_\ell,\alpha'_\ell\right]\right] \tag{66}$$

(67)
$$= \mathbb{E}_{S,S'}\left[\left(Z_{ij}^{(k\ell)}\right)^2\right]$$

where we used the conditional independence of $u_{\ell i}, u'_{\ell i}, x_{\ell ij}, x'_{\ell ij}$ and $u_{\ell s}, u'_{\ell s}, x_{\ell sq}, x'_{\ell sq}$ for (64), the fact that $u_{\ell i}, u'_{\ell i}, x_{\ell ij}, x'_{\ell ij}$ and $u_{\ell s}, u'_{\ell s}, x_{\ell sq}, x'_{\ell sq}$ are identically distributed for (65) and Jensen's inequality for (66). Inserting (63) and (67) into (59), we obtain:

$$\begin{aligned}
\mathrm{Var}(r_{k\ell}) &\leq \mathbb{E}_{S,S'}\left[\frac{1}{(n_u n_x)^2}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\sum_{s=1}^{n_u}\sum_{q=1}^{n_x}\left(Z_{ij}^{(k\ell)}\right)^2\right] \\
&= \mathbb{E}_{S,S'}\left[\frac{1}{n_u n_x}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(Z_{ij}^{(k\ell)}\right)^2\right] \\
&\leq 2\mathbb{E}_{S,S'}\left[\frac{1}{n_u n_x}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(g_k^*[\alpha'_\ell][u'_{\ell i}](x'_{\ell ij})^2 + g_k^*[\alpha_\ell][u_{\ell i}](x_{\ell ij})^2\right)\right]. \qquad (68)
\end{aligned}$$

Our aim is now to estimate T_2 through the moment generating function of $r_{k\ell}$. To this purpose, first re-write T_2 in (57):

$$T_2 \leq 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}}\left[\max_k\left(\frac{1}{n_\alpha}\sum_{\ell=1}^{n_\alpha}\left(r_{k\ell} - \frac{1}{32\beta_V^2}\mathrm{Var}(r_{k\ell})\right)\right)\right] \qquad (69)$$

where we used (68) for (69). Then, we proceed as follows for some $t > 0$:

$$\begin{aligned}
e^{tT_2/2} &\leq \exp\left(t\mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}}\left[\max_k\left(\frac{1}{n_\alpha}\sum_{\ell=1}^{n_\alpha}\left(r_{k\ell} - \frac{1}{32\beta_V^2}\mathrm{Var}(r_{k\ell})\right)\right)\right]\right) \\
&\leq \mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}}\left[\exp\left(t\max_k\left(\frac{1}{n_\alpha}\sum_{\ell=1}^{n_\alpha}\left(r_{k\ell} - \frac{1}{32\beta_V^2}\mathrm{Var}(r_{k\ell})\right)\right)\right)\right] \qquad (70) \\
&\leq \sum_k \mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}}\left[\exp\left(\frac{t}{n_\alpha}\sum_{\ell=1}^{n_\alpha}\left(r_{k\ell} - \frac{1}{32\beta_V^2}\mathrm{Var}(r_{k\ell})\right)\right)\right] \\
&= \sum_k \prod_{\ell=1}^{n_\alpha}\mathbb{E}_{S_{G,\{y_s\},\{c_s\}},S'_{G,\{y_s\},\{c_s\}}}\left[\exp\left(\frac{t}{n_\alpha}r_{k\ell}\right)\right] \times \exp\left(-\frac{t}{n_\alpha}\frac{1}{32\beta_V^2}\mathrm{Var}(r_{k\ell})\right) \qquad (71) \\
&=: \sum_k \prod_{\ell=1}^{n_\alpha} T_3 \exp\left(-\frac{t}{n_\alpha}\frac{1}{32\beta_V^2}\mathrm{Var}(r_{k\ell})\right)
\end{aligned}$$

where we used Jensen's inequality for (70), independence between S and S' as well as Definition 3.2 for (71). The T_3 term is the moment generating function we want to estimate for some $\lambda > 0$ (and fixed ℓ):

$$\begin{aligned}
\mathbb{E}_{S,S'}\left[\exp\left(\frac{t}{n_\alpha}r_{k\ell}\right)\right] &= \mathbb{E}_{S,S'}\left[1 + \lambda r_{k\ell} + \sum_{s=2}^{\infty}\frac{\lambda^s r_{k\ell}^s}{s!}\right] \\
&\leq \mathbb{E}_{S,S'}\left[1 + \lambda r_{k\ell} + \lambda^2 r_{k\ell}^2\sum_{s=2}^{\infty}\frac{\lambda^{s-2} r_{k\ell}^{s-2}}{2\cdot 3^{s-2}}\right] \qquad (72) \\
&\leq \mathbb{E}_{S,S'}\left[1 + \lambda r_{k\ell} + \lambda^2 r_{k\ell}^2\sum_{s=2}^{\infty}\frac{\lambda^{s-2}(8\beta_V^2)^{s-2}}{2\cdot 3^{s-2}}\right] \qquad (73) \\
&= \mathbb{E}_{S,S'}\left[1 + \lambda r_{k\ell} + \frac{\lambda^2 r_{k\ell}^2}{2}\sum_{s=2}^{\infty}\left(\frac{8\lambda\beta_V^2}{3}\right)^{s-2}\right] \qquad (74)
\end{aligned}$$

where used the fact that $s! \geq 2 \cdot 3^{s-2}$ for $s \geq 2$ for (72) (which is simply shown through induction) and the fact that

$$\|r_k(\alpha'_\ell, u'_{\ell i}, x'_{\ell ij}, \alpha_\ell, u_{\ell i}, x_{\ell ij})\|_{\mathrm{L}^\infty((W\times U\times\Omega_V)\times(W\times U\times\Omega_V))} \leq 2\|g_k^*\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)} \leq 8\beta_V^2$$

by Lemma 4.2 for (73). We pick $\lambda < 3/(8\beta_V^2)$ and continue from (74):

$$\begin{aligned}
\mathbb{E}_{S,S'}\left[\exp\left(\lambda r_{k\ell}\right)\right] &\leq \mathbb{E}_{S,S'}\left[1 + \lambda r_{k\ell} + \frac{\lambda^2 r_{k\ell}^2}{2}\frac{1}{1-\frac{8\lambda\beta_V^2}{3}}\right] \\
&= 1 + \frac{3\lambda^2}{6-16\lambda\beta_V^2}\mathrm{Var}(r_{k\ell}) \qquad (75)\\
&\leq \exp\left(\frac{3\lambda^2}{6-16\lambda\beta_V^2}\mathrm{Var}(r_{k\ell})\right) \qquad (76)
\end{aligned}$$

where we used (58) for (75) and the inequality $1 + x \leq e^x$ for $x \geq 0$ for (76). In order to insert (76) into (71), we need to ensure that

$$\frac{t}{n_\alpha} \leq \frac{3}{8\beta_V^2}$$

or, equivalently,

$$t \leq \frac{3n_\alpha}{8\beta_V^2}. \qquad (77)$$

For such t, using (76), we continue from (71) to obtain

$$e^{tT_2/2} \leq \sum_k \prod_{\ell=1}^{n_\alpha} \exp\left(\mathrm{Var}(r_{k\ell})\left[\frac{3\left(\frac{t}{n_\alpha}\right)^2}{6-\frac{16t\beta_V^2}{n_\alpha}} - \frac{t}{n_\alpha 32\beta_V^2}\right]\right). \qquad (78)$$

Now, we pick t such that

$$\frac{3\left(\frac{t}{n_\alpha}\right)^2}{6-\frac{16t\beta_V^2}{n_\alpha}} - \frac{t}{n_\alpha 32\beta_V^2} = 0,$$

or, equivalently,

$$t = \frac{3n_\alpha}{56\beta_V^2}.$$

Since $3/8 \geq 3/56$, we note that this choice of t is compatible with the requirement (77). From (78), we deduce that $e^{tT_2/2} \leq \sum_k 1 = \mathcal{N}\left(\eta, \mathrm{SCl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{cW}}, H^{n_{cU}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)$ where $\mathrm{SCl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{cW}}, H^{n_{cU}}, N^{d_V})$ is defined in Lemma 4.2. In turn, this yields

$$\begin{aligned}
T_2 &\leq \frac{2}{t}\log\left(\mathcal{N}\left(\eta, \mathrm{SCl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{cW}}, H^{n_{cU}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) \\
&= \frac{112\beta_V^2}{3n_\alpha}\log\left(\mathcal{N}\left(\eta, \mathrm{SCl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{cW}}, H^{n_{cU}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right). \qquad (79)
\end{aligned}$$

Substituting (79) in (57) yields the claim of the lemma.

□

Proof of Theorem 3.5. We start by decomposing the expected generalization error into a bias and a variance term. Specifically, we write:

$$T_0 := \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\mathbb{E}_{\alpha\sim\mu_\alpha}\,\mathbb{E}_{u\sim\mu_u}\,\mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}}\left[\frac{1}{n_x}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}][\mathbf{u}](x_j) - G[\alpha][u](x_j)\right)^2\right]$$

$$= 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)^2\right]$$

$$+ T_0 - 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)^2\right]$$

(80) $=: T_1 + T_2.$

Step 1: Bound on T_1 We start by bounding the T_1 term which corresponds to the expected empirical risk, that is the average squared error of the learned operator evaluated on the training inputs, measured against the noise-free outputs. For ease of notation, we will write $\mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}) = \mathrm{Cl}$. We estimate as follows:

(81)

$$T_1 = 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - w_{\ell ij} + \zeta_{\ell ij}\right)^2\right]$$

$$= 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - w_{\ell ij}\right)^2\right]$$

$$+ 4\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - w_{\ell ij}\right)\zeta_{\ell ij}\right]$$

$$+ 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\zeta_{\ell ij}^2\right]$$

(82)

$$=: 2\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\min_{\mathrm{NN}\in\mathrm{Cl}}\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(\mathrm{NN}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - w_{\ell ij}\right)^2\right]$$

$$+ 4\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - w_{\ell ij}\right)\zeta_{\ell ij}\right] + 2T_3$$

(83)

$$\leq 2\min_{\mathrm{NN}\in\mathrm{Cl}}\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(\mathrm{NN}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij}) - \zeta_{\ell ij}\right)^2\right]$$

$$+ 4\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - w_{\ell ij}\right)\zeta_{\ell ij}\right] + 2T_3$$

$$= 2\min_{\mathrm{NN}\in\mathrm{Cl}}\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(\mathrm{NN}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)^2\right]$$

$$- 4\min_{\mathrm{NN}\in\mathrm{Cl}}\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(\mathrm{NN}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)\zeta_{\ell ij}\right]$$

(84)

$$+ 4\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}}\left[\frac{1}{n_\alpha n_u n_x}\sum_{\ell=1}^{n_\alpha}\sum_{i=1}^{n_u}\sum_{j=1}^{n_x}\left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij}) - \zeta_{\ell ij}\right)\zeta_{\ell ij}\right] + 4T_3$$

$$= 2 \min_{\mathrm{NN}\in\mathrm{Cl}} \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \left(\mathrm{NN}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)^2 \right]$$

$$- 4 \min_{\mathrm{NN}\in\mathrm{Cl}} \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \left(\mathrm{NN}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right) \zeta_{\ell ij} \right]$$

(85)

$$+ 4\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right) \zeta_{\ell ij} \right]$$

where we used the definition of $w_{\ell ij}$ for (81), Definition 3.3 for (82), Jensen's inequality and the definition of $w_{\ell ij}$ for (83) as well as the definition of $w_{\ell ij}$ for (84). We continue by noting that $\mathrm{NN}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij})$ is independent of $\zeta_{\ell ij}$ by Definition 3.2 since, for a generic $\mathrm{NN} \in \mathrm{Cl}$, NN is not trained (this is in contrast with the trained operator $G_{a,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} \in \mathrm{Cl}$ which certainly depends on $w_{\ell ij}$ and thus $\zeta_{\ell ij}$). Analogously, $G[\alpha_\ell][u_{\ell i}](x_{\ell ij})$ is also independent of $\zeta_{\ell ij}$ by Definition 3.2. Therefore, we have:

$$\min_{\mathrm{NN}\in\mathrm{Cl}} \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \left(\mathrm{NN}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right) \zeta_{\ell ij} \right]$$

$$(86) \quad = \min_{\mathrm{NN}\in\mathrm{Cl}} \frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\left(\mathrm{NN}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)\right] \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\zeta_{\ell ij}\right]$$

$$(87) \quad = 0$$

where we used the above-noted independence in (86) and the fact that $\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} [\zeta_{\ell ij}] = 0$ by Definition 3.2 for (87). Inserting (87) in (85), we obtain:

$$T_1 \leq 2 \min_{\mathrm{NN}\in\mathrm{Cl}} \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \left(\mathrm{NN}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)^2 \right]$$

$$+ 4\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right) \zeta_{\ell ij} \right]$$

$$= 2 \min_{\mathrm{NN}\in\mathrm{Cl}} \frac{1}{n_\alpha n_u} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_x} \sum_{j=1}^{n_x} \left(\mathrm{NN}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right)^2 \right]$$

$$+ 4\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right) \zeta_{\ell ij} \right]$$

$$(88) \quad = 2 \min_{\mathrm{NN}\in\mathrm{Cl}} \frac{1}{n_\alpha n_u} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \mathbb{E}_{\alpha\sim\mu_\alpha} \mathbb{E}_{u\sim\mu_u} \mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}} \left[\frac{1}{n_x} \sum_{j=1}^{n_x} \left(\mathrm{NN}[\boldsymbol{\alpha}][\mathbf{u}](x_j) - G[\alpha][u](x_j)\right)^2 \right]$$

$$+ 4\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right) \zeta_{\ell ij} \right]$$

$$= 2 \min_{\mathrm{NN}\in\mathrm{Cl}} \mathbb{E}_{\alpha\sim\mu_\alpha} \mathbb{E}_{u\sim\mu_u} \mathbb{E}_{\{x_j\}_{j=1}^{n_x}\sim\mu_x^{\otimes n_x}} \left[\frac{1}{n_x} \sum_{j=1}^{n_x} \left(\mathrm{NN}[\boldsymbol{\alpha}][\mathbf{u}](x_j) - G[\alpha][u](x_j)\right)^2 \right]$$

$$+ 4\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\frac{1}{n_\alpha n_u n_x} \sum_{\ell=1}^{n_\alpha} \sum_{i=1}^{n_u} \sum_{j=1}^{n_x} \left(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S}[\boldsymbol{\alpha}_\ell][\mathbf{u}_{\ell i}](x_{\ell ij}) - G[\alpha_\ell][u_{\ell i}](x_{\ell ij})\right) \zeta_{\ell ij} \right]$$

$$(89) \quad =: T_4 + T_5$$

where we used the facts that $\alpha_\ell \overset{\text{iid}}{\sim} \mu_\alpha$, $u_{\ell i} \overset{\text{iid}}{\sim} \mu_u$ and $x_{\ell ij} \overset{\text{iid}}{\sim} \mu_x$ by Definition 3.2 for (88). For T_4, we estimate as follows:

$$T_4 \leq 2 \sup_{\alpha \in W} \sup_{u \in U} \sup_{x \in \Omega_V} |\,\mathrm{NN}[\boldsymbol{\alpha}][\boldsymbol{u}](x) - G[\alpha][u](x)|^2 \tag{90}$$

$$\leq 2\varepsilon^2 \tag{91}$$

where $\mathrm{NN} \in \mathrm{Cl}$ in (90) is the network such that (3) holds – the latter exists by Corollary 2.9 and Remark 2.11. Combining (91) with Lemma 4.1 (and using the notation from the latter) for T_5, we obtain that

$$\begin{aligned} T_1 &= \mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} - G)^2\right] \\ &\leq 2\varepsilon^2 + 4\eta\sigma + \frac{4\sigma}{\sqrt{n_\alpha n_u n_x}} \left(\eta + \sqrt{\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} - G)^2\right]}\right) \\ &\times \sqrt{\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) + \log(2)}. \end{aligned}$$

As in the proof of [44, Theorem 2], the latter can be equivalently written as $\rho^2 \leq c + 2b\,\rho$ with

$$\rho = \sqrt{\mathbb{E}_{S_{G,\{y_s\},\{c_s\}}} \left[\mathrm{Emp}(G_{a,I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,S} - G)^2\right]},$$

$$\begin{aligned} c &= 2\varepsilon^2 + 4\eta\sigma \\ &+ \frac{4\sigma\eta}{\sqrt{n_\alpha n_u n_x}} \sqrt{\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) + \log(2)} \end{aligned}$$

and

$$b = \frac{2\sigma}{\sqrt{n_\alpha n_u n_x}} \sqrt{\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) + \log(2)}.$$

Rearranging terms yields $(\rho - b)^2 \leq c + b^2$, which implies $\rho \leq \sqrt{c + b^2} + b$. Consequently, we obtain the bound $\rho^2 \leq b^2 + c + b^2 + 2b\sqrt{c + b^2} \leq 2b^2 + c + b^2 + c + b^2 = 2c + 4b^2$ (where the last inequality follows from $2pq \leq p^2 + q^2$) or, equivalently,

$$\begin{aligned} T_1 &\leq 4\varepsilon^2 + 8\eta\sigma \\ &+ \frac{8\sigma\eta}{\sqrt{n_\alpha n_u n_x}} \sqrt{\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) + \log(2)} \\ &+ \frac{16\sigma^2}{n_\alpha n_u n_x} \left(\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) + \log(2)\right). \end{aligned} \tag{92}$$

Step 2: Bound on T_2 By Lemma 4.3, we obtain that

$$T_2 \leq 6\eta + \frac{112\beta_V^2}{3n_\alpha} \log\left(\mathcal{N}\left(\eta, \mathrm{SCl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right),$$

so combining the latter with (80) and (92) yields:

$$\begin{aligned} T_0 &\leq 4\varepsilon^2 + 8\eta\sigma \\ &+ \frac{8\sigma\eta}{\sqrt{n_\alpha n_u n_x}} \sqrt{\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) + \log(2)} \\ &+ \frac{16\sigma^2}{n_\alpha n_u n_x} \left(\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) + \log(2)\right) \\ &+ 6\eta + \frac{112\beta_V^2}{3n_\alpha} \log\left(\mathcal{N}\left(\eta, \mathrm{SCl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) \\ &\leq 4\varepsilon^2 + \eta(8\sigma + 6) \end{aligned}$$

$$+\frac{8\sigma\eta}{\sqrt{n_\alpha n_u n_x}}\sqrt{\log\left(\mathcal{N}\left(\eta,\mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}),\|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right)+\log(2)}$$

$$+\frac{16\sigma^2}{n_\alpha n_u n_x}\left(\log\left(\mathcal{N}\left(\eta,\mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}),\|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right)+\log(2)\right) \tag{93}$$

$$+\frac{112\beta_V^2}{3n_\alpha}\log\left(\mathcal{N}\left(\eta/(4\beta_V),\mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}),\|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right)$$

where $\mathrm{SCl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V})$ is defined in Lemma 4.2 and we used the latter for (93).

□

Proof of Corollary 3.6. We start by estimating:

$$\begin{aligned}
F(L,p,K,\kappa,h) &= \binom{L(p^2+p)}{K}\left(\left\lfloor\frac{2\kappa}{h}\right\rfloor+1\right)^K \\
&\leq (L(p^2+p))^K\left(\frac{3\kappa}{h}\right)^K && (94)\\
&\lesssim \left(\frac{Lp^2\kappa T}{\eta}\right)^K && (95)
\end{aligned}$$

where we used the inequality $\binom{n}{k}\leq n^k$ for (94) and the definition of h in Proposition 3.1 for (95). We continue as follows:

$$\begin{aligned}
&\log\left(\mathcal{N}\left(\eta,\mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}),\|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right)\\
&\leq P^{n_{c_W}}H^{n_{c_U}}N^{d_V}\left[\log(3I/h)+K_3\log\left(L_3p_3^2\kappa_3T\right)+K_2\log\left(L_2p_2^2\kappa_2T\right)+K_1\log\left(L_1p_1^2\kappa_1T\right)\right] && (96)\\
&\lesssim P^{n_{c_W}}H^{n_{c_U}}N^{d_V}\left[\log\left(\frac{T}{\eta}\right)+K_3\log\left(\frac{L_3\kappa_3T}{\eta}\right)+K_2\log\left(\frac{L_2\kappa_2T}{\eta}\right)+K_1\log\left(\frac{L_1\kappa_1T}{\eta}\right)\right] && (97)
\end{aligned}$$

where we used Proposition 3.1 and (95) for (96) as well as the fact that $p_i=\mathcal{O}(1)$ for $1\leq i\leq 3$ by Theorem 3.5 for (97).

We now compute the asymptotic scaling of the all the above constants as a function of ε. In particular, we have

- $n_{c_W}\lesssim\varepsilon^{-d_W}$;
- $L_3\asymp K_3\lesssim(1+d_W)\varepsilon^{-2d_W}\log(\varepsilon^{-1})$;
- for κ_3, we first consider the logarithm:

$$\begin{aligned}
\log(\kappa_3) &\lesssim \left(\frac{n_{c_W}}{2}+1\right)\log(n_{c_W})+(1+n_{c_W})\log(\varepsilon^{-1})\\
&\lesssim \left(1+\frac{d_W}{2}\right)\varepsilon^{-d_W}\log(\varepsilon^{-1})
\end{aligned}$$

  which yields $\kappa_3\lesssim\varepsilon^{-\varepsilon^{-d_W}\left(1+\frac{d_W}{2}\right)}$;

- for $P^{n_{c_W}}$, we first consider the logarithm:

$$\begin{aligned}
\log(P^{n_{c_W}}) &\lesssim \frac{n_{c_W}}{2}\log(n_{c_W})+n_{c_W}\log(\varepsilon^{-1})\\
&\lesssim \left(1+\frac{d_W}{2}\right)\varepsilon^{-d_W}\log(\varepsilon^{-1})
\end{aligned}$$

  which yields $P^{n_{c_W}}\lesssim\varepsilon^{-\varepsilon^{-d_W}\left(1+\frac{d_W}{2}\right)}$;

- $n_{c_U} \lesssim \varepsilon^{-\varepsilon^{-d_W} d_U(1+d_V)\left(1+\frac{d_W}{2}\right)}$ (as in [64, Remark 3.17])
- $L_2 \asymp K_2 \lesssim \varepsilon^{-2\varepsilon^{-d_W} d_U(1+d_V)\left(1+\frac{d_W}{2}\right)} - d_W \log(\varepsilon^{-1})(1+d_U)(1+d_V)\left(1+\frac{d_W}{2}\right)$ (as in [64, Remark 3.17]);
- for κ_2, we first consider the logarithm:

$$\begin{aligned}\log(\kappa_2) &\lesssim \left(1+\frac{n_{c_U}}{2}\right)\log(n_{c_U}) + (d_V+1)(n_{c_U}+1)(n_{c_W}+1)\log(\varepsilon^{-1}) \\ &\lesssim \frac{n_{c_U}}{2}\log(n_{c_U}) + n_{c_U} n_{c_W}\log(\varepsilon^{-1}) \\ &\lesssim \varepsilon^{-\varepsilon^{-d_W} d_U(1+d_V)\left(1+\frac{d_W}{2}\right)}\left[\varepsilon^{-d_W} d_U(1+d_V)\left(1+\frac{d_W}{2}\right)\log(\varepsilon^{-1}) + \varepsilon^{-d_W}\log(\varepsilon^{-1})\right] \\ &= \varepsilon^{-\varepsilon^{-d_W} d_U(1+d_V)\left(1+\frac{d_W}{2}\right)}\varepsilon^{-d_W}\log(\varepsilon^{-1})\left[d_U(1+d_V)\left(1+\frac{d_W}{2}\right)+1\right]\end{aligned}$$

  which yields $\kappa_2 \lesssim \varepsilon^{-\varepsilon^{-d_W} d_U(1+d_V)\left(1+\frac{d_W}{2}\right)\varepsilon^{-d_W}\left[d_U(1+d_V)\left(1+\frac{d_W}{2}\right)+1\right]}$;
- $H^{n_{c_U}} \lesssim \varepsilon^{-\varepsilon^{-\varepsilon^{-d_W} d_U(1+d_V)\left(1+\frac{d_W}{2}\right)-d_W\left[d_U(1+d_V)\left(1+\frac{d_W}{2}\right)+d_W\frac{(d_V+1)}{2}+(d_V+1)\right]}}$ (as in [64, Remark 3.17]);
- $K_1 \asymp L_1 \lesssim \varepsilon^{-d_W}\log(\varepsilon^{-1}) d_V^2\left(1+\frac{d_W}{2}\right)$
- for κ_1, we first consider the logarithm:

$$\begin{aligned}\log(\kappa_1) &\lesssim (1+d_V)(1+n_{c_W})\log(\varepsilon^{-1}) + (1+d_V)\frac{n_{c_W}}{2}\log(n_{c_W}) \\ &\lesssim (1+d_V)\varepsilon^{-d_W}\log(\varepsilon^{-1}) + (1+d_V)\frac{d_W}{2}\varepsilon^{-d_W}\log(\varepsilon^{-1}) \\ &\lesssim (1+d_V)(1+\frac{d_W}{2})\varepsilon^{-d_W}\log(\varepsilon^{-1})\end{aligned}$$

  which yields $\kappa_1 \lesssim \varepsilon^{-\varepsilon^{-d_W}(1+d_V)(1+\frac{d_W}{2})}$;
- $N^{d_V} \lesssim \varepsilon^{-\varepsilon^{-d_W} d_V\left(\frac{d_W}{2}+1\right)}$ (as in [64, Remark 3.17]).

Using the above-derived formulas, we continue from (97) to derive that

$$\begin{aligned}&\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) \\ &\lesssim P^{n_{c_W}} H^{n_{c_U}} N^{d_V}\left[K_2\log\left(\frac{L_2\kappa_2 T}{\eta}\right)\right].\end{aligned}$$

Taking logarithms on the latter, and relying on the formulas again, we have

$$\begin{aligned}&\log\left(\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right)\right) \\ &\lesssim \log(H^{n_{c_U}}) + \log(K_2) + \log\left(\log\left(\frac{L_2\kappa_2 T}{\eta}\right)\right)\end{aligned}$$

from which we deduce that

$$\begin{aligned}&\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right) \\ &\lesssim H^{n_{c_U}}\log\left(\frac{L_2\kappa_2 T}{\eta}\right) \qquad (98) \\ &\lesssim H^{n_{c_U}}\log(L_2\kappa_2 T) + H^{n_{c_U}}\log(\eta^{-1}). \qquad (99)\end{aligned}$$

We now consider T and use the asymptotic formulas to estimate as follows:

$$T \lesssim P^{n_{c_W}} H^{n_{c_U}} N^{n_{c_U}} \left[L_1 \kappa_1^{L_1 - 1} + L_2 \kappa_2^{L_2 - 1} + L_3 \kappa_3^{L_3 - 1} \right] \tag{100}$$

$$\lesssim H^{n_{c_U}} L_2 \kappa_2^{L_2 - 1} \tag{101}$$

$$\lesssim H^{n_{c_U}} \kappa_2^{L_2} \tag{102}$$

where we use the fact that $R_i = 1$ and $p_i = \mathcal{O}(1)$ for $1 \leq i \leq 3$ by Theorem 3.5 for (100) and proceed in analagous fashion to how we obtained (98) for (101) and (102). Writing

$$H^{n_{c_U}} \lesssim \varepsilon^{-\varepsilon^{-\varepsilon^{-d_W} d_U (1+d_V) \left(1 + \frac{d_W}{2}\right) - d_W} \left[d_U (1+d_V) \left(1 + \frac{d_W}{2}\right) + d_W \frac{(d_V+1)}{2} + (d_V+1) \right]} \lesssim: \varepsilon^{-\delta_1 \varepsilon^{-\delta_2 \varepsilon^{-d_W}}},$$

$$\begin{aligned} L_2 &\lesssim \varepsilon^{-2\varepsilon^{-d_W} d_U (1+d_V) \left(1 + \frac{d_W}{2}\right)} - d_W \log(\varepsilon^{-1})(1+d_U)(1+d_V)\left(1 + \frac{d_W}{2}\right) \\ &\lesssim: \delta_3 \varepsilon^{-\delta_4 \varepsilon^{-d_W}} \log(\varepsilon^{-1}) \end{aligned}$$

and

$$\kappa_2 \lesssim \varepsilon^{-\varepsilon^{-\varepsilon^{-d_W} d_U (1+d_V) \left(1 + \frac{d_W}{2}\right)} \varepsilon^{-d_W} \left[d_U (1+d_V) \left(1 + \frac{d_W}{2}\right) + 1 \right]} \lesssim: \varepsilon^{-\delta_5 \varepsilon^{-\delta_6 \varepsilon^{-d_W}}},$$

from (102), we obtain that

$$\begin{aligned} T &\lesssim \varepsilon^{-\delta_1 \varepsilon^{-\delta_2 \varepsilon^{-d_W}}} \left(\varepsilon^{-\delta_5 \varepsilon^{-\delta_6 \varepsilon^{-d_W}}} \right)^{\delta_3 \varepsilon^{-\delta_4 \varepsilon^{-d_W}} \log(\varepsilon^{-1})} \\ &\lesssim \varepsilon^{-\delta_1 \varepsilon^{-\delta_2 \varepsilon^{-d_W}}} \varepsilon^{-\delta_5 \delta_3 \varepsilon^{-(\delta_6 + \delta_4) \varepsilon^{-d_W}} \log(\varepsilon^{-1})} \\ &\lesssim \varepsilon^{-\delta_1 \varepsilon^{-\delta_2 \varepsilon^{-d_W}} - \delta_5 \delta_3 \varepsilon^{-(\delta_6 + \delta_4) \varepsilon^{-d_W}} \log(\varepsilon^{-1})} \end{aligned}$$

and therefore

$$\begin{aligned} \log(L_2 \kappa_2 T) &\lesssim \log(\kappa_2 T) \\ &\lesssim \log\left(\varepsilon^{-\delta_1 \varepsilon^{-\delta_2 \varepsilon^{-d_W}} - \delta_5 \delta_3 \varepsilon^{-(\delta_6 + \delta_4) \varepsilon^{-d_W}} \log(\varepsilon^{-1}) - \delta_5 \varepsilon^{-\delta_6 \varepsilon^{-d_W}}} \right) \\ &\lesssim \left(\delta_1 \varepsilon^{-\delta_2 \varepsilon^{-d_W}} + \delta_5 \delta_3 \varepsilon^{-(\delta_6 + \delta_4) \varepsilon^{-d_W}} \log(\varepsilon^{-1}) + \delta_5 \varepsilon^{-\delta_6 \varepsilon^{-d_W}} \right) \log(\varepsilon^{-1}). \end{aligned} \tag{103}$$

Equation (103) shows that $\log(L_2 \kappa_2 T)$ grows much more slowly than $H^{n_{c_U}}$ so, continuing from (99), we have

$$\begin{aligned} &\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W \times U \times \Omega_V)} \right) \right) \\ &\lesssim \varepsilon^{-\delta_1 \varepsilon^{-\delta_2 \varepsilon^{-d_W}}} \left(1 + \log(\eta^{-1}) \right). \end{aligned}$$

□

Proof of Corollary 3.8. In the proof $C > 0$ will denote a constant that can be arbitrarily large, is independent of the η, n_α, n_u, n_x, and that may change from line to line.

From Theorem 3.5, we want to express

$$\begin{aligned} &4\varepsilon^2 + \eta(8\sigma + 6) \\ &+ \frac{8\sigma\eta}{\sqrt{n_\alpha n_u n_x}} \sqrt{\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W \times U \times \Omega_V)} \right) \right) + \log(2)} \\ &+ \frac{16\sigma^2}{n_\alpha n_u n_x} \left(\log\left(\mathcal{N}\left(\eta, \mathrm{Cl}_a(I, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \{y_s\}, \{c_s\}, P^{n_{c_W}}, H^{n_{c_U}}, N^{d_V}), \|\cdot\|_{\mathrm{L}^\infty(W \times U \times \Omega_V)} \right) \right) + \log(2) \right) \end{aligned}$$

$$+\frac{112\beta_V^2}{3n_\alpha}\log\left(\mathcal{N}\left(\eta/(4\beta_V),\mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}),\|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right)\right)$$
$$=: T$$

as a function of ε. For ease of notation, we write

$$\mathcal{N}(\eta) := \mathcal{N}\left(\eta,\mathrm{Cl}_a(I,\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3,\{y_s\},\{c_s\},P^{n_{c_W}},H^{n_{c_U}},N^{d_V}),\|\cdot\|_{\mathrm{L}^\infty(W\times U\times\Omega_V)}\right).$$

We estimate as follows:

$$T \leq 4\varepsilon^2+\eta(8\sigma+6)+\frac{8\sigma\eta}{\sqrt{n_\alpha n_u n_x}}\sqrt{\log\left(\mathcal{N}(\eta/(4\beta_V))\right)+\log(2)}+\frac{16\sigma^2}{n_\alpha n_u n_x}\left(\log\left(\mathcal{N}\left(\eta/(4\beta_V)\right)\right)+\log(2)\right)$$

(104)
$$+\frac{112\beta_V^2}{3n_\alpha}\log(\mathcal{N}(\eta/(4\beta_V)))$$
$$\lesssim 4\varepsilon^2+\eta(8\sigma+6)+\frac{8\sigma\eta}{\sqrt{n_\alpha n_u n_x}}\sqrt{\log\left(\mathcal{N}(\eta/(4\beta_V))\right)}+\frac{16\sigma^2}{n_\alpha n_u n_x}\log\left(\mathcal{N}\left(\eta/(4\beta_V)\right)\right)$$
$$+\frac{112\beta_V^2}{3n_\alpha}\log(\mathcal{N}(\eta/(4\beta_V)))$$
$$\lesssim 4\varepsilon^2+\eta(8\sigma+6)+\frac{8\sigma\eta}{\sqrt{n_\alpha n_u n_x}}\varepsilon^{-(\delta_1/2)\varepsilon^{-\delta_2\varepsilon^{-d_W}}}\left(1+\log(4\beta_V\eta^{-1})\right)$$

(105)
$$+\frac{16\sigma^2}{n_\alpha n_u n_x}\varepsilon^{-\delta_1\varepsilon^{-\delta_2\varepsilon^{-d_W}}}\left(1+\log(4\beta_V\eta^{-1})\right)+\frac{112\beta_V^2}{3n_\alpha}\varepsilon^{-\delta_1\varepsilon^{-\delta_2\varepsilon^{-d_W}}}\left(1+\log(4\beta_V\eta^{-1})\right)$$

where we used the fact that $\mathcal{N}(\eta)\leq\mathcal{N}(\tilde{\eta})$ if $\tilde{\eta}\leq\eta$ for (104) and Corollary 3.6 for (105). By picking $\eta=4\beta_V n_\alpha^{-1}$, we ensure that the η-dependent terms are of comparable scale to the n_α^{-1}-terms and, continuing from (105), we obtain:

$$T \lesssim 4\varepsilon^2+\frac{C}{n_\alpha}+\frac{C}{n_\alpha^{3/2}(n_u n_x)^{1/2}}\varepsilon^{-(\delta_1/2)\varepsilon^{-\delta_2\varepsilon^{-d_W}}}\log(n_\alpha)$$
$$+\frac{C}{n_\alpha n_u n_x}\varepsilon^{-\delta_1\varepsilon^{-\delta_2\varepsilon^{-d_W}}}\log(n_\alpha)+\frac{C}{n_\alpha}\varepsilon^{-\delta_1\varepsilon^{-\delta_2\varepsilon^{-d_W}}}\log(n_\alpha)$$
$$\lesssim 4\varepsilon^2+\frac{C}{n_\alpha}\varepsilon^{-\delta_1\varepsilon^{-\delta_2\varepsilon^{-d_W}}}\log(n_\alpha)$$
(106)
$$=: T_1+T_2.$$

We now balance the last two terms T_1 and T_2. Our goal is to choose $\varepsilon=\varepsilon(n_\alpha)$ so that the second term is of (at most) the same order as the first. Motivated by the structure of the exponent, we pick

(107)
$$\varepsilon := \left(\frac{d_W}{2\delta_2}\frac{\log\log n_\alpha}{\log\log\log n_\alpha}\right)^{-\frac{1}{d_W}}.$$

We now compute the corresponding asymptotic scales. By definition (107),

$$\log(\varepsilon^{-1}) = \frac{1}{d_W}\log\left(\frac{d_W}{2\delta_2}\frac{\log\log n_\alpha}{\log\log\log n_\alpha}\right)$$
$$= \frac{1}{d_W}\left(\log\frac{d_W}{2\delta_2}+\log\log\log n_\alpha-\log\log\log\log n_\alpha\right)$$
(108)
$$\lesssim \frac{1}{d_W}\log\log\log n_\alpha.$$

We next analyze the intermediate exponential

$$\varepsilon^{-\delta_2\varepsilon^{-d_W}} = \exp\left(\delta_2\varepsilon^{-d_W}\log\frac{1}{\varepsilon}\right)$$

$$\lesssim \exp\left(\delta_2 \frac{d_W}{2\delta_2} \frac{\log\log n_\alpha}{\log\log\log n_\alpha} \frac{\log\log\log n_\alpha}{d_W}\right) \tag{109}$$

$$= \exp\left(\frac{1}{2}\log\log n_\alpha\right)$$

$$= \log(n_\alpha)^{1/2} \tag{110}$$

where we used (108) for (109). We can now estimate the complete exponential factor as follows:

$$\begin{aligned} \varepsilon^{-\delta_1 \varepsilon^{-\delta_2 \varepsilon^{-d_W}}} &= \exp\Big(\delta_1 \varepsilon^{-\delta_2 \varepsilon^{-d_W}} \log \tfrac{1}{\varepsilon}\Big) \\ &\lesssim \exp\left(\frac{\delta_1}{d_W} \log(n_\alpha)^{1/2} \log\log\log n_\alpha\right) \end{aligned} \tag{111}$$

where we used (110) for (111) and thus

$$\log(T_2) \lesssim \log(C) + \log\left(\frac{1}{n_\alpha}\right) + \frac{\delta_1}{d_W} \log(n_\alpha)^{1/2} \log\log\log n_\alpha + \log\log n_\alpha.$$

On the other hand, with the choice (107),

$$\log(T_1) = \log(4) + \frac{-2}{d_W}\left[\log\left(\frac{d_W}{2\delta_2}\right) + \log\log\log n_\alpha - \log\log\log\log n_\alpha\right].$$

We note that $\lim_{n_\alpha \to \infty} \log(T_2) = -\infty$ due to the $\log(n_\alpha^{-1})$ term; we also have $\lim_{n_\alpha \to \infty} \log(T_1) = -\infty$ due to the term $-\log\log\log n_\alpha$. This implies that T_2 goes to 0 much faster than T_1, so T_1 dominates in (106) and, using (107), we conclude

$$T \lesssim 4\varepsilon^2 \lesssim \left(\frac{d_W}{2\delta_2} \frac{\log\log n_\alpha}{\log\log\log n_\alpha}\right)^{-\frac{2}{d_W}}.$$

□

Remark 4.4 (Exact error balancing). In principle, one could choose $\varepsilon = \varepsilon(n_\alpha)$ by solving the implicit balancing relation $T_1(\varepsilon) \asymp T_2(\varepsilon)$. However, since $T_2(\varepsilon)$ contains the nested exponential factor $\varepsilon^{-\delta_1 \varepsilon^{-\delta_2 \varepsilon^{-d_W}}}$, this equation does not admit a tractable closed-form solution. We therefore select an explicit $\varepsilon(n_\alpha)$ for which T_2 is asymptotically negligible (and hence T_1 dominant), yielding an explicit rate.

5 Conclusion

In this work, we provided theoretical insights into statistical generalization for multiple operator learning, where the goal is to learn an operator family $\{G[\alpha] : U \to V\}_{\alpha \in W}$ from data collected hierarchically across sampled operator instances, input functions, and evaluation points. To motivate and contextualize this setting, we presented several representative examples in which the multi-operator viewpoint is intrinsic, including parameterized integral operators, parameterized PDE solution operators, and task-conditioned operator families. For separable neural architectures, in particular the MNO model, we established an explicit generalization bound that makes the dependence on the sampling budgets (n_α, n_u, n_x) and on the statistical complexity of the induced hypothesis class transparent. The analysis combines two main ingredients: (i) an ε-accurate approximation guarantee for MNO (with an explicit ε-dependent architecture) from [64], and (ii) a covering-number estimate for the corresponding clipped separable network class, derived via a parameter-quantization argument. Balancing the approximation scale ε with the covering scale η then yields an explicit learning-rate statement in the operator-sampling budget n_α, highlighting how increased operator variability improves transfer to previously unseen operator instances.

Several extensions of the present theory are natural. First, our analysis treats the trained network as an exact empirical-risk minimizer over the prescribed hypothesis class. In practice, training is approximate and influenced by optimization dynamics, regularization, and early stopping; incorporating an explicit optimization/suboptimality term, or deriving algorithm-dependent generalization guarantees, would sharpen the connection between the theory and real training pipelines. Second, the rates obtained here rely on global L^∞

covering-number control for clipped hypothesis classes, which is a large class. Improved bounds may be possible by using localized or data-dependent complexity measures, and by studying alternative architectures or conditioning regimes that reduce the effective statistical complexity. In particular, it would be interesting to extend the analysis beyond separable MNO-type classes to attention-based operator architectures, in which the operator descriptor α and the input observations are encoded and combined in a shared latent representation. Finally, under our current global complexity control, the bound is asymptotically bottlenecked by the n_α-dependent term, meaning that operator diversity cannot be compensated for by oversampling (u, x) once n_α is fixed. This motivates investigating adaptive or active strategies for selecting operator instances α to accelerate transfer to previously unseen operators under a constrained operator-sampling budget.

Acknowledgments

A. Weihs and H. Schaeffer were supported in part by NSF 2427558.

References

[1] Ben Adcock, Michael Griebel, and Gregor Maier. The sample complexity of learning lipschitz operators with respect to gaussian measures, 2025.

[2] Anima Anandkumar, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Nikola Kovachki, Zongyi Li, Burigede Liu, and Andrew Stuart. Neural operator: Graph kernel network for partial differential equations. In *ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations*, 2019.

[3] Aras Bacho, Aleksei G. Sorokin, Xianjin Yang, Théo Bourdais, Edoardo Calvello, Matthieu Darcy, Alexander Hsu, Bamdad Hosseini, and Houman Owhadi. Operator learning at machine precision, 2025.

[4] Kaushik Bhattacharya, Bamdad Hosseini, Nikola B. Kovachki, and Andrew M. Stuart. Model Reduction And Neural Networks For Parametric PDEs. *The SMAI Journal of computational mathematics*, 7:121–157, 2021.

[5] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 02 2013.

[6] Nicolas Boullé, Seick Kim, Tianyi Shi, and Alex Townsend. Learning green's functions associated with time-dependent partial differential equations. *Journal of Machine Learning Research*, 23(218):1–34, 2022.

[7] Jean Bourgain, Haim Brezis, and Petru Mironescu. Another look at Sobolev spaces. In *Optimal Control and Partial Differential Equations*, 2001.

[8] Yadi Cao, Yuxuan Liu, Liu Yang, Rose Yu, Hayden Schaeffer, and Stanley Osher. Vicon: Vision in-context operator networks for multi-physics fluid dynamics prediction. *arXiv preprint arXiv:2411.16063*, 2024.

[9] Javier Castro. The kolmogorov infinite dimensional equation in a hilbert space via deep learning methods. *Journal of Mathematical Analysis and Applications*, 527(2):127413, 2023.

[10] Javier Castro, Claudio Muñoz, and Nicolás Valenzuela. The calderón's problem via deeponets. *Vietnam Journal of Mathematics*, 52(3):775–806, 2024.

[11] Chuanqi Chen and Jinlong Wu. Neural operator for modeling dynamic systems. *arXiv preprint arXiv:2306.XXXX*, 2023.

[12] Minshuo Chen, Haoming Jiang, Wenjing Liao, and Tuo Zhao. Nonparametric regression on low-dimensional manifolds using deep relu networks: function approximation and statistical recovery. *Information and Inference: A Journal of the IMA*, 11(4):1203–1253, 03 2022.

[13] T. Chen and H. Chen. Approximations of continuous functionals by neural networks with application to dynamic systems. *IEEE Transactions on Neural Networks*, 4(6):910–918, 1993.

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

[15] Maarten V. de Hoop, Daniel Zhengyu Huang, Elizabeth Qian, and Andrew M. Stuart. The cost-accuracy trade-off in operator learning with neural networks, 2022.

[16] Eleonora Di Nezza, Giampiero Palatucci, and Enrico Valdinoci. Hitchhiker's guide to the fractional sobolev spaces. *Bulletin des Sciences Mathématiques*, 2012.

[17] Takashi Furuya, Michael Anthony Puthawala, Matti Lassas, and Maarten V. de Hoop. Globally injective and bijective neural operators. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[18] Craig R. Gin, Daniel E. Shea, Steven L. Brunton, and J. Nathan Kutz. Deepgreen: deep learning of green's functions for nonlinear boundary value problems. *Scientific Reports*, 11(1):21614, 2021.

[19] Somdatta Goswami, Aniruddha Bora, Yue Yu, and George Em Karniadakis. *Physics-Informed Deep Neural Operator Networks*, pages 219–254. Springer International Publishing, Cham, 2023.

[20] Alex Graves, Abdel rahman Mohamed, and Geoffrey E. Hinton. Speech recognition with deep recurrent neural networks. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649, 2013.

[21] Philipp Grohs, Samuel Lanthaler, and Margaret Trautner. Theory-to-practice gap for neural networks and neural operators, 2025.

[22] Jiequn Han, Arnulf Jentzen, and Weinan E. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*, 115(34):8505–8510, 2018.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2016.

[24] Maximilian Herde, Bogdan Raonic, Tobias Rohner, Roger Käppeli, Roberto Molinaro, Emmanuel de Bezenac, and Siddhartha Mishra. Poseidon: Efficient foundation models for PDEs. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[25] Lukas Herrmann, Christoph Schwab, and Jakob Zech. Neural and spectral operator surrogates: unified construction and expression rate bounds. *Advances in Computational Mathematics*, 50(4):72, 2024.

[26] Daniel Zhengyu Huang, Nicholas H. Nelsen, and Margaret Trautner. An operator learning perspective on parameter-to-observable maps. *Foundations of Data Science*, 7(1):163–225, 2025.

[27] Zhongyi Jiang, Min Zhu, Dongzhuo Li, Qiuzi Li, Yanhua O. Yuan, and Lu Lu. Fourier-mionet: Fourier-enhanced multiple-input neural operators for multiphase modeling of geological carbon sequestration. *arXiv preprint arXiv:2303.04778*, 2023.

[28] Pengzhan Jin, Shuai Meng, and Lu Lu. Mionet: Learning multiple-input operators via tensor product. *SIAM Journal on Scientific Computing*, 44(6):A3490–A3514, 2022.

[29] Derek Jollie, Jingmin Sun, Zecheng Zhang, and Hayden Schaeffer. Time-series forecasting and refinement within a multimodal pde foundation model. *Journal of Machine Learning for Modeling and Computing*, 6(2):77–89, 2025.

[30] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020.

[31] Yuehaw Khoo, Jianfeng Lu, and Lexing Ying. Solving parametric pde problems with artificial neural networks. *European Journal of Applied Mathematics*, 32(3):421–435, 2021.

[32] Nikola Kovachki, Samuel Lanthaler, and Siddhartha Mishra. On universal approximation and error bounds for fourier neural operators. *J. Mach. Learn. Res.*, 22(1), January 2021.

[33] Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: learning maps between function spaces with applications to pdes. *J. Mach. Learn. Res.*, 24(1), January 2023.

[34] Nikola B. Kovachki, Samuel Lanthaler, and Hrushikesh Mhaskar. Data complexity estimates for operator learning, 2024.

[35] Nikola B. Kovachki, Samuel Lanthaler, and Andrew M. Stuart. Chapter 9 - operator learning: Algorithms and analysis. In Siddhartha Mishra and Alex Townsend, editors, *Numerical Analysis Meets Machine Learning*, volume 25 of *Handbook of Numerical Analysis*, pages 419–467. Elsevier, 2024.

[36] Samuel Lanthaler. Operator learning with pca-net: upper and lower complexity bounds. *J. Mach. Learn. Res.*, 24(1), January 2023.

[37] Samuel Lanthaler, Siddhartha Mishra, and George E Karniadakis. Error estimates for deeponets: a deep learning framework in infinite dimensions. *Transactions of Mathematics and Its Applications*, 6(1):tnac001, 03 2022.

[38] Samuel Lanthaler and Andrew M Stuart. The parametric complexity of operator learning. *IMA Journal of Numerical Analysis*, page draf028, 08 2025.

[39] Jose Antonio Lara Benitez, Takashi Furuya, Florian Faucher, Anastasis Kratsios, Xavier Tricoche, and Maarten V. de Hoop. Out-of-distributional risk bounds for neural operators with applications to the helmholtz equation. *J. Comput. Phys.*, 513(C), September 2024.

[40] Bian Li, Hanchen Wang, Shihang Feng, Xiu Yang, and Youzuo Lin. Solving seismic wave equations on variable velocity models with fourier neural operator. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–18, 2023.

[41] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Multipole graph neural operator for parametric partial differential equations. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[42] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations (ICLR)*, 2021. preprint arXiv:2010.08895.

[43] Hao Liu, Haizhao Yang, Minshuo Chen, Tuo Zhao, and Wenjing Liao. Deep nonparametric estimation of operators between infinite dimensional spaces. *J. Mach. Learn. Res.*, 25(1), January 2024.

[44] Hao Liu, Zecheng Zhang, Wenjing Liao, and Hayden Schaeffer. Neural scaling laws of deep relu and deep operator network: A theoretical study, 2024.

[45] Yuxuan Liu, Jingmin Sun, Xinjie He, Griffin Pinney, Zecheng Zhang, and Hayden Schaeffer. PROSE-FD: A multimodal PDE foundation model for learning multiple operators for forecasting fluid dynamics. In *Neurips 2024 Workshop Foundation Models for Science: Progress, Opportunities, and Challenges*, 2024.

[46] Yuxuan Liu, Jingmin Sun, and Hayden Schaeffer. Bcat: A block causal transformer for pde foundation models for fluid dynamics. *arXiv preprint arXiv:2501.18972*, 2025.

[47] Yuxuan Liu, Zecheng Zhang, and Hayden Schaeffer. Prose: Predicting multiple operators and symbolic expressions using multimodal transformers. *Neural Networks*, 180:106707, 2024.

[48] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021.

[49] Lu Lu, Xuhui Meng, Shengze Cai, Zhiping Mao, Somdatta Goswami, Zhongqiang Zhang, and George Em Karniadakis. A comprehensive and fair comparison of two neural operators (with practical extensions) based on fair data. *Computer Methods in Applied Mechanics and Engineering*, 393:114778, 2022.

[50] Carlo Marcati and Christoph Schwab. Exponential convergence of deep operator networks for elliptic partial differential equations. *SIAM Journal on Numerical Analysis*, 61(3):1513–1545, 2023.

[51] Carlo Marcati and Christoph Schwab. Expression rates of neural operators for linear elliptic pdes in polytopes. *CoRR*, abs/2409.17552, 2024.

[52] Ivan Markovsky. *Low Rank Approximation: Algorithms, Implementation, Applications*. Communications and Control Engineering. Springer London, 1st edition, 2012.

[53] Michael McCabe, Bruno Régaldo-Saint Blancard, Liam Holden Parker, Ruben Ohana, Miles Cranmer, Alberto Bietti, Michael Eickenberg, Siavash Golkar, Géraud Krawezik, Francois Lanusse, et al. Multiple physics pretraining for physical surrogate models. *arXiv preprint arXiv:2310.02994*, 2023.

[54] Christian Moya, Guang Lin, Tianqiao Zhao, and Meng Yue. On approximating the dynamic response of synchronous generators via operator learning: A step towards building deep operator-based power grid simulators. *arXiv preprint arXiv:2301.12538*, 2023.

[55] Elisa Negrini, Yuxuan Liu, Liu Yang, Stanley J Osher, and Hayden Schaeffer. A multimodal pde foundation model for prediction and scientific text descriptions. *arXiv preprint arXiv:2502.06026*, 2025.

[56] Weigutian Ou and Helmut Bölcskei. Covering numbers for deep relu networks with applications to function approximation and nonparametric regression, 2024.

[57] Weigutian Ou, Philipp Schenkel, and Helmut Bölcskei. Three quantization regimes for relu networks, 2024.

[58] Jaideep Pathak, Shashank Subramanian, Peter Harrington, Sanjeev Raja, Ashesh Chattopadhyay, Morteza Mardani, Thorsten Kurth, David Hall, Zongyi Li, Kamyar Azizzadenesheli, and Anima Anandkumar. Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators. *arXiv preprint arXiv:2202.11214*, 2022.

[59] Christoph Schwab, Andreas Stein, and Jakob Zech. Deep operator network approximation rates for lipschitz operators. *Analysis and Applications*, 24(01):199–239, 2026.

[60] Christoph Schwab and Jakob Zech. Deep learning in high dimension: Neural network expression rates for analytic functions in $\boldsymbol{L^2(\mathbb{R}^d, \gamma_d)}$. *SIAM/ASA Journal on Uncertainty Quantification*, 11(1):199–234, 2023.

[61] Jingmin Sun, Yuxuan Liu, Zecheng Zhang, and Hayden Schaeffer. Towards a foundation model for partial differential equations: Multioperator learning and extrapolation. *Physical Review E*, 111(3):035304, 2025.

[62] Jingmin Sun, Zecheng Zhang, and Hayden Schaeffer. Lemon: Learning to learn multi-operator networks, 2025.

[63] Adrien Weihs, Jalal Fadili, and Matthew Thorpe. Discrete-to-continuum rates of convergence for nonlocal p-Laplacian evolution problems. *Information and Inference: A Journal of the IMA*, 13(4):iaae031, 11 2024.

[64] Adrien Weihs, Jingmin Sun, Zecheng Zhang, and Hayden Schaeffer. A deep learning framework for multi-operator learning: Architectures and approximation theory, 2025.

[65] Liu Yang, Siting Liu, Tingwei Meng, and Stanley J Osher. In-context operator learning with data prompts for differential equation problems. *Proceedings of the National Academy of Sciences*, 120(39):e2310142120, 2023.

[66] Liu Yang, Tingwei Meng, Siting Liu, and Stanley J Osher. Prompting in-context operator learning with sensor data, equations, and natural language. *arXiv preprint arXiv:2308.05061*, 2023.

[67] Zhanhong Ye, Zining Liu, Bingyang Wu, Hongjie Jiang, Leheng Chen, Minyan Zhang, Xiang Huang, Qinghe Meng Zou, Hongsheng Liu, and Bin Dong. Pdeformer-2: A versatile foundation model for two-dimensional partial differential equations. *arXiv preprint arXiv:2507.15409*, 2025.

[68] Benjamin J Zhang, Siting Liu, Stanley J Osher, and Markos A Katsoulakis. Probabilistic operator learning: generative modeling and uncertainty quantification for foundation models of differential equations. *arXiv preprint arXiv:2509.05186*, 2025.

[69] Zecheng Zhang. Modno: Multi-operator learning with distributed neural operators. *Computer Methods in Applied Mechanics and Engineering*, 431:117229, 2024.

[70] Zecheng Zhang, Wing Tat Leung, and Hayden Schaeffer. A discretization-invariant extension and analysis of some deep operator networks. *Journal of Computational and Applied Mathematics*, 456:116226, 2025.

[71] Zecheng Zhang, Christian Moya, Lu Lu, Guang Lin, and Hayden Schaeffer. D2no: Efficient handling of heterogeneous input function spaces with distributed deep neural operators. *Computer Methods in Applied Mechanics and Engineering*, 428:117084, 2024.

[72] Zecheng Zhang, Leung Wing Tat, and Hayden Schaeffer. Belnet: basis enhanced learning, a mesh-free neural operator. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 479(2276):20230043, 2023.

[73] Min Zhu, Shihang Feng, Youzuo Lin, and Lu Lu. Fourier-deeponet: Fourier-enhanced deep operator networks for full waveform inversion with improved accuracy, generalizability, and robustness. *arXiv preprint arXiv:2305.17289*, 2023.