Neural Network Approximation and Structural Complexity of Semilinear PDEs with Lower Bounds

Hayden Schaeffer

Abstract

Neural networks are commonly used to learn the solution operators of PDEs such as the Navier-Stokes equations and viscous conservation laws. On standard benchmarks, moderately sized and often shallow networks achieve small errors that decay at algebraic rates with the network size. These observations are not well described by the current approximation theory, which treats the solution operator as a generic map and derives worst-case bounds over large classes of operators. We show that the structure of the equations and their properties explain the observed behavior. For semilinear parabolic equations, the approximation costs separate into two terms, where one is controlled by an invariant of the governing law and the other by the data. The constitutive law can be parameterized as a single block whose size for accuracy η is $O(\eta^{-p\chi}\log\eta^{-1})$ for every $p > p_f$, where the structural exponent p_f only depends on the computation graph of f. This exponent does not depend on the dimension and is in fact zero for polynomial laws, so for the two-dimensional Navier-Stokes equations the nonlinear block costs only $O(\log\eta^{-1})$ parameters. The data enters through encoding layers with a bounded number of learned weights per mode, so neither term grows with the grid resolution or the time step. While the analysis only uses the max-norm error of the pointwise block, we show that one obtains approximation bounds in a stronger norm along the whole trajectory. For viscous Hamilton-Jacobi equations with a C^s Hamiltonian, the approximation requires on the order of $\eta^{-1/s}$ parameters, and we prove a matching lower bound, up to logarithms. This shows that solution operators for this class of problems have low structural complexity, even though they are nonlinear maps between infinite-dimensional spaces.

1 Introduction

Let $\Omega \subset \mathbb{R}^{d_x}$ be a bounded spatial domain and consider the semilinear evolution equation

$$u_t + Lu = f(u, \nabla u), \qquad u(0) = u_0, \tag{1}$$

where L is a linear differential operator and f the nonlinear constitutive law depending on the state and its derivative. This class of partial differential equations (PDE) covers a wide range of important dynamical models in science and engineering, for example, viscous Burgers and viscous conservation laws, viscous Hamilton–Jacobi equations, reaction-diffusion systems e.g. Allen-Cahn and the diffusive FitzHugh-Nagumo system, Cahn–Hilliard, and the two-dimensional incompressible Navier-Stokes equations [29, 22, 62, 77, 76]. In scientific machine learning (SciML), these systems and their associated solution operators have become benchmark examples to test architectures and algorithms [45, 52, 4, 36, 50, 73, 30, 28].

The goal of these methods is to approximate the solution operator $G_T : u_0 \mapsto u$, which maps the initial data either to the solution at the final time $t = T$ [45, 35, 10] or to the full trajectory on $[0, T]$ [50, 48, 73, 26, 75, 49]. We adopt the latter formulation throughout this work, which is a stronger requirement. Since these equations serve as test cases throughout the literature for learning solution operators with neural networks, we study the structural complexity of approximating their solution operators [45, 52, 36, 50]. In particular, we quantify how many trainable parameters are needed to approximate the solution operator to accuracy η, in the sense of the parametric complexity for operator learning [42]. Our aim is to analyze these systems in a way that allows for a nuanced separation between the quantities that control the cost of approximation. This error leads to terms that depend on the complexity of the nonlinear governing law, the derived regularity from the dynamics, the spatial and temporal resolution, the encoding of the input, and the contraction of the iterates. We show that the first two terms require trainable parameters, and that

their total number does not grow with the grid size or resolution. The remaining error terms can affect the size of the network; however, they do not change the number of parameters that must be learned. In this way, we show that the solution operators of semilinear PDEs have low *structural complexity*, which provides a theoretical explanation for why learned neural approximations perform well on them, even though they are seemingly complicated nonlinear operators between infinite-dimensional spaces.

In practice neural operators [52, 45, 36, 4, 91] utilize an encoder to map the inputs into a latent space. For Fourier Neural Operators (FNO), this is the spectral domain given by the fast Fourier transform (FFT), and for PCA-based networks, it is the span of the leading principal components of the data generated by a PCA layer [45, 4]. Most neural operators utilize linear layers (affine or convolutional) and point-wise nonlinear activation functions applied with the same weights at every spatial location [36, 45, 52, 4]. In empirical studies, neural operators are more parameter efficient than what the theory of generic nonlinear operators shows. Specifically, in operator learning theory the rates have a curse of parametric complexity [42, 47] which yields approximation and sampling bounds that are exponentials of power laws. However, simulations show that the errors can decay at algebraic rates, for example in the size of the training set [52, 15, 72], and can be efficient over larger multi-operator settings [50, 49, 58, 30, 28, 9].

One theoretical limitation is that the universal approximation results [13, 83] and the quantitative complexity results for the class of Lipschitz or smooth operators [42, 47, 82] take the solution operator G_T to be a generic map between Banach spaces. Since these operators are only characterized by their Lipschitz or C^r regularity on infinite-dimensional spaces, the metric entropy grows like $\exp(\eta^{-p})$, and the associated cost for standard architectures is the same. In particular, this leads to exponential scalings for the parameter count [42, 39, 47] and generalization rates that are only logarithmic in terms of the number of samples [37, 1, 47, 81, 82]. To get algebraic approximation rates for Lipschitz operators, one needs non-standard architectures such as super-expressive activation functions or non-standard ReLU architectures [88, 69] which is consistent with the metric-entropy bound for encoding information [39, 47].

Utilizing the structure of the operator or data can help to avoid the curse of parametric complexity. One approach is to assume holomorphic dependence on the input, which yields algebraic rates [70, 61, 31, 54, 65]. For elliptic problems, the size of networks that can approximate the solution scales like the size of networks representing the coefficients or the nonlinearity [56, 55]. Another direction is to relate the architecture to the PDE by showing that it can emulate a numerical scheme. Examples include the Picard iteration for scalar reaction laws [26, 75, 24], learned time-stepping operators [11], iterated one-step operators for convex monotone semigroups [6], and emulating stable spectral schemes [10]. For FNO that emulate stable spectral schemes, [10] showed that a polynomial nonlinearity only adds a logarithmic cost, while the number of retained Fourier modes determines the overall rate.

Compared with these works, we take a complementary approach that applies to laws that are nonlinear in $(u, \nabla u)$. We separate the parameter count into a term that is controlled by the governing equation and a term controlled by the derived regularity of the data. We show that both terms are necessary. We use three properties of semilinear parabolic PDEs that are absent for a generic Lipschitz operator:

1. **Fixed-point Structure:** On a short horizon, the solutions are the unique fixed points of the mild map

$$\Phi[u](t) = S(t)u_0 + \int_0^t S(t-s)f(u(s), \nabla u(s))\, ds,$$

   which is a contraction on a ball in $C([0,T]; X^\alpha)$. Thus, a solution operator can be taken as the limit of an iteration whose only nonlinearity comes from f.

2. **Parabolic Smoothing:** The semigroup $S(t) = e^{-tL}$ is analytic and smoothing, so certain high frequencies are damped. Therefore, we can quantify the truncation and quadrature errors with explicit algebraic rates induced directly from the dynamics itself.

3. **Algebraic Parametric Complexity of the Governing Law:** The term f can be written as a finite directed acyclic graph (DAG) made from elementary operations, and is independent of any discretization. The cost of approximating the governing law is determined by the structure of the DAG rather than by the number of variables of f or the number of space-time points.

Combining these properties allows a standard architecture to use one small point-wise neural network whose parameters are shared across grid points, time levels and iterations, and thus keeps the number of trainable parameters small.

1.1 Main result

The analysis uses the following pipeline to chain the approximations (see Figure 1). The Galerkin truncation, temporal quadrature, and spatial sampling are used to form a discrete Picard map that approximates the mild map. The governing law f is represented by its DAG form and is replaced by a deep ReLU network which approximates each node in the DAG. To apply it over the grid, weight sharing through a 1×1 convolution (Lemma 3.4) is used. This results in a looped structure that mimics a fixed-point method.

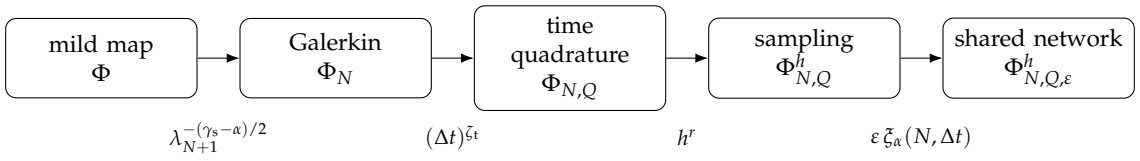


Figure 1: **Approximation Pipeline and Associated Errors.** All steps are discretizations except the last one, which replaces the governing law by a shared ReLU network and the corresponding trainable parameters. The linear steps utilize fixed transforms, learned spectral weights, and parameter sharing. This leads to a loop, which is illustrated in Figure 3.

Let $X^s = D(L^{s/2})$ be the space generated by L, γ_s be the spatial regularity of the data, and $\zeta_t \le 1$ be the resulting temporal order of accuracy (see Section 1.4 for notation). The main complexity result is informally stated below and summarizes the rigorous results Theorems 5.6 and 5.14.

Theorem 1.1 (Main result, informal)**.** *Let u be the mild solution of* (1) *on $[0,T]$ with spatial regularity γ_s. If f can be written as a finite DAG whose nonlinear nodes are multilinear or component-wise C^{s_j}, with $p_{\max} = 1/s_{\min}$, then for every sufficiently small $\eta > 0$, the architecture from Definition 4.7, with*

$$\lambda_{N+1} \asymp \eta^{-2/(\gamma_s-\alpha)}, \qquad h \asymp \eta^{1/(\gamma_s-\alpha)}, \qquad \Delta t \asymp \eta^{1/\zeta_t}, \qquad n \asymp \log \eta^{-1},$$

admits a choice of trainable parameters for which $\|\widehat{u} - u\|_{C([0,T];X^\alpha)} \le \eta$ holds. Its trainable parameters scale as

$$\underbrace{W(\widehat{f}_\varepsilon) \;\le\; C\,\eta^{-p_{\max}\chi_s}\log\eta^{-1}}_{\textit{law}} \qquad + \qquad \underbrace{W_{\text{spec}} \;\le\; C\,\eta^{-d_x/(\gamma_s-\alpha)}}_{\textit{data}},$$

and are independent of the number of grid points, time levels, and sweeps. The full network without parameter sharing scales like

$$W(\mathcal{A}_n) \;\le\; C\,\eta^{-v}\left(\log\eta^{-1}\right)^2, \qquad v = \frac{d_x}{\gamma_s-\alpha} + \frac{1}{\zeta_t} + p_{\max}\chi_s, \qquad \operatorname{dep}(\mathcal{A}_n) \;\le\; C\left(\log\eta^{-1}\right)^2,$$

where $\chi_s \ge 1$ equals 1 for $\alpha < 2$ and decreases to 1 as the data becomes smoother, with an extra factor $(\log\eta^{-1})^{p_{\max}}$ when $\alpha = 2$.

If every node is multilinear, then $v = \frac{d_x}{\gamma_s-\alpha} + \frac{1}{\zeta_t}$ and the complexity is set only by the space-time resolution.

The exponent $p_{\max} = 1/s_{\min}$ is the classical ReLU rate [86, 18] applied to the least smooth nonlinear node in the DAG form of f. It is the only place where the smoothness of f appears explicitly in the bounds. It is multiplied by the stability factor χ_s, which comes from the lack of smoothing by the Duhamel kernel when we measure the error in X^α, $\alpha > 2$. This is sharp (see Proposition 7.1), but under some assumptions it can be improved (see Section 7.1).

1.2 Contributions

We list some of the contributions of this work.

1. **Separation of the Parameter Count (Theorems 5.6 and 5.14).** We split the number of trainable parameters needed to obtain an accuracy of η in $C([0,T];X^\alpha)$ into a term controlled by the governing law, i.e. $C\eta^{-p\chi_s}\log\eta^{-1}$, and a term controlled by the regularity of the data, i.e. $C\eta^{-d_x/(\gamma_s-\alpha)}$. The terms do not grow with the grid, the time step, or the number of iterations of the looped network due to parameter sharing. These results also apply to popular operator learning approaches, for example, those based on FNO or DeepONet (see Remarks 4.8 and 4.9).

2. **The Structural Exponent (Section 3).** For governing laws whose DAG depends on inputs $(u, \nabla u)$, the complexity is set by the least smooth nonlinear node (see Theorem 3.7), building on [68, 64]. Taking the infimum over all DAG forms of f yields the structural exponent p_f which is an invariant of the system and does not depend on d, d_x or the number of state components. For polynomial systems, it is equal to zero.

3. **A Derived Error Decomposition (Section 5).** All of the sources of error and assumptions on the regularity are derived from the mild formulation. They are bounded separately rather than assumed through the accuracy of a particular scheme (Theorem 5.4). The error is measured in $C([0,T];X^\alpha)$ which is a stronger norm than is typically used [26, 75, 10] but still achieves a scaling for the depth that is only $C(\log\eta^{-1})^2$ (see Theorem 5.13), the same as in [26, 24]. The decomposition is written in the interpretable form of

$$\underbrace{\kappa^n}_{\text{iteration}} + \underbrace{C\varepsilon\,\xi_\alpha(N,\Delta t)}_{\text{network}} + \underbrace{\varepsilon_N^{\text{tail}}(u_0) + \varepsilon_{N,h}^{\text{enc}}(u_0)}_{\text{encoding}} + \underbrace{C\lambda_{N+1}^{-(\gamma_s-\alpha)/2}}_{\text{Galerkin}} + \underbrace{Ch^r}_{\text{spatial}} + \underbrace{C(\Delta t)^{\zeta_t}}_{\text{temporal}}.$$

   For $\alpha > 2$ the network error is amplified, which multiplies its exponent by χ_s. This amplification is sharp by Proposition 7.1 in general, but can be removed by certain formulations for gradient-free laws when $d_x \le 3$ (see Section 7.1).

4. **Lower Bounds (Section 7).** Any architecture whose weights depend continuously on the governing laws needs at least $c\,\eta^{-1/(s+1)}$ parameters for gradient-free governing laws and at least $c(\eta\log\eta^{-1})^{-1/s}$ over a class of gradient dependent governing laws. For gradient dependent laws and $s \ge 2$, the lower bound rates match the upper bound rates up to logs (see Theorems 7.3 and 7.5). If the spectrum of L is trainable, at least one weight mode is required, so the data term in our analysis is sharp (Proposition 7.6).

In addition, our results are extended to longer time intervals with the same network (Proposition 5.9). For gradient-free laws, the error is controlled by a one-sided rate rather than only using contraction (Lemma 5.10). This also extends to the solution operator using just one set of weights for all data in some fixed ball (Theorem 6.5). We compute the exponents for seven commonly used PDEs and examine the constants in the Allen-Cahn case, where the block has a few hundred parameters on a single window and the trainable count is dominated by the spectral weights (Section 8).

1.3 Related work

There are several quantitative results for learning the solutions to PDEs using neural networks. In [40], error estimates are provided for DeepONet, and in [47], approximation and generalization rates are provided over the class of Lipschitz operators. Generic error bounds for operator learning that apply to nonlinear parabolic equations were shown in [17]. Error bounds were derived for FNO in [35], and upper and lower complexity bounds for PCA-Net can be found in [38]. Approximation and generalization bounds for encoder-decoder architectures were analyzed in [85]. In [89], universal approximation was derived for a discretization-invariant extension of DeepONet, where the network has an additional component to resolve the grids, and general universal approximation results appeared in [83]. In [41], they proved lower

bounds showing that operator learning approaches that use a linear reconstruction step, e.g. DeepONet and PCA-Net, cannot efficiently approximate discontinuous solutions. In [25], they showed that bijective neural operators between infinite-dimensional Hilbert spaces may not admit continuous finite-dimensional discretizations. In [43], the authors examined the discretization error associated with evaluating an FNO on a grid and proved an algebraic rate with respect to the resolution.

As shown in [42], the curse of parametric complexity over Lipschitz or C^s classes leads to parameter counts of $\exp(\varepsilon^{-r})$ for operator learning, including DeepONet, PCA-Net, and FNO. However, using structures or other information can alleviate this restriction. FNO can emulate pseudo-spectral methods for periodic domains using C^3 activation functions (not ReLU), and so can approximate the Darcy and Navier-Stokes solution operators with explicit bounds [35]. In [40], they used the Cole-Hopf transformation to obtain explicit DeepONet bounds for viscous Burgers. In our results, the bound follows from the bilinear structure of the law. In [38], it is shown that PCA-Net can avoid the curse of parametric complexity for Darcy flow by using the structure of the parametrized coefficient and avoids it for the Navier-Stokes equations through emulation of a numerical method. Their results are measured in the averaged error. For the setting of self-adjoint linear operators, [16] showed that the generalization rates are algebraic and depend on the eigenvalue decay and data smoothness. In [10], under the assumption that the solution operator has a stable and accurate Fourier spectral discretization, they proved that FNO learns an accurate approximation with polynomial sample complexity. They show that the complexity for the problem $u_t + \mathcal{A}u + \mathcal{D}_s(\mathcal{G}(u)) = f$ on the torus, under mild conditions, leads to algebraic rates in L^2 and provide generalization bounds.

In [26, 75], neural operators were constructed using Picard iterations in order to approximate the solution operators of scalar reaction equations $u_t + Lu = F(u)$. Within these networks, the nonlinear scalar F is a Lipschitz function and can be approximated with order ϵ^{-1} neurons (up to logs), thus a polynomial and a Lipschitz nonlinearity scale the same. In [26, 75], the authors noted that these Picard layers share weights and that the networks' depth scales like $(\log \epsilon^{-1})^2$. In both of these papers, the linear term is approximated without a rate and discretization is not analyzed. In the recent work [24], Picard layers with a truncated eigenbasis were utilized for reaction-diffusion systems, with explicit bounds in the number of retained modes and depth scaling $C(\log \epsilon^{-1})^2$.

Our work complements these prior results, since we consider governing equations that can be nonlinear in ∇u on domains with an explicit eigenbasis, and derive the explicit conditions for the numerical emulator. In addition, we show the error bound over the whole trajectory in stronger norms induced by the dynamics, with upper and lower rates.

The architecture has loop structure due to the unrolling of fixed-point iteration, as seen in the learned iterative shrinkage [27], algorithm unrolling [59] and deep equilibrium models [3], including weight-tied Fourier neural operators for steady-state PDEs [57]. Banach fixed-point iterations unrolled into a loop were shown to be general operator networks in [23], where their depth is logarithmic in accuracy given an accurate approximation of the iteration steps. The loop structure also allows for parameter sharing across layers, utilized in neural ordinary differential equations [12, 74]. In [11], they estimated the generalization error of networks that learn time-stepping operators of implicit schemes, and the authors of [6] approximated convex monotone semigroups using a learned one-step operator. In both of these results, the learned time-stepper can be composed to generate predictions. Multilevel Picard methods and the associated neural network emulator can approximate single solutions of high-dimensional semilinear parabolic equations with only polynomially many parameters [21, 33, 60]. Recent work has extended the learning task to families of governing laws, either through in-context and multimodal operator learning [9, 50], multi-task operators and PDE foundation models [73, 48, 49, 58, 30, 28], and operators with heterogeneous inputs [34, 90]. The analysis of multi-task operator learning, including approximation and generalization was studied in [83, 81, 82]. Specifically, in [82], the theoretical rates for learning families of operators are equivalent to those of learning one operator with the same Lipschitz regularity. In [80], the rate for multi-task kernel learning is controlled by the hardest constituent task, which is similar to the least regular node in our analysis.

1.4 Organization and Notation

Section 2 introduces the function spaces, the mild map, and the systems' derived regularity. Section 3 provides the DAG approximation and the structural exponent used through the main results. In Section 4,

we analyze the discretization and the neural Picard map. The main results for one solution are in Section 5, which shows the total error, the complexity theorem, and the size of the full network, and extend to operator learning in Section 6. Sharpness and lower bounds are derived in Section 7. Lastly, we show that the theory applies to a range of examples in Section 8. A collection of proofs is deferred to Appendices A and B.

We use the following notation throughout this work. First, H is a real Hilbert space, L is as above with eigenpairs (λ_k, e_k), and define the spaces $X^s = D(L^{s/2})$, $X_T^s = C([0,T]; X^s)$. The exponents satisfy $0 \le \beta < \alpha < \gamma$ with $\alpha - \beta < 2$ and $\gamma - \beta < 2$, and we set

$$\sigma := \frac{\alpha - \beta}{2}, \qquad \theta := 1 - \sigma \in (0,1), \qquad \zeta := \frac{\gamma - \alpha}{2}, \qquad \bar{\sigma} := \frac{\gamma - \beta}{2}, \qquad \chi := 1 + \frac{(\alpha - 2)_+}{2\zeta},$$

so $\sigma + \zeta = \bar{\sigma}$ and $\zeta < \theta$. The term χ is the amplification of the network error by the discrete Duhamel kernel (see Theorem 4.21). It is equal to 1 when $\alpha \le 2$ and equals $\frac{\gamma-2}{\gamma-\alpha}$ for $\alpha > 2$. The state variable's dimension is d, i.e. $u(t,x) \in \mathbb{R}^d$, and the spatial dimension is denoted by d_x. Thus the gradient is $\nabla u(t,x) \in \mathbb{R}^{d \times d_x}$. For a ReLU network $\mathcal{N}$, we denote the depth by $\mathrm{dep}(\mathcal{N})$ and the number of nonzero parameters by $W(\mathcal{N})$. The term $\mathcal{D}$ denotes the Duhamel operator and $D(L)$ the domain of L. The main symbols are summarized in Table 1.

Symbol	Def.	Symbol	Def.
$L,\ \lambda_k,\ e_k$	elliptic operator, eigenpairs	N	retained Galerkin modes
$X^s = D(L^{s/2})$	Hilbert spaces	$h,\ M(h)$	mesh size, sample points
$X_T^s = C([0,T]; X^s)$	trajectory space	$Q,\ \Delta t = T/Q$	time levels, step
$\beta < \alpha < \gamma$	forcing, well-posedness, regularity	n	Picard iterations
$\sigma = \frac{\alpha-\beta}{2}, \theta = 1-\sigma$	Duhamel improvement	ε	local network tolerance
$\zeta = \frac{\gamma-\alpha}{2}, \bar{\sigma} = \sigma + \zeta$	regularity improvement	η	target accuracy
$\gamma_{\mathrm{s}} \ge \gamma, \zeta_{\mathrm{s}} = \frac{\gamma_{\mathrm{s}}-\alpha}{2}, \zeta_{\mathrm{t}}$	spatial index, temporal order	W_{spec}	learned spectral weights
$\chi = 1 + \frac{(\alpha-2)_+}{2\zeta}$	error amplification exponent	υ	full-network exponent
$\Phi,\ \mathcal{D}$	mild map, Duhamel operator	$\widehat{f_\varepsilon}$	shared local network
$\Psi^h_{N,Q},\ \Psi^h_{N,Q,\varepsilon}$	discrete, neural Picard maps	$\xi_\alpha(N, \Delta t)$	network-error amplification
a, b, a_∞	coefficient sequences	$\kappa_0,\ \kappa$	contraction constants
$\mathcal{L}_{N,Q},\ \mathcal{J}_Q$	piecewise-linear lifting	$\mathcal{L}^S_{N,Q}$	semigroup reconstruction
δ	DAG node tolerance	$p_j,\ p_{\max},\ p_f$	DAG node, graph, structural exponents
$\mathcal{A}_n$	n-sweep network	$\Pi_{N,h}$	discrete encoder

Table 1: Notation: The exponents $\sigma, \theta, \zeta, \bar{\sigma}$ are fixed by β, α, γ but are used throughout. We also use σ to denote the ReLU activation, which should be clear from context.

We chose to use η for the target accuracy of the trajectory and reserve ε for the accuracy of the point-wise block or local networks, which is smaller when $\alpha > 2$.

2 Preliminaries

We first discuss the functional analytic results that will be used throughout the work, specifically, the Hilbert spaces generated by a positive self-adjoint operator L [53, 76].

Definition 2.1 (Variable Hilbert spaces)**.** *Let H be a Hilbert space and let $L : D(L) \subset H \to H$ be a densely defined, self-adjoint, strictly positive operator with compact inverse. The eigenvalues of L are denoted by*

$$0 < \lambda_1 \le \lambda_2 \le \cdots,$$

where $\lambda_k \to \infty$, and the corresponding orthonormal eigenfunctions are denoted by $\{e_k\}_{k=1}^\infty$. For $s \ge 0$, define the function spaces generated by L by $X^s := D(L^{s/2})$, equipped with the norm

$$\|u\|_{X^s} := \|L^{s/2} u\|_H = \left(\sum_{k=1}^\infty \lambda_k^s |\langle u, e_k \rangle_H|^2 \right)^{1/2}.$$

For fixed indices $0 \le \beta < \alpha < \gamma$ with $\alpha \ge 1$, the following three spaces, X^γ, X^α, and X^β, denote the regularity space, the well-posedness space, and the space for the nonlinear governing equation, respectively. In particular, the spaces are continuously embedded, $X^\gamma \hookrightarrow X^\alpha \hookrightarrow X^\beta$. For $s \ge 0$ and $T > 0$, define $X_T^s := C([0,T]; X^s)$ equipped with the norm

$$\|u\|_{X_T^s} := \sup_{0 \le t \le T} \|u(t)\|_{X^s},$$

which preserves the same embedding order $X_T^\gamma \hookrightarrow X_T^\alpha \hookrightarrow X_T^\beta$.

For simplicity, we assume that X^α is smooth enough so that point-wise evaluation of u and ∇u is well-defined.

Assumption 2.2 (C^1 Embedding)**.** *For the well-posedness index α, assume that $X^\alpha \hookrightarrow C^1(\overline{\Omega}; \mathbb{R}^d)$ continuously, i.e., there exists a constant $C_{\text{emb}} > 0$ such that*

$$\|u\|_{C^1(\overline{\Omega})} \le C_{\text{emb}} \|u\|_{X^\alpha}.$$

With this assumption, for any $u \in X^\alpha$, the inputs to the network are well-defined for standard computations. One can weaken this assumption at the cost of additional density and approximation terms.

For an explicit example, consider the space $H = L^2(\Omega)$ and the standard Laplacian $L = -\Delta$ on a bounded domain $\Omega \subset \mathbb{R}^{d_x}$, with boundary conditions that force L to be self-adjoint and strictly positive with compact inverse (e.g. Dirichlet BC with $D(L) = H^2(\Omega) \cap H_0^1(\Omega)$). Up to norm equivalence, X^s is the Sobolev space $H^s(\Omega)$ with the corresponding boundary conditions, and $X^\gamma \hookrightarrow X^\alpha \hookrightarrow X^\beta$ is the Sobolev embedding. Since $H^\alpha(\Omega) \hookrightarrow C^1(\overline{\Omega})$ for $\alpha > 1 + \frac{d_x}{2}$ [22], Assumption 2.2 holds. The same characterization holds if we extend L to be a uniformly elliptic second-order operator [46, 53]. We will keep the abstract formulation in terms of L, without committing to a domain, boundary condition, or eigenbasis.

For a semigroup $S(t)$, the Duhamel operator is defined by

$$\mathcal{D}[g](t) := \int_0^t S(t-s) g(s)\, ds.$$

The following properties for the Duhamel operator lead to estimates and regularity bounds.

Assumption 2.3 (Analytic Semigroup Regularity)**.** *The operator L generates an analytic semigroup on H denoted by $S(t) = e^{-tL}$ for $t \ge 0$. The semigroup preserves the Hilbert spaces $\{X^s\}_{s \ge 0}$, i.e., for $s \ge 0$,*

$$\|S(t)v\|_{X^s} \le \|v\|_{X^s},$$

for $t \ge 0$, and for $t > 0$ it smooths in the sense

$$\|S(t)v\|_{X^r} \le C_{r-s}\, t^{-(r-s)/2} \|v\|_{X^s},$$

where $r \ge s$. Thus, if $0 \le s < r$ and $r - s < 2$, then the Duhamel operator satisfies

$$\|\mathcal{D}[g]\|_{X_T^r} \le C\, T^{1-(r-s)/2} \|g\|_{X_T^s},$$

where $C > 0$ is independent of T.

Although stated as an assumption, any L from Definition 2.1 will satisfy Assumption 2.3. Taking the spectral decomposition, $S(t)u = \sum_k e^{-\lambda_k t} \langle u, e_k \rangle e_k$ is contractive on every X^s, and

$$\sup_{\lambda > 0} \lambda^{(r-s)/2} e^{-\lambda t} = C_{r-s} t^{-(r-s)/2}$$

where the constant is defined as $C_{r-s} = \sup_{x>0} x^{(r-s)/2} e^{-x}$. After integration, we get the Duhamel bound with $C = C_{r-s}\,(1 - \frac{r-s}{2})^{-1}$.

We also need that $\mathcal{D}[g] \in C([0,T]; X^r)$ for $g \in C([0,T]; X^s)$ and $0 \le r - s < 2$, so that the mild map acts on X_T^α. To see this, note that since $g([0,T])$ is compact in X^s and the projections P_K (see Definition 2.11)

converge strongly to the identity with norm bounded by 1, we have $\sup_t \|(I - P_K)g(t)\|_{X^s} \to 0$. The function $\mathcal{D}[P_K g]$ is continuous and lives on a finite-dimensional space, so applying the Duhamel bound yields

$$\sup_t \|\mathcal{D}[(I - P_K)g](t)\|_{X^r} \leq CT^{1-(r-s)/2} \sup_t \|(I - P_K)g(t)\|_{X^s} \to 0,$$

for $t \in [0, T]$ and T fixed. Therefore, the functions converge, i.e. $\mathcal{D}[P_K g] \to \mathcal{D}[g]$ in the norm of X_T^r, and since $C([0, T]; X^r)$ is complete, we have $\mathcal{D}[g] \in C([0, T]; X^r)$. This is needed so that Φ is a map from X_T^α into itself in the Banach fixed point arguments.

2.1 Mild map and Fixed Points

The approximation theory uses the fixed-point of the mild map (i.e., the mild solution) as the target function. The regularity and behavior of the mild solution can be directly characterized by the semigroup formulation for semilinear evolution equations [62]. In particular, this covers popular examples including: viscous Burgers-type equations, reaction-diffusion systems, and the Navier-Stokes equations (under certain settings) [77]. We use the mild map and the associated Picard iteration rather than a general time-stepping scheme, so that we can control the bounds via consistency and contraction only, with constants independent of the discretization. This is due to the fact that every time level is updated simultaneously which will lead to a logarithmic depth.

Definition 2.4 (Mild solution)**.** *Consider the initial value problem*

$$u_t + Lu = f(u, \nabla u), \qquad u(0) = u_0. \tag{2}$$

Let $S(t) = e^{-tL}$ denote the semigroup generated by the linear operator $-L$. The mild map $\Phi : X_T^\alpha \to X_T^\alpha$ is defined as

$$\Phi[u](t) := S(t)u_0 + \int_0^t S(t - s)f(u(s), \nabla u(s))\, ds.$$

for $0 \leq t \leq T$. A function $u \in X_T^\alpha$ is called a mild solution of (2) *if and only if it is a fixed point of the mild map Φ, i.e. $u = \Phi[u]$.*

The existence and regularity of the solutions are a consequence of fixed-point analysis.

Proposition 2.5 (Contraction of the Mild Map)**.** *Let $S(t)$ be the analytic semigroup generated by $-L$ and $\mathcal{D}[\cdot]$ be the associated Duhamel operator. Assume that the nonlinear map $u \to f(u, \nabla u)$ maps X_T^α into X_T^β, where $0 \leq \beta < \alpha$ and $\alpha - \beta < 2$. Assume that for every $R > 0$, there exists $L_R > 0$ such that*

$$\|f(u, \nabla u) - f(v, \nabla v)\|_{X_T^\beta} \leq L_R \|u - v\|_{X_T^\alpha},$$

for $\|u\|_{X_T^\alpha}, \|v\|_{X_T^\alpha} \leq R$. For simplicity assume that $f(0, 0) = 0$. Then, for every $u_0 \in X^\alpha$, there exist constants $R > 0$ and $T_0 > 0$ such that, for every $T \in (0, T_0]$, the following hold

1. *The ball is invariant under Φ, i.e., $\Phi(B_R^\alpha) \subseteq B_R^\alpha$, where $B_R^\alpha := \left\{ u \in X_T^\alpha : \|u\|_{X_T^\alpha} \leq R \right\}$.*
2. *The map is contractive with rate $\kappa_0 = CT^{1-(\alpha-\beta)/2} L_R \leq \frac{1}{2}$, i.e.,*

$$\|\Phi[u] - \Phi[v]\|_{X_T^\alpha} \leq \kappa_0 \|u - v\|_{X_T^\alpha}$$

   for all $u, v \in B_R^\alpha$.
3. *The initial value problem has a unique mild solution on $[0, T]$ in B_R^α.*

The proof is a standard Banach fixed-point argument and appears in Appendix A.1. The assumption $f(0, 0) = 0$ is chosen for simplicity and is not essential for the proofs. It is sufficient to assume a growth condition. For example, for every $R > 0$, there exists a constant $C_R > 0$ such that

$$\|f(u, \nabla u)\|_{X_T^\beta} \leq C_R \left(1 + \|u\|_{X_T^\alpha}\right)$$

for all $\|u\|_{X_T^\alpha} \leq R$. Then the invariance proof would follow from

$$\|\Phi[u]\|_{X_T^\alpha} \leq \|u_0\|_{X^\alpha} + CT^{1-(\alpha-\beta)/2} C_R(1+R),$$

and $T > 0$ may be chosen sufficiently small to ensure that $\Phi(B_R^\alpha) \subseteq B_R^\alpha$. The contraction results would not change since it only depends on the local Lipschitz continuity of the nonlinear map f. Thus, we could keep the $f(0,0)$ in the bounds:

$$\|f(u, \nabla u)\|_{X_T^\beta} \leq \|f(0,0)\|_{X_T^\beta} + L_R \|u\|_{X_T^\alpha},$$

and the same argument holds.

We will repeatedly use the implication that, since f acts point-wise in time, the condition on X_T^β is the same as

$$\|f(v, \nabla v) - f(w, \nabla w)\|_{X^\beta} \leq L_R \|v - w\|_{X^\alpha} \qquad \text{for all } v, w \in X^\alpha \text{ with } \|v\|_{X^\alpha}, \|w\|_{X^\alpha} \leq R. \tag{3}$$

The equation (3) is implied directly from the X_T form when applied to the constant trajectories $\widetilde{v} \equiv v$ and $\widetilde{w} \equiv w$ in B_R^α. To show the converse, the X_T formulation follows from (3) after taking a supremum over t. This will be used without reference in several results.

Under certain conditions on the spaces, the solutions have the standard parabolic smoothing property for $t > 0$ [62, 53]. The bounds degrade as t approaches zero.

Proposition 2.6 (Regularity)**.** *Let $u \in X_T^\alpha$ be the mild solution of* (2) *from Proposition 2.5, where $u_0 \in X^\alpha$. Assume that for some $\gamma > \alpha$, we have $\gamma - \beta < 2$. Then, for every $t \in (0, T]$, the mild solution is in X^γ with*

$$\|u(t)\|_{X^\gamma} \leq Ct^{-(\gamma-\alpha)/2} \|u_0\|_{X^\alpha} + Ct^{1-(\gamma-\beta)/2} \|f(u, \nabla u)\|_{X_T^\beta},$$

where $C > 0$ is independent of t. In particular, $u \in C((0, T]; X^\gamma)$.

The proof is given in Appendix A.1.

When approximating the system by the fully discrete form, we will need the following bounds and smoothness, which only require that $u_0 \in X^\alpha$.

Lemma 2.7 (Derived Regularity)**.** *Let $u_0 \in X^\alpha$ with $\|u_0\|_{X^\alpha} \leq R$, let $u \in B_R^\alpha$ be the mild solution of* (2) *on $[0, T]$ obtained by Proposition 2.5, and set*

$$g(t) := f\left(u(t), \nabla u(t)\right), \qquad F_R := \sup_{v \in B_R^\alpha} \|f(v, \nabla v)\|_{X_T^\beta}.$$

Recall the exponents: $\sigma = \frac{\alpha-\beta}{2}$, $\theta = 1 - \sigma$, $\zeta = \frac{\gamma-\alpha}{2}$ and $\bar{\sigma} = \sigma + \zeta = \frac{\gamma-\beta}{2} < 1$. Then the following holds:

1. $\sup_{0 \leq t \leq T} \|g(t)\|_{X^\beta} \leq F_R$;
2. $\sup_{0 < t \leq T} t^\zeta \|u(t)\|_{X^\gamma} \leq R_\gamma := C\left(R + T^\theta F_R\right)$;
3. *for all $0 < s < t \leq T$,*

$$\|u(t) - u(s)\|_{X^\alpha} \leq C(t-s)^\zeta s^{-\zeta} \left(R_\gamma + T^\theta F_R\right),$$

and thus using the weighted Hölder seminorm

$$[g]_\zeta := \sup_{0 < s < t \leq T} \frac{\|g(t) - g(s)\|_{X^\beta}}{(t-s)^\zeta s^{-\zeta}},$$

we have $[g]_\zeta \leq CL_R\left(R_\gamma + T^\theta F_R\right) < \infty$.

The constants in each bound only depend on the differences $\alpha - \beta$ and $\gamma - \alpha$, and the constants of Assumption 2.3.

The proof uses the regularity results in Proposition 2.6 with an intermediate time point s and is derived in Appendix A.1.

Lemma 2.7 only uses that $u_0 \in X^\alpha$ and by construction has the restriction $\gamma < \beta + 2$. This is a consequence of the fact that the Duhamel operator smooths by fewer than two derivatives. For smoother data, this restriction is an artifact since the forcing inherits the regularity of the solution, and the smoothing can be iterated upon. We use a bootstrap argument to show this.

Assumption 2.8 (Higher Regularity)**.** *There is a $\gamma' \geq \gamma$ such that for every $R > 0$ there is K_R with*

$$\|f(v, \nabla v)\|_{X^{s-(\alpha-\beta)}} \leq K_R \left(1 + \|v\|_{X^s}\right),$$

for all $s \in [\alpha, \gamma']$ and $v \in X^s$ with $\|v\|_{X^\alpha} \leq R$.

As an example, on a torus with $X^s = H^s$, this should hold for every governing equation that is smooth in $(u, \nabla u)$, by applying the Moser estimates for compositions [2, 66]. In the setting where the governing equation has sufficient but finite smoothness, this will hold for some γ' up to a limit defined by the governing equation's smoothness.

Lemma 2.9 (Regularity Bootstrap)**.** *If Assumption 2.8 holds, $u_0 \in X^{\gamma'}$ with $\|u_0\|_{X^\alpha} \leq R$, and $u \in B_R^\alpha$ is the mild solution on $[0, T]$, then*

$$\sup_{0 \leq t \leq T} \|u(t)\|_{X^{\gamma'}} \leq R_{\gamma'} := C \left(1 + \|u_0\|_{X^{\gamma'}}\right),$$

and

$$\sup_{0 \leq t \leq T} \|f(u(t), \nabla u(t))\|_{X^{\gamma'-(\alpha-\beta)}} \leq K_R \left(1 + R_{\gamma'}\right),$$

where C depends only on R, T, K_R, $\gamma' - \alpha$, $\alpha - \beta$, λ_1, and the constants of Assumption 2.3.

The proof iterates the Duhamel estimate over steps of size θ (see Appendix A.1).

The lemma lets us extend the spatial regularity γ_s up to an amount determined by the governing equation. This is used for measuring the tail of the Galerkin projection and the spatial quadrature. For notation, we set $\gamma_s := \gamma'$ to be the exponent in Assumption 2.8 when $u_0 \in X^{\gamma'}$, and $\gamma_s := \gamma$ otherwise. Similarly, we denote $\zeta_s := \frac{\gamma_s - \alpha}{2} \geq \zeta$. The order in time is set by the Hölder regularity of the force as a function in time, and Lemma 2.7 implies the exponent is ζ, with the restriction $\gamma < \beta + 2$. This is extended to $\min\{1, \zeta_s\}$ by Lemma 2.10, which can be at most first order. We write $\zeta_t := \min\{1, \zeta_s\}$ when $\gamma_s = \gamma' > \gamma$, and $\zeta_t := \zeta$ otherwise. Note that in both cases, the powers have the following ordering $\zeta \leq \zeta_t \leq \zeta_s$.

Lemma 2.10 (Time Bootstrap)**.** *If the hypotheses of Lemma 2.9 hold, $\zeta' := \frac{\gamma' - \alpha}{2}$, and $\kappa := \min\{1, \zeta'\}$, then for $0 \leq s < t \leq T$*

$$\|u(t) - u(s)\|_{X^\alpha} \leq C_t (t - s)^\kappa,$$

and

$$\|f(u(t), \nabla u(t)) - f(u(s), \nabla u(s))\|_{X^\beta} \leq L_R C_t (t - s)^\kappa,$$

with C_t depending only on $R_{\gamma'}$, K_R, T, λ_1, $\alpha - \beta$, κ and the constants of Assumption 2.3.

The bound holds for all $0 \leq s < t \leq T$, including $s = 0$, and the proof is given in Appendix A.1.

The derived regularity from Lemma 2.7 only assumed $u_0 \in X^\alpha$ so the additional smoothness comes from parabolic smoothing. Smoother data improves the regularity (up to 1) in time through Lemma 2.10, which is the limit for our temporal quadrature arguments. For analytic semigroups, the weighted bound is the classical regularity of the mild solution [53]. In (iii), we utilize prior works to make it a modulus of continuity for the force term in order to improve the temporal quadrature rates from Section 4.

We will need to estimate the error incurred when projecting the continuous solution into a finite-dimensional space. One projection is onto the subspace spanned by the first N eigenfunctions of the linear operator L via a spectral Galerkin approximation [51].

Definition 2.11 (Galerkin projection associated with L)**.** *Let H, L, $\{e_k\}_{k=1}^{\infty}$, and $\{\lambda_k\}_{k=1}^{\infty}$ be defined from Definition 2.1. For $N \in \mathbb{N}$, define the finite-dimensional spectral subspace $E_N := span\{e_1, \dots, e_N\} \subset H$, and let $P_N : H \to E_N$ be the orthogonal projection defined as*

$$P_N u := \sum_{k=1}^{N} \langle u, e_k \rangle_H \, e_k.$$

Equivalently, if we define $\mathcal{C}_N : H \to \mathbb{R}^N$, $\mathcal{C}_N u := (\langle u, e_1 \rangle_H, \dots, \langle u, e_N \rangle_H)$, and $\mathcal{S}_N : \mathbb{R}^N \to E_N$, $\mathcal{S}_N a := \sum_{k=1}^{N} a_k e_k$, then the projection is $P_N = \mathcal{S}_N \circ \mathcal{C}_N$.

For $s \geq 0$, the restriction of the projection P_N on X^s is a map of X^s into $E_N \subset X^s$. For $u \in X^s$, the Galerkin approximation is defined by $u_N := P_N u$. When the gradient is well-defined, we define the gradient of the Galerkin approximation by $\nabla u_N := \nabla(P_N u)$.

The projection P_N is contractive on X^s since $\|P_N u\|_{X^s} \leq \|u\|_{X^s}$.

Proposition 2.12 (Galerkin projection error)**.** *If $\alpha > \beta \geq 0$ and P_N is the Galerkin projection, then for every $u \in X^\alpha$,*

$$\|u - P_N u\|_{X^\beta} \leq \lambda_{N+1}^{-(\alpha-\beta)/2} \|u\|_{X^\alpha}.$$

The proof is given in Appendix A.1.

Let $\Lambda_N := \operatorname{diag}(\lambda_1, \dots, \lambda_N)$, then the Galerkin mild map is a map on the time-dependent coefficient in the Galerkin space, i.e. $a(t) \in \mathbb{R}^N$, defined by

$$\Psi_N[a](t) := e^{-t\Lambda_N} \mathcal{C}_N u_0 + \int_0^t e^{-(t-s)\Lambda_N} \mathcal{C}_N f\left(\mathcal{S}_N a(s), \nabla \mathcal{S}_N a(s)\right) ds.$$

Through direct computation, we see that $S(t)\mathcal{S}_N = \mathcal{S}_N e^{-t\Lambda_N}$ and $\mathcal{C}_N \mathcal{S}_N = I$, and thus using $\mathcal{S}_N$, the Galerkin map is equivalent to the E_N-valued map

$$\Phi_N[u](t) = S(t) P_N u_0 + \int_0^t S(t-s) P_N f(P_N u(s), \nabla P_N u(s)) \, ds$$

with $\Phi_N[\mathcal{S}_N a] = \mathcal{S}_N \Psi_N[a]$. Instead of bounding the difference $\Phi_N - \Phi$ separately, we use the Ψ_N operator. In Theorem 4.22, we analyze the difference between the exact mild map and the fully discrete one, the spectral truncation is used in Proposition 2.12 and Lemma 5.3.

3 Neural Network and DAG Approximation

One part of the approximability of the solution operator comes from the structure of the constitutive law through f. In this section, we show that the smoothness and low-dimensionality of f allow for the construction of low parameter count deep ReLU approximations for the solution operator. The DAG construction in this section is compositional, akin to [68, 64], but uses the computation graph of the governing law. In [11], compositions of linear maps and low-dimensional nonlinearities was used, where the rates were determined by Lipschitz constants rather than by the smoothness of the nodes. For component-wise nodes, $p_j = 1/s_j$ is the ratio of input dimension to smoothness that governs the rates for compositions in [68]. The difference in this section is that multilinear nodes only cost $\log \varepsilon^{-1}$, and we use the infimum over all representations of the law, i.e. the invariant p_f

3.1 Deep ReLU Networks

The main quantity we would like to measure is the complexity of approximating the solution operator using a Deep ReLU network. The main nonlinearity approximations in our constructive arguments come from f where we utilize the ReLU activation, dense affine layers, and 1×1 convolutional layers.

Definition 3.1 (Neural Network Components). *Let $\sigma(y) := \max\{y, 0\}$ denote the ReLU activation, applied element-wise.*

A fully connected layer with input dimension n and output dimension n' is defined as the affine map $\mathcal{A}_{A,b} : \mathbb{R}^n \to \mathbb{R}^{n'}$ given by

$$\mathcal{A}_{A,b}(z) := Az + b, \qquad A \in \mathbb{R}^{n' \times n},\ b \in \mathbb{R}^{n'}.$$

Its trainable parameters are the components A and b with a total of $\|A\|_0 + \|b\|_0$ nonzero weights.

For spatial discretizations, we define a variable on the grid that has c channels as

$$X = (X_1, \dots, X_M) \in (\mathbb{R}^c)^M$$

where $X_i \in \mathbb{R}^c$ is the state evaluated at grid point x_i. A 1×1 convolution layer with c input and c' output channels is given by the map

$$\mathcal{K}_{A,b} : (\mathbb{R}^c)^M \to \left(\mathbb{R}^{c'}\right)^M, \quad (\mathcal{K}_{A,b}X)_i := AX_i + b \quad \text{for every } 1 \le i \le M, \tag{4}$$

where the parameters $A \in \mathbb{R}^{c' \times c}$ and $b \in \mathbb{R}^{c'}$ are shared for each i. The parameter count is that of a single dense layer $\mathcal{A}_{A,b}$ on $\mathbb{R}^c$, i.e. $\|A\|_0 + \|b\|_0$ total parameters independent of M. If the convolution is written as a single affine layer on $(\mathbb{R}^c)^M$, with matrix $I_M \otimes A$ and bias $\mathbf{1}_M \otimes b$, it would have $M(\|A\|_0 + \|b\|_0)$ nonzero weights.

Next, we define the full network class considered, which is used in Theorem 3.7 to count the total parameter complexity.

Definition 3.2 (Neural Network Class). *Set the depth $L_0 \in \mathbb{N}$, the input dimension n_0, the output dimension n_{L_0}, and the intermediate widths $n_1, \dots, n_{L_0-1} \in \mathbb{N}$. We define a local ReLU neural network $\mathcal{N} : \mathbb{R}^{n_0} \to \mathbb{R}^{n_{L_0}}$, which maps z to $\widehat{z}^{(L_0)}$ by the following three steps.*

1. (Input layer.) $\widehat{z}^{(1)} = \left(\widehat{z}_1^{(1)}, \dots, \widehat{z}_{n_1}^{(1)}\right)$ *is given by*

$$\widehat{z}^{(1)} := \sigma\left(A_1 z + b_1\right),$$

where $A_1 \in \mathbb{R}^{n_1 \times n_0}$ and $b_1 \in \mathbb{R}^{n_1}$.

2. (Hidden layers.) *For $1 \le l \le L_0 - 2$, the arrays $\widehat{z}^{(l+1)} = \left(\widehat{z}_1^{(l+1)}, \dots, \widehat{z}_{n_{l+1}}^{(l+1)}\right)$ are given recursively by*

$$\widehat{z}^{(l+1)} := \sigma\left(A_{l+1}\widehat{z}^{(l)} + b_{l+1}\right),$$

where $A_{l+1} \in \mathbb{R}^{n_{l+1} \times n_l}$ and $b_{l+1} \in \mathbb{R}^{n_{l+1}}$.

3. (Output layer.) *The output is an affine map given by*

$$\widehat{z}^{(L_0)} := A_{L_0}\widehat{z}^{(L_0-1)} + b_{L_0}$$

where $A_{L_0} \in \mathbb{R}^{n_{L_0} \times n_{L_0-1}}$ and $b_{L_0} \in \mathbb{R}^{n_{L_0}}$.

Altogether the Deep ReLU network is:

$$\mathcal{N}(z) = A_{L_0}\,\sigma\left(A_{L_0-1}\,\sigma\left(\cdots\sigma(A_1 z + b_1)\cdots\right) + b_{L_0-1}\right) + b_{L_0}. \tag{5}$$

The weight matrices A_l and the bias vectors b_l together form the parameter vector $\{(A_l, b_l)\}_{l=1}^{L_0}$. We measure the size by the following:

$$\mathrm{dep}(\mathcal{N}) := L_0, \qquad \mathrm{wid}(\mathcal{N}) := \max_{1 \le l < L_0} n_l, \qquad H(\mathcal{N}) := \sum_{l=1}^{L_0-1} n_l, \qquad W(\mathcal{N}) := \sum_{l=1}^{L_0} \left(\|A_l\|_0 + \|b_l\|_0\right),$$

referred to as the depth, the width, the number of neurons, and the number of nonzero weights and biases.

The network class is defined as follows. For a compact set $K \subset \mathbb{R}^{n_0}$ and constants $L_0, p, W, H \in \mathbb{N}$ and $\Theta, R > 0$, we define the class by:

$$\mathcal{F}_{n_0,n_{L_0}}^{L_0,H,\sigma}(p,W,\Theta,R;K) := \left\{ \mathcal{N} \text{ as in (5)} : \begin{array}{l} \operatorname{dep}(\mathcal{N}) \le L_0,\ \operatorname{wid}(\mathcal{N}) \le p,\ H(\mathcal{N}) \le H,\ W(\mathcal{N}) \le W, \\ \max_l \|A_l\|_\infty,\ \max_l \|b_l\|_\infty \le \Theta,\ \|\mathcal{N}\|_{L^\infty(K)} \le R \end{array} \right\}$$

which are deep ReLU networks with depth at most L_0, at most H neurons, sparsity at most W, parameter magnitude at most Θ, and max-norm bounded by R on K.

For grid-based functions, we replace each dense layer $z \mapsto A_l z + b_l$ in the three steps above by the point-wise convolutional layer $\mathcal{K}_{A_l,b_l}$ of (4), with no change to the activation, so that map $\mathcal{N}^h : (\mathbb{R}^{n_0})^M \to \left(\mathbb{R}^{n_{L_0}}\right)^M$ has the same depth, with the same parameter vector $\{(A_l, b_l)\}_{l=1}^{L_0}$.

A layer is called fixed if its weights are pre-determined and trainable otherwise.

We will utilize several properties of the ReLU networks, used throughout the analysis [7, 63]. We summarize them in the lemma below.

Lemma 3.3 (Operations on ReLU networks)**.** *Let $\mathcal{N}_1, \mathcal{N}_2$ be two ReLU networks from Definition 3.2 with compatible dimensions. Each operation produces a new parameter vector $\{(A_l, b_l)\}$, and by Lemma 3.4 each statement holds directly for grid realizations, with W multiplied by the number of points and the parameter count unchanged.*

1. (Affine layers.) *A map $z \mapsto Az + b$ is a network of depth 1 with $\|A\|_0 + \|b\|_0$ weights. Composing two networks with an affine map does not increase the depth.*
2. (Composition.) *The network $\mathcal{N}_2 \circ \mathcal{N}_1$ has* $\operatorname{dep} \le \operatorname{dep}(\mathcal{N}_1) + \operatorname{dep}(\mathcal{N}_2)$ *and* $W \le 2W(\mathcal{N}_1) + 2W(\mathcal{N}_2)$.
3. (Parallelization.) *The map $z \mapsto (\mathcal{N}_1(z), \mathcal{N}_2(z))$ is a network with* $\operatorname{dep} = \max_i \operatorname{dep}(\mathcal{N}_i)$. *If the two depths are the same,* $\operatorname{wid} \le \sum_i \operatorname{wid}(\mathcal{N}_i)$ *and* $W \le \sum_i W(\mathcal{N}_i)$. *If the depths differ, then we extend the shallower network $\mathcal{N}_i$ by padding with $\Delta := |\operatorname{dep}(\mathcal{N}_1) - \operatorname{dep}(\mathcal{N}_2)|$ identity layers which adds at most $W(\mathcal{N}_i) + 2m_i\Delta$ weights and $2m_i$ in width, where m_i is the output dimension of $\mathcal{N}_i$.*
4. (Identity and Skip Connections.) *The identity on $\mathbb{R}^k$ can be realized exactly by $y = \sigma(y) - \sigma(-y)$, with $4k$ nonzero weights per layer. Mapping a vector of dimension k through l layers has a total cost of at most $4kl$ weights and no added depth.*
5. (Clipping.) *For threshold $\mathrm{a} > 0$, the clipping operation* $\operatorname{Clip}_{\mathrm{a}}(y) = \min\{\max(y, -\mathrm{a}), \mathrm{a}\} = \sigma(y + \mathrm{a}) - \sigma(y - \mathrm{a}) - \mathrm{a}$ *can be realized exactly by one hidden layer that has 7 nonzero weights and biases per coordinate (4 hidden, 3 output).*

The proof is given in Appendix A.2.

Lemma 3.4 (Parameter Sharing)**.** *Let $g : \mathbb{R}^c \to \mathbb{R}^{c'}$ be a ReLU network as in Definition 3.2, with parameters $\{(A_l, b_l)\}_{l=1}^{L_0}$, and let*

$$\left(G^h(X)\right)_i := g(X_i), \qquad i = 1, \dots, M,$$

be its point-wise application on a grid. Then G^h is the grid realization g^h from Definition 3.2, obtained by replacing each dense layer $\mathcal{A}_{A_l,b_l}$ by the point-wise convolution $\mathcal{K}_{A_l,b_l}$ of (4) without changing the activation function. Thus

$$\operatorname{dep}(G^h) = \operatorname{dep}(g),\ \ \#\{\text{distinct parameters of } G^h\} = W(g),\ \ W(G^h) = M\,W(g),$$

where $W(G^h)$ counts the nonzero weights of G^h written without parameter sharing, with block-diagonal matrices $I_M \otimes A_l$ and biases $\mathbf{1}_M \otimes b_l$.

The grid-based saving comes from considering it as a convolution layer, $W(g)$ total shared weights, instead of a dense ReLU layer. The proof is a direct computation and given in Appendix A.2.

3.2 The DAG formulation

We will form the approximation of f by first writing it as a directed acyclic graph (DAG) and show that the DAG can be approximated by a deep ReLU network. This allows us to probe how the complexity depends on the actual structure of f.

Assumption 3.5 (Local constitutive map with finite DAG representation)**.** *Let $R > 0$ and let*

$$B_R^\alpha := \left\{ u \in X_T^\alpha : \|u\|_{X_T^\alpha} \le R \right\}$$

be the ball from Proposition 2.5. Under Assumption 2.2, the set of admissible point-wise states

$$K := \overline{\left\{ (u(t,x), \nabla u(t,x)) \ : \ u \in B_R^\alpha,\ t \in [0,T],\ x \in \overline{\Omega} \right\}} \ \subset \ \mathbb{R}^d \times \mathbb{R}^{d \times d_x}$$

is bounded, since $\|u(t,\cdot)\|_{C^1(\overline{\Omega})} \le C_{\mathrm{emb}} R$ for every $u \in B_R^\alpha$ and $t \in [0,T]$, and therefore K is compact.

Let $U \subset \mathbb{R}^d \times \mathbb{R}^{d \times d_x}$ be an open neighborhood of K and let $f : U \to \mathbb{R}^m$ be the law from the PDE (2), where $m = d$ for the law itself and $m \ge d$ for a local part f_{loc} as from Lemma 5.17. Assume that f admits a representation by a single finite directed acyclic graph $G = (\mathcal{V}, \mathcal{E})$, whose input nodes are the coordinates of $z = (u, \nabla u)$ and whose (unique) output node j_{out} produces the vector $f(z) \in \mathbb{R}^m$, and where each node takes one of the following three forms.

1. Affine maps: $A : \mathbb{R}^n \to \mathbb{R}^p$, $A(x) = Mx + b$, *with $M \in \mathbb{R}^{p \times n}$ and $b \in \mathbb{R}^p$.*
2. Multilinear maps: $T : (\mathbb{R}^n)^r \to \mathbb{R}^p$ *for some $r \ge 2$, linear in each argument separately.*
3. Component-wise nonlinearities: $G : \mathbb{R}^n \to \mathbb{R}^n$ *with $G(x_1, \dots, x_n) = (g(x_1), \dots, g(x_n))$, where $g \in C^s(I)$ for some finite integer $s \ge 1$ and some open bounded interval $I \subset \mathbb{R}$.*

For every j that is not an input node, let K_j denote the range of the intermediate variable at node j over all inputs $z \in K$. Denote the set of all parent nodes of j by $\mathrm{pa}(j)$, and define

$$K_j^{\mathrm{in}} := \prod_{i \in \mathrm{pa}(j)} K_i.$$

Assume that there exists an open set U_j contained in the domain of the corresponding node map with $K_j^{\mathrm{in}} \Subset U_j$, where $\Subset$ means compact containment. Let $\mathcal{V}_{\mathrm{in}}$ be the set of input nodes and let $\mathcal{V}_{\mathrm{nl}}$ be the set of non-affine (multilinear or component-wise) nodes, where we assume $\mathcal{V}_{\mathrm{nl}} \neq \emptyset$.

For the main results, the exponents will depend on the DAG formulation, which is not unique for a given f. For example, the cubic map $f(u) = u^3$ can be written as two multilinear nodes, as one component-wise C^∞ node, or as one component-wise C^3 node. Each of these DAGs are valid under Assumption 3.5, but they lead to different rates. To avoid this issue, we define the following admissible representation.

Definition 3.6 (Admissible Representations and the Structural Exponent)**.** *An admissible representation of f is a finite DAG G satisfying the conditions of the assumption on K. For all admissible G, we define*

$$p_{\max}(G) := \max_{j \in \mathcal{V}_{\mathrm{nl}}(G)} p_j, \qquad p_j := \begin{cases} 0, & j \text{ multilinear}, \\ 1/s_j, & j \text{ component-wise } C^{s_j}, \end{cases}$$

where $p_{\max}(G) := 0$ if G has no component-wise node.

The structural exponent of f (on K) is defined by

$$p_f := \inf \left\{ p_{\max}(G) \ : \ G \text{ an admissible representation of } f \right\}.$$

The infimum defines a unique exponent that reveals the structure of f directly. If f is a polynomial map, then $p_f = 0$ and the infimum is attained by the DAG using each monomial as a tree of multilinear nodes. If f is a component-wise map $x \mapsto (g(x_1), \dots, g(x_n))$ where $g \in C^s$, then $p_f \le 1/s$ and if $g \in C^\infty$ then $p_f = 0$.

Theorem 5.6 uses a given representation, i.e. for $p_{\max}(G)$. Note that since it holds for every admissible G and since the bound is monotone in the exponent, it will hold when $p_{\max}$ is replaced by any $p > p_f$. If the

infimum is attained by the graph, then $p_{\max}$ can be replaced by $p = p_f$. The difference is that $p_{\max}$ refers to a particular representation used in the proofs while the main structural results will use the best one obtained through p_f. For the explicit examples in Section 8, we can obtain the infimum for all but the Arrhenius case (Example 8.2), since $p_f = 0$ is approached but not attained. Figure 2 illustrates the computation graphs for two of the examples of Section 8.

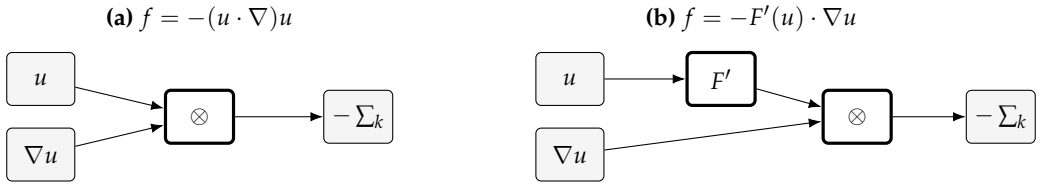


Figure 2: Computation Graphs for Two Constitutive Equations. The bold boxes are the nonlinear nodes, the shaded boxes are affine and realized exactly. In (a) the only nonlinear node is multilinear, yielding $(p, q) = (0, 1)$ and a network of size $O(\log \varepsilon^{-1})$. In (b) the node $F' \in C^{s-1}$ leads to $(p, q) = (1/(s-1), 1)$.

The next theorem constructs a deep ReLU network that approximates a DAG map node-wise. To avoid the trivial case, we assume that there is at least one nonlinear node.

Theorem 3.7 (Deep ReLU Approximation for Finite DAG)**.** *Let f have the structure from Assumption 3.5. If the computation graph contains no nonlinear nodes, then f is affine and is exactly representable by a ReLU network of depth and size bounded independently of ε.*

If f's DAG has at least one nonlinear node, then there exist constants $\varepsilon_0 \in (0, 1)$ and $C > 0$, depending only on the computation graph, the compact set K, the node dimensions, and the node maps, such that for every $\varepsilon \in (0, \varepsilon_0]$ there exists a ReLU network $\mathcal{N}_\varepsilon : \mathbb{R}^d \times \mathbb{R}^{d \times d_x} \to \mathbb{R}^m$ satisfying

$$\sup_{z \in K} \|f(z) - \mathcal{N}_\varepsilon(z)\|_{\mathbb{R}^m} \leq \varepsilon.$$

Additionally, there exist constants $c_j > 0$, $p_j \geq 0$, and $q_j \geq 0$, indexed by the nonlinear nodes $j \in \mathcal{V}_{\text{nl}}$, such that

$$\operatorname{dep}(\mathcal{N}_\varepsilon) \leq \sum_{j \in \mathcal{V}_{\text{nl}}} c_j \log(\varepsilon^{-1}), \qquad W(\mathcal{N}_\varepsilon) \leq \sum_{j \in \mathcal{V}_{\text{nl}}} c_j \, \varepsilon^{-p_j} \left(\log(\varepsilon^{-1})\right)^{q_j},$$

where $(p_j, q_j) = (0, 1)$ for multilinear nodes and $(p_j, q_j) = (1/s_j, 1)$ for nodes j induced by a scalar function $g_j \in C^{s_j}$.

The proof approximates each nonlinear node by Theorem B.1 or Theorem B.3, and propagates the node-wise errors along the topological ordering of the graph. One subtlety in the proof is the use of a buffer around the input set for each node so that perturbations remain within a region where the operations are Lipschitz. The detailed analysis is given in Appendix A.2.

Remark 3.8 (Dependence on d and d_x)**.** *The exponents p_j in Theorem 3.7 are dimension-free. Only the coefficients c_j depend on the node dimensions and thus are dependent on the input dimension. A component-wise node on $\mathbb{R}^n$ is just n copies of one scalar network, so $c_j = O(n)$. By Theorem B.3, multilinear node $T : (\mathbb{R}^n)^r \to \mathbb{R}^p$ costs $O(\#\{\text{nonzero coefficients}\}) \leq O(p\, n^r)$.*

For standard examples, the cost is $C\, d\, d_x \log \varepsilon^{-1}$ for the term $-(u \cdot \nabla)u$, $C\, d \log \varepsilon^{-1}$ for a polynomial reaction term, and $C\, d\, d_x \varepsilon^{-1/(s-1)} \log \varepsilon^{-1}$ for flux terms $-F'(u) \cdot \nabla u$ with $F \in C^s$. They are all linear in the dimension. If we applied a generic result for C^s map $\mathbb{R}^{d+dd_x} \to \mathbb{R}^d$, the expected rate would be $\varepsilon^{-(d+dd_x)/s}$. So informally, we say that the cost or complexity is d_x-independent if the exponents do not directly depend on the dimension.

Several important physical systems have constitutive equations that include nonlocal effects, i.e., they are in the form $\mathcal{M} f_{\text{loc}}(u, \nabla u)$ where f_{loc} is local and $\mathcal{M}$ is a bounded nonlocal linear operator. For example, for the incompressible Navier-Stokes equation the operator is the Leray projection or the Biot-Savart operator. This additional cost is proportional to the number of spectral weights per mode, where we show in Lemma 5.17 of Section 5.6 that it can be absorbed into the spectral layers. Note also that finite characterization of the governing PDE was also used in the context of [67].

4 Discretization and the Neural Picard Approximation

In this section, we construct a discrete mild map and bound the error over each step of discretization. The discretization is a theoretical tool rather than a numerical method. We utilize Galerkin theory [78, 51, 77] as the approximating technique. We would like to show the existence of such networks with uniform bounds, and thus utilize an approximation that can be taken around the continuous solution u. We first replace the Duhamel operator by a quadrature in time and then the inner products by a spatial quadrature which defines the fully discrete neural Picard map. We show that the maps are contractive and consistent (in the numerical analysis sense). Section 5 combines these results to bound the total error incurred over multiple neural Picard steps.

4.1 Temporal Discretization

As a function of time, forcing along the trajectory is

$$g(t) = f\left(u(t), \nabla u(t)\right) \in L^\infty\left(0, T; X^\beta\right),$$

bounded by F_R, and by Lemma 2.7 Part (3), it is Hölder continuous in the weighted norm. These are the assumptions used in the next results, and only require that the data is in X^α.

Proposition 4.1 (Temporal Consistency)**.** *Let $t_q := q\Delta t$, $q = 0, \dots, Q$, be the nodes with time step $\Delta t = T/Q$, let $0 \le \beta < \alpha$ with $\alpha - \beta < 2$, and set $\sigma := \frac{\alpha-\beta}{2}$ and $\theta := 1 - \sigma > 0$. Let $\zeta \in (0, \theta)$ and assume that $g : [0, T] \to X^\beta$ satisfies*

$$\|g\|_{\infty,\beta} := \sup_{0\le t\le T} \|g(t)\|_{X^\beta} < \infty, \qquad [g]_\zeta := \sup_{0<s<t\le T} \frac{\|g(t) - g(s)\|_{X^\beta}}{(t-s)^\zeta s^{-\zeta}} < \infty.$$

If the discrete Duhamel operator is given by

$$(\mathcal{D}_Q[g])^q := \Delta t \sum_{r=0}^{q-1} S(t_q - t_r) g(t_r),$$

then there is a constant $C > 0$, depending only on σ and ζ and independent of Δt, T and Q, such that

$$\max_{0\le q\le Q} \left\|\mathcal{D}[g](t_q) - (\mathcal{D}_Q[g])^q\right\|_{X^\alpha} \le C\,\|g\|_{\infty,\beta}\,(\Delta t)^\theta + C\,[g]_\zeta\,T^{\theta-\zeta}\,(\Delta t)^\zeta.$$

The consistency is $\mathcal{O}((\Delta t)^\zeta)$ since $\zeta < \theta$.

In addition, if for some $\kappa \in (0, 1]$ and some $\beta' \ge \beta$ with $\kappa < \theta + \frac{\beta'-\beta}{2}$, we had

$$\sup_{0\le t\le T} \|g(t)\|_{X^{\beta'}} \le G, \qquad \|g(t) - g(s)\|_{X^\beta} \le H\,(t-s)^\kappa \quad (0 \le s < t \le T),$$

then, with $a := \left(\sigma - \frac{\beta'-\beta}{2} + \kappa\right)_+ < 1$ and C depending only on σ, κ and $1 - a$, and λ_1 (when $\beta' - 2\kappa > \alpha$ due to embedding), we have

$$\max_{0\le q\le Q} \left\|\mathcal{D}[g](t_q) - (\mathcal{D}_Q[g])^q\right\|_{X^\alpha} \;\le\; C\left(T^\theta H + T^{1-a} G\right)(\Delta t)^\kappa.$$

The proof, given in Appendix A.3, splits the error into two parts. One freezes the kernel on each subinterval, which needs only boundedness of g and gives the term $\|g\|_{\infty,\beta}(\Delta t)^\theta$, and the second freezes g which gives the term $[g]_\zeta(\Delta t)^\zeta$. The order of accuracy is determined by the time regularity of the forcing term, see Remark 4.2.

Note that the time discretization is a version of the Lawson-Euler method [44, 32], which uses the exact semigroup (implicitly) and is explicit in forcing term. We use this as opposed to exponential Euler [14] since it will allow for extension by energy estimates in Section 5.3 using just a one-sided growth bound on the governing law.

Remark 4.2. *Any γ with $\alpha < \gamma < \beta + 2$ is admissible in Lemma 2.7. As γ approaches the upper limit $\beta + 2$, $\zeta \to \theta$ and $\chi \to 1 + \frac{(\alpha-2)_+}{2\theta}$. This is the same choice that improves the spectral requirement and the limit in (7), so no exponent is needed to trade off. The additional theoretical cost is the factor $(1 - \bar{\sigma})^{-1} = (\theta - \zeta)^{-1}$, which then appears in later constant, so we cannot set it to θ. If the force is smoother then the bootstrapping result in Lemma 2.10 helps.*

4.2 Spatial Discretization

Consider a discretization Ω_h of the spatial domain Ω, and define the grid spacing by

$$h := \max_{\Omega' \in \Omega_h} \operatorname{diam}(\Omega'),$$

i.e., the maximum diameter of a subdomain of the mesh. Let $\{x_i\}_{i=1}^M \subset \Omega$ denote the sampling points and let $\{\omega_i\}_{i=1}^M$ denote the weights associated with the spatial quadrature rule.

Definition 4.3 (Discrete Galerkin operators)**.** *Let $E_N := \operatorname{span}\{e_1, \dots, e_N\}$, and recall the continuous (Galerkin) coefficient and synthesis operators*

$$\mathcal{C}_N : H \to \mathbb{R}^N, \qquad (\mathcal{C}_N v)_k := \langle v, e_k \rangle_H,$$

and

$$\mathcal{S}_N : \mathbb{R}^N \to E_N, \qquad \mathcal{S}_N a := \sum_{k=1}^N a_k e_k,$$

so that $P_N = \mathcal{S}_N \mathcal{C}_N$ is the spectral Galerkin projection onto E_N. Define the sampling operator $\mathcal{I}_h : C(\overline{\Omega}) \to \mathbb{R}^M$ by $(\mathcal{I}_h v)_i := v(x_i)$, for $i = 1, \dots, M$ and is defined for every continuous function. For $a \in \mathbb{R}^N$, the sampled Galerkin function is defined as

$$\mathcal{S}_N^h a := \mathcal{I}_h \mathcal{S}_N a \in \mathbb{R}^M, \qquad (\mathcal{S}_N^h a)_i = \sum_{k=1}^N a_k e_k(x_i).$$

So $\mathcal{S}_N a \in E_N$ denotes the continuous Galerkin function associated with a, while $\mathcal{S}_N^h a$ denotes its values on the spatial grid.

For a grid function $v = (v_1, \dots, v_M)^T \in \mathbb{R}^M$, define the discrete coefficient map $\mathcal{C}_N^h : \mathbb{R}^M \to \mathbb{R}^N$ by

$$(\mathcal{C}_N^h v)_k := \sum_{i=1}^M \omega_i v_i e_k(x_i), \qquad k = 1, \dots, N,$$

which approximates the continuous coefficient map $\mathcal{C}_N$ by replacing the spatial inner products with a quadrature rule. The lifted discrete Galerkin operator

$$\mathcal{P}_N^h := \mathcal{S}_N \mathcal{C}_N^h : \mathbb{R}^M \to E_N$$

maps grid values to a function in the continuous Galerkin space E_N. Its sampled version is

$$P_N^h := \mathcal{I}_h \mathcal{P}_N^h = \mathcal{S}_N^h \mathcal{C}_N^h : \mathbb{R}^M \to \mathbb{R}^M.$$

The Galerkin semigroup acting on coefficient vectors is defined as the truncated operator

$$S_N(t) := e^{-t\Lambda_N}, \qquad \Lambda_N := \operatorname{diag}(\lambda_1, \dots, \lambda_N).$$

Since $S(t)e_k = e^{-\lambda_k t} e_k$, the continuous synthesis operator satisfies $S(t)\mathcal{S}_N = \mathcal{S}_N S_N(t)$. For every $s \geq 0$,

$$\|\mathcal{S}_N a\|_{X^s} = \|a\|_{\ell^2_s}, \qquad \|a\|_{\ell^2_s} := \left(\sum_{k=1}^N \lambda_k^s |a_k|^2 \right)^{1/2},$$

and thus $\mathcal{S}_N$ is an isometry from $(\mathbb{R}^N, \|\cdot\|_{\ell^2_s})$ onto E_N equipped with the X^s norm, uniformly in N.

Define the sampled gradient operator $\mathcal{I}_h^{grad} : E_N \to (\mathbb{R}^{d\times d_x})^M$ *by*

$$(\mathcal{I}_h^{grad} v)_i := \nabla v(x_i), \qquad i = 1, \dots, M,$$

and for $a \in \mathbb{R}^N$ *define*

$$\nabla_h \mathcal{S}_N^h a := \mathcal{I}_h^{grad} \mathcal{S}_N a = \left(\sum_{k=1}^{N} a_k \nabla e_k(x_i) \right)_{i=1}^{M}.$$

Both $\mathcal{S}_N^h a$ and $\nabla_h \mathcal{S}_N^h a$ are obtained by first constructing the continuous Galerkin function $\mathcal{S}_N a$ and then sampling the function and its gradient, respectively, at the spatial grid points. When u is vector-valued, say on $\mathbb{R}^d$ with $d > 1$, every operator above acts component-wise and the coefficient space becomes $(\mathbb{R}^N)^d$. To keep the notation simple, we write $\mathbb{R}^N$ and $\mathbb{R}^M$, and only use $\mathbb{R}^{dN}$ and $\mathbb{R}^{dM}$ when necessary for examples.

In general, P_N^h is not a projection, since the quadrature rule does not need to satisfy $\mathcal{C}_N^h \mathcal{S}_N^h = I_{\mathbb{R}^N}$. The composition is

$$(\mathcal{C}_N^h \mathcal{S}_N^h)_{jk} = \sum_{i=1}^{M} \omega_i e_j(x_i) e_k(x_i),$$

and is equal to $I_{\mathbb{R}^N}$ only when the quadrature rule is exact on the products $e_j e_k$, $1 \le j, k \le N$.

To discretize the Picard map, the input to f uses the reconstructed Galerkin function so the approximation is in the spatial dimension, not the discretization dimension. This is akin to a pseudo-spectral approximation and thus for $a \in \mathbb{R}^N$,

$$f\left(\mathcal{S}_N^h a,\ \nabla_h \mathcal{S}_N^h a\right)$$

denotes the evaluation of the nonlinearity on the reconstructed function and its discrete gradient at the spatial sample points. The gradient $\nabla_h \mathcal{S}_N^h a$ is obtained by differentiating the basis elements and evaluating the result on the same grid.

Definition 4.4 (Fully Discrete Neural Picard Map)**.** *Suppose we use the temporal quadrature rule from Proposition 4.1, the spatial discretization and the discrete operators* $\mathcal{C}_N^h$, $\mathcal{S}_N^h$, S_N, *and* $\nabla_h \mathcal{S}_N^h$ *from Definition 4.3. Let*

$$\left(\mathcal{F}^h(U,P)\right)_i := f(U_i, P_i), \qquad \left(\widehat{\mathcal{F}}_\varepsilon^h(U,P)\right)_i := \widehat{f}_\varepsilon(U_i, P_i), \qquad i = 1, \dots, M,$$

denote the point-wise evaluation of the exact constitutive law and of its ReLU approximation $\widehat{f}_\varepsilon := \mathcal{N}_\varepsilon$ *from Theorem 3.7. The parameters are shared since a single network is applied at every grid point.*

Let $a = (a^0, \dots, a^Q) \in (\mathbb{R}^N)^{Q+1}$ *and let* $a_0^h := \mathcal{C}_N^h \mathcal{I}_h u_0$ *be the discrete initial coefficients. The fully discrete Picard map in coefficient space is defined as*

$$\left(\Psi_{N,Q}^h(a)\right)^q := S_N(t_q)\, a_0^h + \Delta t \sum_{r=0}^{q-1} S_N(t_q - t_r)\, \mathcal{C}_N^h \left[\mathcal{F}^h\left(\mathcal{S}_N^h a^r,\ \nabla_h \mathcal{S}_N^h a^r\right)\right]$$

for $q = 0, \dots, Q$ *and where the sum is zero when* $q = 0$*. The neural approximation is given by*

$$\left(\Psi_{N,Q,\varepsilon}^h(a)\right)^q := S_N(t_q)\, a_0^h + \Delta t \sum_{r=0}^{q-1} S_N(t_q - t_r)\, \mathcal{C}_N^h \left[\widehat{\mathcal{F}}_\varepsilon^h\left(\mathcal{S}_N^h a^r,\ \nabla_h \mathcal{S}_N^h a^r\right)\right].$$

The corresponding E_N*-valued maps are obtained by applying* $\mathcal{S}_N$ *at each time level and interpolating piecewise linearly in time, i.e., for* $u \in X_T^\alpha$ *we write*

$$\left(\Phi_{N,Q,\varepsilon}^h[u]\right)(t_q) := \mathcal{S}_N\left(\Psi_{N,Q,\varepsilon}^h(b(u))\right)^q, \qquad b(u)^r := \mathcal{C}_N^h \mathcal{I}_h u(t_r),$$

and similarly for $\Phi_{N,Q}^h$*. Since* $E_N \subset X^\alpha$*, this defines operators* $\Phi_{N,Q}^h, \Phi_{N,Q,\varepsilon}^h : X_T^\alpha \to X_T^\alpha$*. Lastly, we call a sequence* $a_{N,Q}^h$ *a fully discrete fixed point if* $a_{N,Q}^h = \Psi_{N,Q}^h(a_{N,Q}^h)$*, and similarly in the neural approximation case.*

The difference between $\mathcal{S}_N$ and $\mathcal{S}_N^h$ is grid-evaluation and is only an implementation aspect. Definition 4.4 can be stated as an architecture so that each component of Section 3.1 is clear. The iteration of the Picard maps is equivalent to a parameter shared loop.

Similar to other neural operators, we assume that we have access to fast transforms between grid values and coefficients. These transforms are fixed, while all other weights and layers are learned. The following assumption separates the two cases.

Assumption 4.5 (Fast Transforms)**.** *There are fixed linear maps $\mathcal{T}_{\text{syn}}, \mathcal{T}_{\text{syn},1}, \dots, \mathcal{T}_{\text{syn},d_x}$ and $\mathcal{T}_{\text{an}}$, each a fast transform in the sense of Assumption 5.11, a partition of the retained modes into blocks of at most b_0 modes, and maps $D_{\text{an}}^{\text{ex}}, D_1^{\text{ex}}, \dots, D_{d_x}^{\text{ex}}$ (block-diagonal), such that*

$$\mathcal{S}_N^h = \mathcal{T}_{\text{syn}}, \qquad \partial_{h,j}\mathcal{S}_N^h = \mathcal{T}_{\text{syn},j}\, D_j^{\text{ex}}, \qquad \mathcal{C}_N^h = D_{\text{an}}^{\text{ex}}\, \mathcal{T}_{\text{an}}.$$

As an example, on the torus with the real Fourier basis, $\mathcal{T}_{\text{syn}} = \mathcal{T}_{\text{syn},j}$ is the IFFT and $\mathcal{T}_{\text{an}}$ the FFT. The blocks are the cosine-sine pairs of one frequency, so $b_0 = 2$ and $D_{\text{an}}^{\text{ex}}$ is the normalized quadrature weights, and D_j^{ex} maps each pair to its derivative, a 2×2 block with two nonzero entries. On these uniform grids all quadrature weights are equal to the same value ω (e.g. $\omega = |\Omega|/M$ on the torus). The exact value of $D_{\text{an}}^{\text{ex}}$ is this weight times the L^2-normalization constants of the eigenfunctions, so the quadrature weights are absorbed into the learned spectral layer D_{an} and cost no extra parameters.

Definition 4.6 (Learned Spectral Layers)**.** *The learned spectral layers are maps $D_{\text{an}}, D_1, \dots, D_{d_x}, D_S, D_W$ and, when the law is post-composed with a fixed operator $\mathcal{M}$ that is diagonal in the eigenbasis (Lemma 5.17), $D_{\mathcal{M}}$, each restricted to the sparsity pattern of its exact counterpart*

$$D_{\text{an}}^{\text{ex}}, \qquad D_j^{\text{ex}}, \qquad D_S^{\text{ex}} := S_N(\Delta t), \qquad D_W^{\text{ex}} := \Delta t\, S_N(\Delta t), \qquad D_{\mathcal{M}}^{\text{ex}} := D_\mu.$$

They are used in the neural architecture (Definition 4.7) in place of the exact maps. The spectral weights are the elements of these maps, they are shared across grid points, time levels and sweeps, and their count satisfies

$$W_{\text{spec}} \le\ C_{\text{spec}}\, N, \qquad C_{\text{spec}} \le b_0 \left(d^2(3 + d_x) + d\, m\, \mathbf{1}_{\mathcal{M}} \right),$$

where C_{spec} is the number of nonzero elements per mode of the exact maps.

On the torus, in the component-wise Fourier basis, one can show that $C_{\text{spec}} = d(3 + d_x)$, the $d\, d_x$ derivative weights are removed for $f(u)$ without gradient terms. An additional linear term $\mathcal{M}$ adds a $d \times m$ block per mode, where m is the output dimension of f_{loc}. For the Leray projection $m = d$ and the block is the identity.

Definition 4.7 (The Looped Architecture)**.** *Fix an admissible pair $(N, h) \in \mathcal{A}$ (Assumption 4.16), Q, n, the network $\widehat{f}_\varepsilon$ from Definition 3.2, with $n_0 = d(1 + d_x)$ (or $n_0 = d$ for gradient-free and divergence-form laws) and $n_{L_0} = m$ ($m = d$ unless the law has the additional term $\mathcal{M}$), and learned spectral layers $D_{\text{an}}, D_1, \dots, D_{d_x}, D_S, D_W$, and $D_{\mathcal{M}}$ when needed, from Definition 4.6. The state propagated through the loop is the entire discrete trajectory,*

$$a = (a^0, \dots, a^Q) \in \left(\mathbb{R}^{dN} \right)^{Q+1}.$$

The architecture consists of three parts, where the only fixed layers are the fast transforms from Assumption 4.5.

1. (Encoder) *From the sampled data $w = \mathcal{I}_h u_0 \in \mathbb{R}^{dM}$ compute*

$$a_0^h := D_{\text{an}}\, \mathcal{T}_{\text{an}} w, \qquad A^0 := \left(a_0^h, \dots, a_0^h \right),$$

   which uses the fixed transform, a learned spectral layer, and replication across the $Q + 1$ time levels.

2. (Looping) *One step $a \mapsto \Psi(a)$ is the composition of four layers. For each time level $r = 0, \dots, Q - 1$:*

   (2a) Synthesis: *$U^r := \mathcal{T}_{\text{syn}} a^r \in (\mathbb{R}^d)^M$ and $P_j^r := \mathcal{T}_{\text{syn},j} D_j a^r$ for $j = 1, \dots, d_x$, so that $P^r \in (\mathbb{R}^{d \times d_x})^M$. That is, the learned derivative multipliers act on the coefficients, and the fixed transforms then produce the grid values of the state and its gradient.*

(2b) point-wise block: $G^r := \widehat{\mathcal{F}}_\varepsilon^h(U^r, P^r)$, *that is,* $(G^r)_i = \widehat{f}_\varepsilon\left(U_i^r, P_i^r\right)$ *at every grid point* $i = 1, \dots, M$. *By Lemma 3.4 this layer is the grid realization of* $\widehat{f}_\varepsilon$ *in the sense of Definition 3.2. Each dense layer* $A_l \cdot + b_l$ *of the block acts as the* 1×1 *convolution* $\mathcal{K}_{A_l, b_l}$, *with the same matrices at every point. This is the only nonlinear layer.*

(2c) Analysis: $\widehat{g}^r := D_{\mathcal{M}} D_{\mathrm{an}} \mathcal{T}_{\mathrm{an}} G^r \in \mathbb{R}^{dN}$. *That is, a fixed transform maps the grid values to coefficients, and the learned spectral layers then normalize them. If the law is not post-composed with an operator* $\mathcal{M}$, *then* $D_{\mathcal{M}}$ *is the identity.*

Then, across time levels,

(2d) Collect:

$$v^0 := a_0^h, \qquad v^{q+1} := D_S\, v^q + D_W\, \widehat{g}^q \quad (q = 0, \dots, Q-1), \qquad (\Psi(a))^q := v^q.$$

This recurrence equation is linear, and is diagonal in the eigenbasis and causal in time. The learned parts are the decay factors and Duhamel weights as coefficients. Since these coefficients are the same at every step, it can be realized in $O(\log Q)$ *depth by a parallel scan [5] (Theorem 5.13).*

The loop is then $A^k := \Psi(A^{k-1})$ *for* $k = 1, \dots, n$, *the layers (2a)-(2d) are the same at every sweep.*

3. (Output:) *The output* A^n *gives coefficients at the time nodes* t_q. *Between nodes, the trajectory is reconstructed by the semigroup lifting in Definition 4.12, which applies* $S_N(\tau)$ *for* $0 < \tau < \Delta t$. *The architecture has no separate layer for* $S_N(\tau)$, *specifically, it uses* $D_S^{\tau/\Delta t}$ *instead (powers applied element-wise). At the exact value* $D_S = S_N(\Delta t) = e^{-\Delta t \Lambda_N}$ *this equals* $e^{-\tau \Lambda_N} = S_N(\tau)$. *It is well defined whenever* D_S *has positive diagonal components and it does not add parameters.*

By Lemma 3.4, applied to the MQ space-time points of one sweep, the block $\widehat{f}_\varepsilon$ is constructed by $W(\widehat{f}_\varepsilon)$ parameters and contributes $MQ\, W(\widehat{f}_\varepsilon)$ nonzero weights per sweep, thus a cost of $nMQ\, W(\widehat{f}_\varepsilon)$ over n sweeps.

Remark 4.8 (The Loop as an FNO)**.** *This architecture includes as a special case the FNO. On the torus, using the Fourier basis and a uniform grid, Definition 4.7 is an FNO with parameter sharing. First, one can arrange the* $Q+1$ *time levels as channels. The synthesis, derivative, and analysis layers are just Fourier multipliers. Collecting (2d) is a channel-mixing multiplier (done mode-wise) that is lower triangular in the time index. The point-wise block is a sequence of Fourier layers with zero Fourier part applied block-diagonally on the channels. The result is an FNO with a channel width that is proportional to* Q *and depth* $O(n \operatorname{dep}(\widehat{f}_\varepsilon))$. *Every sweep uses the same layers with the same weights. The bounds from Theorems 5.6 and 5.14 apply to this class of Fourier neural operators, with the trainable parameter count* $W(\widehat{f}_\varepsilon) + W_{\mathrm{spec}}$. *In practice, FNOs share weights across grid points and use an autoregressive rollout in time. Without sharing across time levels, the trainable count would grow by a factor* $Q \asymp \eta^{-1/\zeta_t}$. *Without sharing parameters across the Picard sweeps, an extra factor* $n \asymp \log \eta^{-1}$ *is needed.*

Remark 4.9 (The Loop as an Encoder-Decoder Operator)**.** *The architecture of Definition 4.7 can also be viewed as an encoder-decoder operator network, e.g. Chen-Chen, PCA-Net, or DeepONet type. The encoder is just the linear map* $w \mapsto a_0^h = D_{\mathrm{an}} \mathcal{T}_{\mathrm{an}} w$ *on the sensor values, i.e. the evaluation* $w = \mathcal{I}_h u_0$. *The latent space processes forward using the n-sweeps* $\mathcal{A}_n$. *The decoder is the semigroup reconstruction from Definition 4.12, which can be written as*

$$\tilde{u}(t,x) = \sum_{q,k} (A^n)_k^q\, \phi_{k,q}(t,x), \qquad \phi_{k,q}(t,x) := \mathbf{1}_{[t_q, t_{q+1})}(t)\, e^{-\lambda_k (t - t_q)}\, e_k(x),$$

with the final node $t = T$ *included as* $\phi_{k,Q}(t,x) := \mathbf{1}_{\{T\}}(t)\, e_k(x)$.

In terms of the branch-trunk construction used in DeepONet [52], $\mathcal{A}_n$ *is a branch network applied to the M sensor values and the functions* $\phi_{k,q}$ *are the trunks. The decay factor* $e^{-\lambda_k (t - t_q)}$ *in the trunk is computed from the learned layer* D_S, *so the trunk has one trainable parameter per mode.*

In terms of PCA-Net, the PCA basis replaces the eigenbasis of L. As an encoder-decoder operator, the architecture belongs to the class considered in [42], where an exponential lower bound over the Lipschitz class was shown. Our work uses the structure of the law to avoid this lower bound.

Proposition 4.10 (Exact Spectral Layers)**.** *Using Assumption 4.5, if we set every learned spectral layer to its exact counterpart, the loop Ψ from Definition 4.7 agrees with the neural Picard map $\Psi^h_{N,Q,\varepsilon}$ from Definition 4.4 (with C^h_N replaced by $D_\mu C^h_N$ when needed), and the architecture computes exactly the neural Picard iterates and the semigroup reconstruction. Consequently, every associated error bound from Sections 4-6 holds for it, and its trainable parameters are those of $\widehat{f}_\varepsilon$ together with the $W_{\mathrm{spec}} \le C_{\mathrm{spec}} N$ spectral weights.*

The proof is in Appendix A.3.

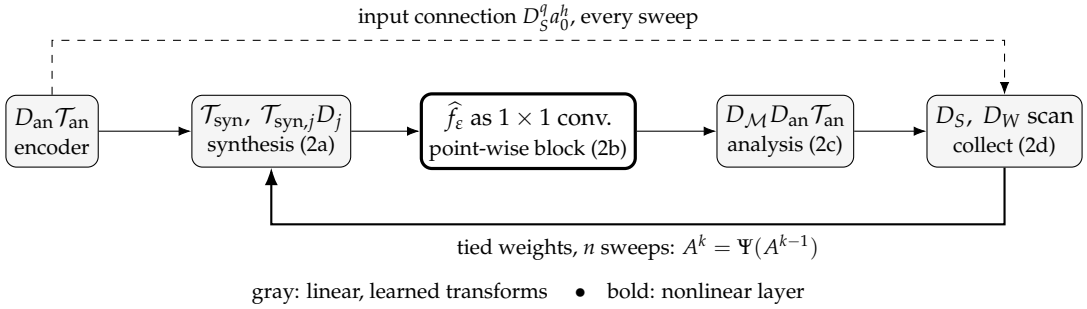


Figure 3: The architecture of Definition 4.7, drawn as a loop. One sweep consists of the four layers (2a)-(2d), the loop is applied n times with the same weights, and the encoded data is connected to every sweep. The trainable parameters are the spectral weights of the shaded layers and those of the single block $\widehat{f}_\varepsilon$, which acts as a 1×1 convolution across the M sample points and is reused at every one of the Q time levels entering the Duhamel sum and at every sweep.

By Lemma 3.4, the block of step (2b) uses the same parameters at all M grid points, Q time levels and n sweeps, and the spectral layers are shared across time levels and sweeps. Therefore,

$$\#\{\text{distinct parameters}\} = W(\widehat{f}_\varepsilon) + W_{\mathrm{spec}},$$

is independent of M, Q and n, while the weight count without sharing is at least $nMQ\,W(\widehat{f}_\varepsilon)$ (see Theorem 5.13).

Definition 4.11 (Discrete Solution Space)**.** *For $a = (a^0, \dots, a^Q) \in (\mathbb{R}^N)^{Q+1}$, let $\mathcal{J}_Q$ be the piecewise-linear interpolation on the temporal grid and define the continuous lifting operation by*

$$\mathcal{L}_{N,Q}a := \mathcal{J}_Q\left(\mathcal{S}_N a^0, \dots, \mathcal{S}_N a^Q\right) \in C([0,T]; E_N), \qquad \|a\|_{X^s_{N,Q}} := \|\mathcal{L}_{N,Q}a\|_{X^s_T}, \quad s \ge 0.$$

The piece-wise linear lifting fits the fixed-point formulation since its norm is the maximum of the norms at the nodes, see Proposition 4.13. However, it ignores the dynamics so we reconstruct the final output using the semigroup itself, which costs one diagonal multiplier.

Definition 4.12 (Semigroup Reconstruction)**.** *For $a = (a^0, \dots, a^Q) \in (\mathbb{R}^N)^{Q+1}$ define $\mathcal{L}^S_{N,Q}a \in C([0,T]; E_N)$ by*

$$\left(\mathcal{L}^S_{N,Q}a\right)(t) := \mathcal{S}_N S_N(t - t_q)\, a^q, \qquad t \in [t_q, t_{q+1}),\ q = 0, \dots, Q-1,$$

and $\left(\mathcal{L}^S_{N,Q}a\right)(T) := \mathcal{S}_N a^Q$. Since $S_N(\tau)$ is a contraction on ℓ^2_α we have $\|\mathcal{L}^S_{N,Q}a\|_{X^\alpha_T} \le \|a\|_{X^\alpha_{N,Q}}$.

The piecewise linear objects $\mathcal{L}_{N,Q}$ and $\mathcal{J}_Q$ are used only to express node-based quantities. For the difference with u, we use $\mathcal{L}^S_{N,Q}$ whose error between nodes is of order $(\Delta t)^\theta$ (see Lemma 5.1) rather than $(\Delta t)^\zeta$.

Proposition 4.13 (Inclusion of the Discrete Ball)**.** *For $a \in (\mathbb{R}^N)^{Q+1}$,*

$$\|\mathcal{L}_{N,Q}a\|_{X^\alpha_T} = \max_{0\le q\le Q} \|a^q\|_{\ell^2_\alpha}.$$

Therefore, with $B^{N,Q}_R := \{a \in (\mathbb{R}^N)^{Q+1} : \max_q \|a^q\|_{\ell^2_\alpha} \le R\}$, we have $\mathcal{L}_{N,Q}B^{N,Q}_R \subseteq B^\alpha_R$ and $\mathcal{L}^S_{N,Q}B^{N,Q}_R \subseteq B^\alpha_R$.

The proof is given in Appendix A.3. In particular, the assumptions made on B_R^α apply to lifted discrete sequences, which makes parameter sharing valid.

Corollary 4.14 (Discrete Input Set)**.** *Let $a \in B_R^{N,Q}$ and let K be the compact set of Assumption 3.5. Then*

$$\Big((\mathcal{S}_N^h a^q)_i,\ (\nabla_h \mathcal{S}_N^h a^q)_i\Big) \in K \qquad \text{for all } q = 0, \dots, Q \text{ and } i = 1, \dots, M.$$

The proof is given in Appendix A.3. Thus a network trained once on K is evaluated only on K, at every space-time location and every Picard iterate, uniformly in N, h, Q and n.

4.3 Spatial Sampling and Learned Quadrature

In the ideal setting, we would choose the quadrature and sampling to match the spectral method. However, we assume that the model is learned and thus we do not have control over the weights and associated sampling points. In this section, we place some assumptions in order to obtain reasonable approximations.

Definition 4.15 (Negative-order Spaces)**.** *For $s > 0$, let X^{-s} be the completion of H under*

$$\|u\|_{X^{-s}} := \|L^{-s/2}u\|_H = \left(\sum_{k=1}^{\infty} \lambda_k^{-s} |\langle u, e_k\rangle_H|^2\right)^{1/2}.$$

Equivalently, X^{-s} is the dual of X^s, and $|\langle u, v\rangle| \le \|u\|_{X^s}\|v\|_{X^{-s}}$ for $u \in X^s$, $v \in X^{-s}$.

The spectral computation behind Assumption 2.3 applies directly for all real $r \ge s$, since it only uses the multiplier $\lambda^{r-s}e^{-2\lambda t}$. Thus $\|S(t)v\|_{X^r} \le C_{r-s}t^{-(r-s)/2}\|v\|_{X^s}$ holds on negative-order spaces and, in particular,

$$\|S(t)v\|_{X^{-\beta}} \le C_{\alpha-\beta}t^{-(\alpha-\beta)/2}\|v\|_{X^{-\alpha}}, \qquad v \in X^{-\alpha}.$$

Assumption 4.16 (Uniform Quadrature Stability)**.** *Let $\mathcal{A} \subset \mathbb{N} \times (0, \infty)$ be the set of admissible pairs (N, h). For each $(N, h) \in \mathcal{A}$, let $\{x_i\}_{i=1}^{M(h)} \subset \overline{\Omega}$ be the spatial sampling locations and $\omega_i^{N,h} \ge 0$ the weights, with $\sum_i \omega_i^{N,h} \le C_{\text{mass}}$ independent of (N, h). The quadrature rule is written as $Q_{N,h}(v, \phi) := \sum_i \omega_i^{N,h} v(x_i)\phi(x_i)$. Assume there are constants $C_{\text{stab}}^0, C_{\text{stab}}^\beta, C_{\text{stab}}^\alpha > 0$, independent of $(N, h) \in \mathcal{A}$, such that:*

(i) For every $\phi \in E_N$, $Q_{N,h}(\phi, \phi) \le C_{\text{stab}}^0\|\phi\|_H^2$. When f depends on ∇u, we also assume $\sum_i \omega_i^{N,h}|\nabla\phi(x_i)|^2 \le C_{\text{stab}}^0\|\nabla\phi\|_H^2$;

(ii) For $\beta > d_x/2$ and for $s \in \{\beta, \alpha\}$

$$\big|Q_{N,h}(v, \phi)\big| \le C_{\text{stab}}^s\|v\|_{X^s}\|\phi\|_{X^{-s}}$$

for every $v \in X^s \cap C(\overline{\Omega})$ and every $\phi \in E_N$.

We write $C_{\text{stab}} := \max_s C_{\text{stab}}^s$. For any grid function $v = (v_i)_{i=1}^M$, we also write $Q_{N,h}(v, \phi) := \sum_i \omega_i^{N,h} v_i\, \phi(x_i)$. Since the weights are nonnegative, the Cauchy-Schwarz inequality and (i) give, for every $\phi \in E_N$,

$$\big|Q_{N,h}(v, \phi)\big| \le (C_{\text{stab}}^0)^{1/2}\|v\|_\omega\|\phi\|_H, \qquad \|v\|_\omega := \left(\sum_i \omega_i^{N,h}|v_i|^2\right)^{1/2} \le C_{\text{mass}}^{1/2} \max_i |v_i|. \tag{6}$$

Assumption 4.16(ii) with $s = \alpha$ also implies that the discrete encoder is bounded on X^α, uniformly in the discretization, shown in the following lemma.

Lemma 4.17 (Discrete Encoder)**.** *Let $(N, h) \in \mathcal{A}$ and $\Pi_{N,h} := \mathcal{S}_N\mathcal{C}_N^h\mathcal{I}_h : X^\alpha \to E_N$. If Assumptions 2.2 and 4.16 hold, then*

$$\left\|\mathcal{C}_N^h\mathcal{I}_h v\right\|_{\ell_\alpha^2} = \left\|\Pi_{N,h}v\right\|_{X^\alpha} \le C_{\text{stab}}^\alpha\|v\|_{X^\alpha},$$

for every $v \in X^\alpha$.

The proof is given in Appendix A.3.

Assumption 4.18 (Uniform Quadrature Consistency)**.** *There exist $r > 0$ and $C_{\mathrm{quad}} > 0$, independent of $(N, h) \in \mathcal{A}$, such that*

$$\left\| \mathcal{C}_N^h \mathcal{I}_h v - \mathcal{C}_N v \right\|_{\ell_s^2} \leq C_{\mathrm{quad}} h^r \|v\|_{X^{s+r}},$$

for $s \in \{\alpha, \beta\}$ and every $v \in X^{s+r}$. We require that

$$r \;\leq\; \gamma_{\mathrm{s}} - \alpha, \tag{7}$$

so that the right-hand side is finite for the regularity in Lemma 2.7.

The limit imposed by (7) come from the regularity lemmas. Without the bootstrap the limit is set by $\gamma < \beta + 2$, and with the bootstrap it is set by the data and the law (see Remark 4.2 and Lemma 2.9). We will show that this makes the exponent d_x/r of Theorem 5.13 at least $d_x/(\gamma_{\mathrm{s}} - \alpha)$.

Remark 4.19 (A uniformly stable and consistent quadrature)**.** *On the torus $\Omega = \mathbb{T}^{d_x}$ take $L = -\nu\Delta + c_0 I$ with $\nu, c_0 > 0$. The term $c_0 u$ can be absorbed into the constitutive law. The eigenfunctions are the standard trigonometric functions with $\lambda_k = \nu|k|^2 + c_0$. The retained modes form a ball $|k| \leq R$, which is in the box $|k|_\infty \leq \Lambda$ with $\Lambda \leq R$. On the uniform grid with $M_1 \geq 2\Lambda + 1$ points in each spatial direction, the trapezoidal rule is exact on products of these retained modes. Therefore, $\mathcal{C}_N^h \mathcal{S}_N^h = I$, (i) holds with constant 1 (also for the gradients inequality), (ii) follows from the embedding $H^\beta \hookrightarrow C(\mathbb{T}^{d_x})$, and Assumption 4.18 holds for every $r > 0$. The same holds on the square $(0, \pi)^{d_x}$ with Dirichlet conditions where the eigenfunctions are the sine modes on the interior grid $x_j = \pi j / M_1$, as long as $M_1 \geq \Lambda + 1$. Since $\lambda_N \geq \nu R^2 \geq \nu \Lambda^2$ for the largest retained $|k| = R$, the resolution condition $h\lambda_N^{1/2} \leq c_{\mathrm{res}}$ implies the scale $c_{\mathrm{res}} \leq \frac{2\pi}{3}\nu^{1/2}$ on the torus or $c_{\mathrm{res}} \leq \frac{\pi}{2}\nu^{1/2}$ on the square.*

For a second-order operator L, the admissible discretizations will impose a resolution condition $h\lambda_N^{1/2} \leq c_{\mathrm{res}}$. In two dimensions, by Weyl's law [84] we expect $\lambda_N \asymp N$ and thus $N \lesssim h^{-2}$. It agrees with the parameter choices of Theorem 5.6 precisely when $r \leq \gamma_{\mathrm{s}} - \alpha$.

4.4 Contraction and Consistency of the Discrete Maps

We need to control the loop depth of the Picard iteration in the discrete setting, but contractive bounds on the neural Picard map are not available. The neural approximations are in the sup-norm while contractive or fixed-point methods require a Lipschitz bound. Instead, we show contraction of the fully discrete map and consistency between the neural approximation and the fully discrete one.

Theorem 4.20 (Fully discrete fixed point)**.** *Let $(N, h) \in \mathcal{A}$ and $Q \in \mathbb{N}$, and set $\Delta t = T/Q$ and $t_q = q\Delta t$ for $q = 0, \ldots, Q$. Assume the conditions of Proposition 2.5 on B_R^α hold, and set $\theta := 1 - \frac{\alpha - \beta}{2} > 0$. Assume, in addition, that the spatial quadrature satisfies Assumption 4.16. Define $F_R := \sup_{u \in B_R^\alpha} \|f(u, \nabla u)\|_{X_T^\beta}$ and let $a_0^h := \mathcal{C}_N^h \mathcal{I}_h u_0$ be in the discrete ball of radius R.*

Then there exists a constant $C > 0$, depending only on the semigroup estimate and on the difference $\alpha - \beta$, but independent of N, h, and Q, such that, for every $a \in B_R^{N,Q}$,

$$\|\Psi_{N,Q}^h(a)\|_{X_{N,Q}^\alpha} \leq \|a_0^h\|_{\ell_\alpha^2} + CC_{\mathrm{stab}} F_R T^\theta,$$

and for every $a, b \in B_R^{N,Q}$,

$$\|\Psi_{N,Q}^h(a) - \Psi_{N,Q}^h(b)\|_{X_{N,Q}^\alpha} \leq \kappa \|a - b\|_{X_{N,Q}^\alpha},$$

with contraction constant $\kappa := CC_{\mathrm{stab}} L_R T^\theta$, independent of N, h, and Q.

If $\|a_0^h\|_{\ell_\alpha^2} + CC_{\mathrm{stab}} F_R T^\theta \leq R$, then $\Psi_{N,Q}^h(B_R^{N,Q}) \subseteq B_R^{N,Q}$. In addition, if $\kappa < 1$, then there exists a unique $a_{N,Q}^h \in B_R^{N,Q}$ satisfying $a_{N,Q}^h = \Psi_{N,Q}^h(a_{N,Q}^h)$, i.e., a unique discrete fixed point.

The proof is given in Appendix A.3. It differs from standard approximation results in numerical analysis in that the quadrature estimates build in the smoothing of the discrete semigroup.

Theorem 4.21 (Neural Consistency)**.** *Let $\Psi^h_{N,Q}$ and $\Psi^h_{N,Q,\varepsilon}$ denote the fully discrete and neural Picard maps of Definition 4.4. Assume that the conditions from Proposition 2.5, Assumption 3.5, and Assumption 4.16 hold, and that the neural approximation satisfies*

$$\sup_{z\in K}\left\|f(z)-\widehat{f}_\varepsilon(z)\right\|_{\mathbb{R}^m}\le\varepsilon$$

on the compact set K from Assumption 3.5. Then there exists $C_R>0$, independent of N, h, Q, and ε, such that for every $a\in B_R^{N,Q}$

$$\left\|\Psi^h_{N,Q,\varepsilon}(a)-\Psi^h_{N,Q}(a)\right\|_{X^\alpha_{N,Q}}\le C_R\,\varepsilon\;\xi_\alpha(N,\Delta t),$$

where the factor is

$$\xi_\alpha(N,\Delta t):=\begin{cases}\min\left\{\lambda_N,\ \Delta t^{-1}\right\}^{\alpha/2-1}, & \alpha>2,\\ 1+\log\left(1+T\min\{\lambda_N,\ \Delta t^{-1}\}\right), & \alpha=2,\\ 1, & \alpha<2,\end{cases}$$

and C_R may depend on T when $\alpha\le 2$. The same bound holds for the lifted maps $\Phi^h_{N,Q,\varepsilon}[u]-\Phi^h_{N,Q}[u]$ in X^α_T for every $u\in B^\alpha_R$ with $b(u)\in B^{N,Q}_R$.

Proof of Theorem 4.21. For the time points $r=0,\dots,Q-1$, the error from the neural network approximation is

$$g^r_\varepsilon:=\widehat{\mathcal{F}}^h_\varepsilon\left(\mathcal{S}^h_N a^r,\nabla_h\mathcal{S}^h_N a^r\right)-\mathcal{F}^h\left(\mathcal{S}^h_N a^r,\nabla_h\mathcal{S}^h_N a^r\right).$$

By Corollary 4.14, the inputs to the local constitutive maps live in the compact K, and thus

$$\|g^r_\varepsilon\|_{\ell^\infty}:=\max_{1\le i\le M(h)}\|(g^r_\varepsilon)_i\|_{\mathbb{R}^m}\le\varepsilon.$$

Taking the difference and applying the continuous Galerkin operation leads to

$$\mathcal{S}_N\left(\Psi^h_{N,Q,\varepsilon}(a)-\Psi^h_{N,Q}(a)\right)^q=\mathcal{S}_N\left[\Delta t\sum_{r=0}^{q-1}S_N(t_q-t_r)\mathcal{C}^h_N g^r_\varepsilon\right],$$

where the initial value terms match and cancel out. Since $\mathcal{S}_N$ is an isometry from ℓ^2_0 to E_N equipped with the X^0-norm, we first estimate $\mathcal{C}^h_N$ on bounded grid functions. For $w\in\mathbb{R}^{M(h)}$,

$$\|\mathcal{C}^h_N w\|_{\ell^2_0}=\sup_{\|c\|_{\ell^2_0}=1}\left|\sum_{i=1}^{M(h)}\omega_i^{N,h}w_i(\mathcal{S}_N c)(x_i)\right|\le\|w\|_{\ell^\infty}\left(\sum_{i=1}^{M(h)}\omega_i^{N,h}\right)^{1/2}\sup_{\substack{\phi\in E_N\\ \|\phi\|_{X^0}=1}}\left(\sum_{i=1}^{M(h)}\omega_i^{N,h}|\phi(x_i)|^2\right)^{1/2},$$

where we set $\phi:=\mathcal{S}_N c$, so that $\|\phi\|_{X^0}=\|c\|_{\ell^2_0}$ by orthonormality of the e_k. By the uniform bound on the total mass and the sampling inequality of Assumption 4.16(i),

$$\sum_{i=1}^{M(h)}\omega_i^{N,h}|\phi(x_i)|^2\le C^0_{\mathrm{stab}}\|\phi\|^2_{X^0}.$$

Thus taking the sup over norm 1 functions, $\|\mathcal{C}^h_N w\|_{\ell^2_0}\le C_{\mathrm{samp}}\|w\|_{\ell^\infty}$, where $C_{\mathrm{samp}}:=(C_{\mathrm{mass}}C^0_{\mathrm{stab}})^{1/2}$, and therefore, since $\mathcal{C}^h_N$ acts component-wise on the m components of g^r_ε, each of which has ℓ^∞ norm at most ε,

$$\|\mathcal{C}^h_N g^r_\varepsilon\|_{\ell^2_0}=\left(\sum_{j=1}^m\|\mathcal{C}^h_N(g^r_\varepsilon)^{(j)}\|^2_{\ell^2_0}\right)^{1/2}\le\sqrt{m}\,C_{\mathrm{samp}}\,\varepsilon,$$

and the $\sqrt{m}$ factor can be absorbed by C_R in the next steps.

For $\tau > 0$, the spectral form of $S_N(\tau)$ gives

$$\begin{aligned}\left\|\mathcal{S}_N S_N(\tau)\mathcal{C}_N^h g_\varepsilon^r\right\|_{X^\alpha} &= \left(\sum_{k=1}^N \lambda_k^\alpha e^{-2\lambda_k\tau}\left|(\mathcal{C}_N^h g_\varepsilon^r)_k\right|^2\right)^{1/2} \\ &\leq \min\left\{\lambda_N^{\alpha/2}, C_\alpha \tau^{-\alpha/2}\right\}\|\mathcal{C}_N^h g_\varepsilon^r\|_{\ell_0^2} \\ &\leq C_R\varepsilon\min\left\{\lambda_N^{\alpha/2}, \tau^{-\alpha/2}\right\}.\end{aligned}$$

And so the difference is bounded by:

$$\left\|\mathcal{S}_N\left(\Psi_{N,Q,\varepsilon}^h(a) - \Psi_{N,Q}^h(a)\right)^q\right\|_{X^\alpha} \leq C_R\varepsilon\Delta t\sum_{j=1}^q \min\left\{\lambda_N^{\alpha/2}, (j\Delta t)^{-\alpha/2}\right\}.$$

We estimate the discrete sum, first in the setting where $\alpha > 2$, so $\alpha/2 > 1$. If $\lambda_N\Delta t \leq \frac{1}{2}$, set $J := \left\lfloor \frac{1}{\lambda_N\Delta t}\right\rfloor$, so that $J \geq 1$ and $J \geq \frac{1}{2\lambda_N\Delta t}$. For the terms with $j \leq J$,

$$\Delta t\sum_{j=1}^{\min\{q,J\}} \lambda_N^{\alpha/2} \leq \Delta t\, J\, \lambda_N^{\alpha/2} \leq \lambda_N^{\alpha/2-1},$$

where we used $\Delta t J \leq \lambda_N^{-1}$. For the remaining terms, note that since $s^{-\alpha/2}$ is monotone, we have the upper bound to the integral by

$$\Delta t\sum_{j=J+1}^q (j\Delta t)^{-\alpha/2} \leq \int_{J\Delta t}^\infty s^{-\alpha/2}\,ds = \tfrac{2}{\alpha-2}(J\Delta t)^{1-\alpha/2} \leq C\lambda_N^{\alpha/2-1},$$

using $J\Delta t \geq (2\lambda_N)^{-1}$. Thus, in the first case,

$$\Delta t\sum_{j=1}^q \min\left\{\lambda_N^{\alpha/2}, (j\Delta t)^{-\alpha/2}\right\} \leq C\lambda_N^{\alpha/2-1}.$$

If $\lambda_N\Delta t > \frac{1}{2}$, then $\min\{\lambda_N^{\alpha/2}, (j\Delta t)^{-\alpha/2}\} \leq (j\Delta t)^{-\alpha/2}$, and so

$$\Delta t\sum_{j=1}^q (j\Delta t)^{-\alpha/2} \leq \Delta t^{1-\alpha/2}\sum_{j=1}^\infty j^{-\alpha/2} \leq C\Delta t^{1-\alpha/2},$$

where we used that $\alpha/2 > 1$ so the sum is bounded.

Both settings give a bound of the form $C\min\{\lambda_N, \Delta t^{-1}\}^{\alpha/2-1}$. In the first setting, $\lambda_N \leq (2\Delta t)^{-1} < \Delta t^{-1}$, so $\min\{\lambda_N, \Delta t^{-1}\} = \lambda_N$. In the second setting, $\Delta t^{-1} < 2\lambda_N$, so $\Delta t^{-1} \leq 2\min\{\lambda_N, \Delta t^{-1}\}$, and since $\alpha/2 - 1 > 0$,

$$\Delta t^{1-\alpha/2} = (\Delta t^{-1})^{\alpha/2-1} \leq 2^{\alpha/2-1}\min\{\lambda_N, \Delta t^{-1}\}^{\alpha/2-1}.$$

Therefore, in all cases,

$$\Delta t\sum_{j=1}^q \min\left\{\lambda_N^{\alpha/2}, (j\Delta t)^{-\alpha/2}\right\} \leq C\,\min\left\{\lambda_N,\ \Delta t^{-1}\right\}^{\alpha/2-1},$$

where C depends only on α and is independent of N, h, Q, and q.

For $\alpha < 2$, the summand $(j\Delta t)^{-\alpha/2}$ is integrable at the origin, and since it is decreasing,

$$\Delta t\sum_{j=1}^q (j\Delta t)^{-\alpha/2} \leq \int_0^T s^{-\alpha/2}\,ds = \frac{T^{1-\alpha/2}}{1-\alpha/2}.$$

For $\alpha = 2$ the same splitting applies. If we have $\lambda_N \Delta t \le \frac{1}{2}$, the terms $j \le J$ contribute at most $\Delta t\, J \lambda_N \le 1$, and the rest gives

$$\sum_{j=J+1}^{q} j^{-1} \le \log(q/J) \le \log(2T\lambda_N),$$

since $q \le T/\Delta t$ and $J \ge (2\lambda_N \Delta t)^{-1}$. If we have $\lambda_N \Delta t > \frac{1}{2}$, the sum is at most

$$\sum_{j=1}^{q} j^{-1} \le 1 + \log(T/\Delta t) \le 1 + \log\left(2T \min\{\lambda_N, \Delta t^{-1}\}\right).$$

In both cases the sum is at most $3\,\xi_2(N, \Delta t)$. Taking the maximum over q and using Definition 4.11 gives the result, where the statement for the lifted maps comes from applying the estimates to $a = b(u)$. □

We kept the minimum of λ_N and Δt^{-1} instead of bounding it by one term since the bound is sharp (see Proposition 7.1) and under the choices of Theorem 5.6 the two entries are comparable. The cases of ξ_α come from the discrete sum in its proof, specifically, the integrability of $t^{-\alpha/2}$ for different α's. The network error acts as a perturbation of the forcing of size ε at every time level. The discrete Duhamel sum maps it into X^α with the kernel $\min\{\lambda_N^{\alpha/2}, \tau^{-\alpha/2}\}$, which for $\alpha > 2$ is not integrable near $\tau = 0$ with error scaling with $\min\{\lambda_N, \Delta t^{-1}\}^{\alpha/2-1}$. If we measure the error in X^s with $s < 2$, then the kernel is integrable and we avoid this issue.

Theorem 4.22 (Consistency of the fully discrete mild map)**.** *Let $u \in B_R^\alpha \subset X_T^\alpha$ be the mild solution of $u = \Phi[u]$ with the conditions from Propositions 2.5 and 2.6, and let $\Phi_{N,Q}^h$ be the fully discrete mild map. Assume that the spatial discretization satisfies Assumptions 4.16 and 4.18 with $0 < r \le \gamma_{\mathrm{s}} - \alpha$, that*

$$F(t) := f(u(t), \nabla u(t)) \in L^\infty\left(0, T; X^\beta\right), \qquad \|F\|_{\infty,\beta} + [F]_\zeta \le C_R,$$

and,

$$\sup_{\substack{(N,h)\in\mathcal{A} \\ 0<t\le T}} t^\varrho \, \|f(P_N u(t), \nabla P_N u(t))\|_{X^{\beta+r}} \le C_R,$$

where $\varrho = 0$ if $\gamma_{\mathrm{s}} = \gamma' > \gamma$ or otherwise $\varrho = r/2$. Define $E_{\mathrm{init}}^h := \|a_0^h - \mathcal{C}_N u_0\|_{\ell_\alpha^2}$. Then there exists $C_R > 0$, independent of N, h and Q, such that

$$\max_{0\le q\le Q} \left\| \left(\Phi_{N,Q}^h[u]\right)(t_q) - u(t_q) \right\|_{X^\alpha} \le C_R \left[E_{\mathrm{init}}^h + \|(I - P_N)u_0\|_{X^\alpha} + \lambda_{N+1}^{-(\gamma_{\mathrm{s}}-\alpha)/2} + h^r + (\Delta t)^{\zeta_{\mathrm{t}}} \right] =: E_{\mathrm{cons}}.$$

The same bound holds for $\|\Phi_{N,Q}^h[u] - \mathcal{J}_Q u\|_{X_T^\alpha}$, where $\mathcal{J}_Q u$ is the piecewise-linear interpolant of the values $u(t_q)$.

The proof is given in Appendix A.3. The bound for $f(P_N u, \nabla P_N u)$ follows from Assumption 2.8 with $\gamma' \ge \alpha + r$. For $\varrho = 0$, this is shown in the proof. For $\varrho = r/2$ it follows from the estimate $\|f(v, \nabla v)\|_{X^{\beta+r}} \le K_R(1 + \|v\|_{X^{\alpha+r}})$ for $\|v\|_{X^\alpha} \le R$, i.e., Assumption 2.8 with $s = \alpha + r$, together with the interpolation inequality and Lemma 2.7 Part 2:

$$\|P_N u(t)\|_{X^{\alpha+r}} \le \|u(t)\|_{X^\alpha}^{1-r/(2\zeta)} \|u(t)\|_{X^\gamma}^{r/(2\zeta)} \le R^{1-r/(2\zeta)} R_\gamma^{r/(2\zeta)} \, t^{-r/2}.$$

The weighted norm is needed when we only assume X^α, since $\|P_N u_0\|_{X^{\alpha+r}}$ cannot be controlled near zero.

5 Total Error and Structural Complexity

In this section, we put together the prior consistency, contraction, and smoothing estimates to obtain an upper bound on the number of parameters needed to construct a neural network approximation to the

solution of the class of semilinear equations considered. Throughout this section, we assume the conditions and results from Theorems 4.20, 4.21, 4.22, and Lemma 2.7 hold. Let

$$b = (b^0, \dots, b^Q), \qquad b^q := \mathcal{C}_N^h \mathcal{I}_h u(t_q),$$

denote coefficients of the exact solution after sampling on the grid (so sampling then transformation). We use just the contraction of the discrete Picard map $\Psi_{N,Q}^h$, whose constant is $\kappa < 1$ and thus by Theorem 4.20 has a unique fixed point $a_\infty \in B_R^{N,Q}$. We first record two elementary lemmas.

Lemma 5.1 (Reconstruction)**.** *Let u be the mild solution with $\|u\|_{X_T^\alpha} \le R$ and $F_R = \sup_{v \in B_R^\alpha} \|f(v, \nabla v)\|_{X_T^\beta}$, and let $a \in (\mathbb{R}^N)^{Q+1}$. Then*

$$\left\| \mathcal{L}_{N,Q}^S a - u \right\|_{X_T^\alpha} \le \underbrace{\max_{0 \le q \le Q} \left\| \mathcal{S}_N a^q - u(t_q) \right\|_{X^\alpha}}_{\text{nodal error}} + C\, F_R\, (\Delta t)^\theta,$$

where C depends only on the difference $\alpha - \beta$. If Assumption 2.8 holds and $u_0 \in X^{\gamma'}$, the term $CF_R(\Delta t)^\theta$ improves to $CK_R(1 + R_{\gamma'})(\Delta t)^{\min\{1, \theta + \zeta_s\}}$.

The proof uses the mild formulation starting at each node and is given in Appendix A.3. The reconstruction error is order $(\Delta t)^\theta$ rather than $(\Delta t)^\zeta$, which is better since $\zeta < \theta$. It does not require additional regularity of u beyond what is given through the mild formulation. Using the bootstrap, it is at most a constant times $(\Delta t)^{\zeta_t}$.

When the dynamics only yields data in X^α, we cannot control parts of the error in the X^γ norm (which will be unbounded as $t \to 0^+$). Instead we split the error into a part that is controlled by the semi-group smoothing (an algebraic rate) and a part that measures the tail of the data (no explicit rate). We define the tail of the initial data below.

Definition 5.2 (Input Data Tails and Encoding)**.** *For $u_0 \in X^\alpha$ and admissible (N, h), let*

$$\varepsilon_N^{\text{tail}}(u_0) := \|(I - P_N) u_0\|_{X^\alpha}, \qquad \varepsilon_{N,h}^{\text{enc}}(u_0) := \sup_{0 \le t \le T} \left\| \left(\mathcal{C}_N^h \mathcal{I}_h - \mathcal{C}_N \right) S(t) u_0 \right\|_{\ell^2_\alpha}.$$

The supremum over the entire dynamics $S(t)u_0$ is needed since $\mathcal{C}_N^h \mathcal{I}_h$ only commutes with $S(t)$ when the quadrature is exact, which can lead to different decay rates. Both terms in the definition tend to 0 but they do not admit rates directly. When extra regularity is given, we have $\varepsilon_N^{\text{tail}}(u_0) \le \lambda_{N+1}^{-\zeta_s} \|u_0\|_{X^{\gamma_s}}$ if $u_0 \in X^{\gamma_s}$, and $\varepsilon_{N,h}^{\text{enc}}(u_0) \le C_{\text{quad}} h^r \|u_0\|_{X^{\alpha+r}}$ if $u_0 \in X^{\alpha+r}$.

Lemma 5.3 (Sampling Error of Mild Solution)**.** *Let $u \in B_R^\alpha$ be the mild solution with initial $u_0 \in X^\alpha$ and let Assumption 4.18 hold with $0 < r \le \gamma_s - \alpha$. Then*

$$E_{\text{samp}} := \max_{0 \le q \le Q} \left\| \mathcal{S}_N b^q - u(t_q) \right\|_{X^\alpha} \le \varepsilon_N^{\text{tail}}(u_0) + \varepsilon_{N,h}^{\text{enc}}(u_0) + C_R \left(\lambda_{N+1}^{-\zeta_s} + h^r \right),$$

where C_R depends on R, T, F_R and the constants of Assumptions 2.3 and 4.18, and when using bootstrap, it also depends on $R_{\gamma'}$ and $\|u_0\|_{X^{\gamma'}}$.

In particular, $b \in B_R^{N,Q}$ if $\|u\|_{X_T^\alpha} \le R_0 < R$ and $E_{\text{samp}} \le R - R_0$.

The proof splits the solution into its linear dynamics and the Duhamel operator, and is given in Appendix A.3.

5.1 Total Error of the Neural Picard Iteration

We look at the total error induced by the Neural Picard iteration and compare the n-iterate to the continuous mild solution.

Theorem 5.4 (Total error). *Let u be the mild solution with $\|u\|_{X_T^\alpha} \le R_0 < R$, and assume the following hold:*

(i) $\kappa < 1$ *and* $\Psi^h_{N,Q}(B^{N,Q}_R) \subseteq B^{N,Q}_{R_0}$;

(ii) $E_{\text{samp}} \le R - R_0$, *so that* $b \in B^{N,Q}_R$ *by Lemma 5.3;*

(iii) $E_{\text{net}} := C_R\, \varepsilon\, \xi_\alpha(N, \Delta t)$, *with* ξ_α *from Theorem 4.21, satisfies*

$$\frac{E_{\text{net}}}{1-\kappa} \le R - R_0. \tag{8}$$

For $a^0 \in B^{N,Q}_{R_0}$, define the exact and neural Picard iterates by $a^n := \Psi^h_{N,Q}(a^{n-1})$, $a^n_\varepsilon := \Psi^h_{N,Q,\varepsilon}(a^{n-1}_\varepsilon)$, where $a^0_\varepsilon := a^0$. Then $a^n, a^n_\varepsilon \in B^{N,Q}_R$ for every $n \ge 0$, and

$$\left\| \mathcal{L}^S_{N,Q} a^n_\varepsilon - u \right\|_{X^\alpha_T} \le \underbrace{\kappa^n \left\| a^0 - a_\infty \right\|_{X^\alpha_{N,Q}}}_{\textit{iteration}} + \underbrace{\frac{E_{\text{net}}}{1-\kappa}}_{\textit{network}} + \underbrace{\frac{E_{\text{cons}} + E_{\text{samp}}}{1-\kappa} + E_{\text{samp}}}_{\textit{discretization}} + \underbrace{E_{\text{rec}}}_{\textit{reconstruction}},$$

where E_{cons}, E_{samp} and E_{rec} are the error terms from Theorem 4.22, Lemma 5.3 and Lemma 5.1. Consequently there is $C_R > 0$, independent of N, h, Q, n and ε, such that

$$\left\| \mathcal{L}^S_{N,Q} a^n_\varepsilon - u \right\|_{X^\alpha_T} \le C_R \left[\kappa^n + \varepsilon\, \xi_\alpha(N, \Delta t) + \varepsilon^{\text{tail}}_N(u_0) + \varepsilon^{\text{enc}}_{N,h}(u_0) + \lambda_{N+1}^{-\zeta_s} + h^r + (\Delta t)^{\zeta_t} \right].$$

Condition (i) holds uniformly in N, h and Q once we set $R_0 \ge \max\{\|u\|_{X^\alpha_T}, 2C^\alpha_{\text{stab}}\|u_0\|_{X^\alpha}\}$ and $R > R_0$, with possible shrinking of the final time T. This can be seen by applying Theorem 4.20 which needs $\|a^h_0\|_{\ell^2_\alpha} + CC_{\text{stab}}F_R T^\theta \le R_0$ and $CC_{\text{stab}}L_R T^\theta < 1$. We have $\|a^h_0\|_{\ell^2_\alpha} \le C^\alpha_{\text{stab}}\|u_0\|_{X^\alpha} \le R_0/2$ by Lemma 4.17. In the final bound, we use that $E^h_{\text{init}} \le \varepsilon^{\text{enc}}_{N,h}(u_0)$ and $\|(I - P_N)u_0\|_{X^\alpha} = \varepsilon^{\text{tail}}_N(u_0)$, while E_{rec} can be absorbed into $(\Delta t)^{\zeta_t}$.

Proof of Theorem 5.4. We set the norm for the coefficient sequences as $\|\cdot\|_{X^\alpha_{N,Q}}$ of Definition 4.11, use $\|\cdot\|$ for short.

We want control over the neural discrete map, but will use the non-neural discrete map as a first approximation. First, we can show that the neural iterates remain admissible and bound its closeness to the discrete iterates. Since $a^0 \in B^{N,Q}_{R_0} \subseteq B^{N,Q}_R$ and $\Psi^h_{N,Q}$ maps $B^{N,Q}_R$ into $B^{N,Q}_{R_0}$ by Theorem 4.20, induction on n shows that $a^n \in B^{N,Q}_{R_0}$ for every $n \ge 1$. For the neural iterates, we claim that

$$a^n_\varepsilon \in B^{N,Q}_R \qquad \text{and} \qquad \|a^n_\varepsilon - a^n\| \;\le\; \frac{E_{\text{net}}}{1-\kappa}. \tag{9}$$

for all $n \ge 0$. For $n = 0$ it is true since $a^0_\varepsilon = a^0 \in B^{N,Q}_{R_0} \subseteq B^{N,Q}_R$. Next, if (9) holds for $n-1$, then a^{n-1}_ε and a^{n-1} both lie in $B^{N,Q}_R$, so we can apply the consistency Theorem 4.21 to a^{n-1}_ε and the Lipschitz estimate of Theorem 4.20 to the difference, to get:

$$\begin{aligned}
\|a^n_\varepsilon - a^n\| &\le \left\| \Psi^h_{N,Q,\varepsilon}(a^{n-1}_\varepsilon) - \Psi^h_{N,Q}(a^{n-1}_\varepsilon) \right\| + \left\| \Psi^h_{N,Q}(a^{n-1}_\varepsilon) - \Psi^h_{N,Q}(a^{n-1}) \right\| && (10)\\
&\le E_{\text{net}} + \kappa \left\| a^{n-1}_\varepsilon - a^{n-1} \right\| && (11)\\
&\le E_{\text{net}} + \frac{\kappa\, E_{\text{net}}}{1-\kappa} = \frac{E_{\text{net}}}{1-\kappa}, && (12)
\end{aligned}$$

where we use iteration definitions, Theorem 4.21 and Theorem 4.20 for (11), and the induction hypothesis for (12). Combining the resulting bound with $a^n \in B^{N,Q}_{R_0}$ and the radius condition (8),

$$\|a^n_\varepsilon\| \;\le\; \|a^n\| + \|a^n_\varepsilon - a^n\| \;\le\; R_0 + \frac{E_{\text{net}}}{1-\kappa} \;\le\; R_0 + (R - R_0) \;=\; R,$$

which shows the induction and proves (9).

Next, we look at the non-neural discrete iteration, which converges to its fixed point by Theorem 4.20, $\Psi^h_{N,Q}$ using that it is a κ-contraction on $B^{N,Q}_R$. Specifically, there exists a unique fixed point a_∞ and

$$\|a^n - a_\infty\| \;\le\; \kappa^n \left\|a^0 - a_\infty\right\| \;\le\; 2R\,\kappa^n, \tag{13}$$

using that both sequences are in $B^{N,Q}_R$.

We can compare the fixed point with the sampled exact solution b (with components $b^q = \mathcal{C}^h_N \mathcal{I}_h u(t_q)$). By Lemma 5.3 and the condition $E_{\text{samp}} \le R - R_0$, we have $b \in B^{N,Q}_R$. Therefore, the contraction estimate from Theorem 4.20 can be applied to get:

$$\|a_\infty - b\| \le \left\|\Psi^h_{N,Q}(a_\infty) - \Psi^h_{N,Q}(b)\right\| + \left\|\Psi^h_{N,Q}(b) - b\right\| \le \kappa\,\|a_\infty - b\| + \left\|\Psi^h_{N,Q}(b) - b\right\|,$$

and since $\kappa < 1$:

$$\|a_\infty - b\| \le \frac{1}{1-\kappa}\left\|\Psi^h_{N,Q}(b) - b\right\|. \tag{14}$$

The residual measures the difference when sampling the exact solution and using the discrete map, and needs a consistency and sampling bound. Recall that $\mathcal{L}_{N,Q}\Psi^h_{N,Q}(b)$ and $\mathcal{L}_{N,Q}b$ are piecewise linear in time, with $\mathcal{S}_N\Psi^h_{N,Q}(b)^q$ and $\mathcal{S}_N b^q$ at the nodes, respectively. Thus, by Proposition 4.13 the norm of a piecewise linear function is attained at a node and

$$\left\|\Psi^h_{N,Q}(b) - b\right\| = \left\|\mathcal{L}_{N,Q}\Psi^h_{N,Q}(b) - \mathcal{L}_{N,Q}b\right\|_{X^\alpha_T} = \left\|\Phi^h_{N,Q}[u] - \mathcal{L}_{N,Q}b\right\|_{X^\alpha_T} \tag{15}$$

$$\le \left\|\Phi^h_{N,Q}[u] - \mathcal{J}_Q u\right\|_{X^\alpha_T} + \left\|\mathcal{J}_Q u - \mathcal{L}_{N,Q}b\right\|_{X^\alpha_T} \le E_{\text{cons}} + E_{\text{samp}}, \tag{16}$$

where the second equality (15) uses $\mathcal{L}_{N,Q}\Psi^h_{N,Q}(b) = \Phi^h_{N,Q}[u]$ from Definition 4.4, and the first term in (16) is bounded by Theorem 4.22 and the second by Lemma 5.3 (both $\mathcal{J}_Q u$ and $\mathcal{L}_{N,Q}b$ having nodal values of $u(t_q)$ and $\mathcal{S}_N b^q$).

The total error can be chained. By Lemma 5.1, the reconstruction error of $\mathcal{L}^S_{N,Q}$ is controlled by E_{rec} and by the nodal error:

$$\max_{0\le q\le Q}\left\|\mathcal{S}_N(a^n_\varepsilon)^q - u(t_q)\right\|_{X^\alpha} \le \|a^n_\varepsilon - a^n\| + \|a^n - a_\infty\| + \|a_\infty - b\| + \max_{0\le q\le Q}\left\|\mathcal{S}_N b^q - u(t_q)\right\|_{X^\alpha} \tag{17}$$

$$\le \frac{E_{\text{net}}}{1-\kappa} + 2R\,\kappa^n + \frac{E_{\text{cons}} + E_{\text{samp}}}{1-\kappa} + E_{\text{samp}}, \tag{18}$$

where (17) is the triangle inequality and uses that $\|a\| = \max_q \|\mathcal{S}_N a^q\|_{X^\alpha}$ from Proposition 4.13. For equation (18), we bound each term by (9), (13), (14) with (16), and Lemma 5.3. Since $1 - \kappa \le 1$, the sum is at most

$$\frac{E_{\text{net}} + E_{\text{cons}} + 2E_{\text{samp}}}{1-\kappa} + 2R\kappa^n,$$

which gives us the main result (after adding the reconstruction term E_{rec} for Lemma 5.1).

To get the rates, we use the bounds in Theorem 4.22 and Lemma 5.3. □

Remark 5.5 (Neural Approximation Only Uses Max-Norm Bounds)**.** *The proofs were written specifically to only require that the exact discrete map be a Lipschitz contraction. The neural networks errors are included through a point-wise consistency estimate (Theorem 4.21), so we do not need to control the Lipschitz constant of $\widehat{f}_\varepsilon$, which would be impractical. This shows that the approximation can inherit smoother control, X^α bounds, from a max-norm controlled network. This is related to the perturbed fixed-point iterations of [23], where the approximate iterations are assumed to be Lipschitz.*

5.2 Parameter Complexity Bounds

Theorem 5.6 (Learnable Parameter Count, One Block)**.** *Assume the conditions from Theorem 5.4, Assumption 3.5, Assumption 2.8 with $\gamma' \geq \alpha + r$, and (26). Set the exponent to $p_{\max} := \max_{j \in \mathcal{V}_{\mathrm{nl}}} p_j = 1/s_{\min}$, where p_j are the node exponents from Theorem 3.7 and $s_{\min}$ is the minimal smoothness exponent of a DAG node (where we set $p_{\max} = 0$ if there is none). Let C_R be the constant of Theorem 5.4 and ε_0 the error threshold for Theorem 3.7, and set*

$$c_{\mathrm{err}} := \frac{1}{7C_R}, \qquad \eta_0 := \min\left\{1,\ \tfrac{7}{2}(1-\kappa)(R-R_0),\ \frac{\varepsilon_0}{c_{\mathrm{err}}},\ \frac{T^{\zeta_t}}{c_{\mathrm{err}}}\right\}$$

and $\eta \in (0, \eta_0]$ such that

(i) $n := \lceil \log(1/(c_{\mathrm{err}}\eta))/\log(1/\kappa) \rceil$;

(ii) $N := \min\left\{N' :\ \lambda_{N'+1} \geq (c_{\mathrm{err}}\eta)^{-1/\zeta_s} \text{ and } \varepsilon^{\mathrm{tail}}_{N'}(u_0) \leq c_{\mathrm{err}}\eta\right\}$;

(iii) $h := 2^{-k}$ *where k is smallest integer such that* $h^r \leq c_{\mathrm{err}}\eta$, $(N,h) \in \mathcal{A}$ *and* $\varepsilon^{\mathrm{enc}}_{N,h}(u_0) \leq c_{\mathrm{err}}\eta$, *and* $M := M(h)$;

(iv) $Q := \left\lceil T(c_{\mathrm{err}}\eta)^{-1/\zeta_t} \right\rceil$ *and* $\Delta t := T/Q$;

(v) $\varepsilon := c_{\mathrm{err}}\eta / \max\{1, \xi_\alpha(N, \Delta t)\}$.

Then the architecture from Definition 4.7 with $\widehat{f}_\varepsilon$ from Theorem 3.7 and the spectral weights from Proposition 4.10, satisfies $\|\mathcal{L}^S_{N,Q} a^n_\varepsilon - u\|_{X^\alpha_T} \leq \eta$, and

$$\varepsilon = c_{\mathrm{err}}\eta \quad (\text{if } \alpha < 2), \qquad \varepsilon \geq \frac{c_{\mathrm{err}}\eta}{1 + \log\left(1 + 2T(c_{\mathrm{err}}\eta)^{-1/\zeta}\right)} \quad (\text{if } \alpha = 2), \qquad \varepsilon \geq 2^{1-\alpha/2}(c_{\mathrm{err}}\eta)^{\chi} \quad (\text{if } \alpha > 2).$$

In addition, if $u_0 \in X^{\gamma_s}$ and $\alpha > 2$, then $\varepsilon \geq \max\{1, \lambda_1\}^{1-\alpha/2} m_0^{-(\alpha-2)/(2\zeta_s)} (c_{\mathrm{err}}\eta)^{\chi_s}$, where $\chi_s := 1 + \frac{(\alpha-2)_+}{2\zeta_s} \leq \chi$ and $m_0 := \max\{1, \|u_0\|_{X^{\gamma_s}}\}$. Thus the depth is bounded by $\mathrm{dep}(\widehat{f}_\varepsilon) \leq C \log \eta^{-1}$ and the parameter count by

$$W(\widehat{f}_\varepsilon) \leq \begin{cases} C\,\eta^{-p_{\max}} \log \eta^{-1}, & \alpha < 2, \\ C\,\left(\eta^{-1} \log \eta^{-1}\right)^{p_{\max}} \log \eta^{-1}, & \alpha = 2, \\ C\,\eta^{-p_{\max}\chi_s} \log \eta^{-1}, & \alpha > 2, \end{cases}$$

with C depending on the computation graph of f, c_{err} and T, and for $\alpha > 2$ also on m_0 and λ_1.

The min in (ii) and (iii) exist by (26) and Lemma 6.4. The conditions on the tail of the input data is the only place without an algebraic rate so those statement are not explicit. The trainable count and the size of the n-sweep network are shown in Theorem 5.14. For laws in divergence form, (v) uses $\xi_{\alpha+1}$ instead of ξ_α (see Proposition 7.2).

Proof of Theorem 5.6. The bound in Theorem 5.4 is C_R times seven terms, thus we can set $C_R c_{\mathrm{err}} = \frac{1}{7}$, so that each term is at most $c_{\mathrm{err}}\eta$. It holds for κ^n by assumption (i) in the theorem. By (v) it holds for $\varepsilon\xi_\alpha \leq \varepsilon \max\{1, \xi_\alpha\}$. Condition (ii) gives us control for the data tail bound and $\lambda_{N+1}^{-\zeta_s}$. The spatial component h^r is given by (iii) and the temporal resolution $(\Delta t)^{\zeta_t}$ by (iv). Lastly, the reconstruction error from Lemma 5.1 is order $(\Delta t)^{\zeta_t}$ and is absorbed by the other term.

We next check that we have the assumptions for applying Theorem 5.4. Since C_R is an upper-bound constant, we may make it larger before choosing the parameters. We pick $C_R \geq 2\max\{1, C_{\mathrm{smp}}\}$, where C_{smp} is the constant C_R from Lemma 5.3. Since $\eta \leq \eta_0 \leq \frac{7}{2}(1-\kappa)(R-R_0)$, we have $E_{\mathrm{net}} \leq \frac{\eta}{7} \leq \frac{1}{2}(1-\kappa)(R-R_0)$, which gives (8). Each of $\varepsilon^{\mathrm{tail}}_N$, $\varepsilon^{\mathrm{enc}}_{N,h}$, $\lambda_{N+1}^{-\zeta_s}$ and h^r is at most $c_{\mathrm{err}}\eta$, so Lemma 5.3 and $c_{\mathrm{err}} = 1/(7C_R)$ gives us the bound

$$E_{\mathrm{samp}} \leq 4\max\{1, C_{\mathrm{smp}}\} c_{\mathrm{err}}\eta \leq \frac{2\eta}{7} \leq (1-\kappa)(R-R_0) \leq R - R_0.$$

Finally $\varepsilon \leq c_{\mathrm{err}}\eta_0 \leq \varepsilon_0$, so Theorem 3.7 applies.

For the lower bounds on ε, since $c_{\mathrm{err}}\eta \le \min\{T^{\zeta_t}, 1\}$ and $\zeta_t \ge \zeta$, we have $\Delta t^{-1} = Q/T \le 2(c_{\mathrm{err}}\eta)^{-1/\zeta_t} \le 2(c_{\mathrm{err}}\eta)^{-1/\zeta}$. For $\alpha < 2$, the factor is $\xi_\alpha = 1$. For $\alpha = 2$, the factor is bounded by $\xi_2 \le 1 + \log(1 + T\Delta t^{-1})$, which gives the stated result. For $\alpha > 2$, we have

$$\xi_\alpha \le (\Delta t^{-1})^{\alpha/2-1} \le 2^{\alpha/2-1}(c_{\mathrm{err}}\eta)^{-(\alpha-2)/(2\zeta)}.$$

The right-hand side is at least 1, so it also bounds $\max\{1, \xi_\alpha\}$. By (v),

$$\varepsilon = \frac{c_{\mathrm{err}}\eta}{\max\{1, \xi_\alpha\}} \ge 2^{1-\alpha/2}(c_{\mathrm{err}}\eta)^{1+\frac{\alpha-2}{2\zeta}} = 2^{1-\alpha/2}(c_{\mathrm{err}}\eta)^{\chi}.$$

If $u_0 \in X^{\gamma_s}$, set $\Lambda := (m_0/(c_{\mathrm{err}}\eta))^{1/\zeta_s} > 1$ and let N'' be the smallest index where the eigenvalues are above Λ, i.e., $\lambda_{N''+1} \ge \Lambda$. It satisfies both conditions in (ii), the spectral one because $m_0 \ge 1$ and the tail bound because $\varepsilon^{\mathrm{tail}}_{N''}(u_0) \le \lambda^{-\zeta_s}_{N''+1}\|u_0\|_{X^{\gamma_s}} \le c_{\mathrm{err}}\eta$, so $N \le N''$. By construction, $\lambda_{N''} < \Lambda$ (we assume that $N'' \ge 2$ or set $\lambda_{N''} = \lambda_1$ otherwise), so $\lambda_N \le \max\{1, \lambda_1\}\Lambda$. If we bound ξ_α by the first entry we get $\max\{1, \xi_\alpha\} \le (\max\{1, \lambda_1\}\Lambda)^{\alpha/2-1}$, which is the stated bound with χ_s.

Lastly, Theorem 3.7 gives $\mathrm{dep}(\widehat{f}_\varepsilon) \le C\log\varepsilon^{-1}$ and $W(\widehat{f}_\varepsilon) \le C\varepsilon^{-p_{\max}}\log\varepsilon^{-1}$. Therefore, after incorporating the ε lower bounds and using $\log\varepsilon^{-1} \le 2\chi\log\eta^{-1} + C$ we have the result. □

Corollary 5.7 (Structural Complexity, Block)**.** *For every $p > p_f$, the neural block in Theorem 5.6 may be chosen so that $W(\widehat{f}_\varepsilon) \le C_p\,\eta^{-p\chi_s}\log\eta^{-1}$ and $\mathrm{dep}(\widehat{f}_\varepsilon) \le C_p\log\eta^{-1}$ for $\alpha \ne 2$, with C_p depending on p and the quantities from Theorem 5.6. If the infimum for p_f is attained (e.g. polynomial governing equations), we can take $p = p_f$ and, in particular, $W(\widehat{f}_\varepsilon) \le C\log\eta^{-1}$ when $p_f = 0$ is attained.*

The proof, given in Appendix A.3, applies Theorem 5.6 to a representation G with $p_{\max}(G) < p$. The corollary realizes every exponent above p_f, which is defined independently of any architecture or discretization. The optimality of the exponent over the smooth classes is discussed in Remark 5.8 and Theorems 7.3 and 7.5.

Remark 5.8 (Block Lower Bounds on the Trainable Parameters)**.** *Over smooth function classes, the exponent $p_{\max} = 1/s$ of a C^s node cannot be improved. By the lower bound on continuous nonlinear widths from [19], any family of ReLU networks that approximates every function in the unit ball of $W^{s,\infty}(I)$ up to an accuracy of ε and whose weights depend continuously on the target function, requires at least $c\,\varepsilon^{-1/s}$ weights.*

The bound applies to our construction in Remark B.2, which attains the rate of Theorem B.1 with weights that depend linearly on the Taylor series terms for g. This implies that the exponent $1/s$ is optimal for continuous weights. Without continuity, the VC-dimension bound of [86] gives a lower bound of $c\,\varepsilon^{-1/(2s)}$ up to logs, and for $s = 1$ this is attained by deep networks whose weights are not continuous with respect to g [87]. These are worst-case statements over the class and do not contradict the improvements for polynomial governing laws.

5.3 Long Time Horizons

The contraction arguments shrink the time window to $[0, T]$ with $T \le T_0$. If we want a prediction up to $T_{\mathrm{final}} = JT$, we effectively can run the same (pretrained) network on J consecutive time windows, where the output of one interval becomes the input to the next. A block-wise rollout of the same learned model was also analyzed in [75]. If Θ_{win} is the neural network amplification factor for each window's error, summing them leads to $\eta_{\mathrm{win}}(1 + \Theta_{\mathrm{win}} + \cdots + \Theta^{J-1}_{\mathrm{win}})$, where η_{win} is the error of one window started from exact data. Since J and Θ_{win} do not depend on η, the final time T_{final} just changes the constant.

We can write this as a time-stepper over the windows. For $c \in \mathbb{R}^N$ let $G_N(c) := \mathcal{C}^h_N\left[\mathcal{F}^h(\mathcal{S}^h_N c, \nabla_h \mathcal{S}^h_N c)\right]$. For c with $\|c\|_{\ell^2_\alpha} + CC_{\mathrm{stab}}F_R T^\theta \le R_0$, let $x(c) = (x^0(c), \ldots, x^Q(c))$ be the fixed point of $\Psi^h_{N,Q}$ with a^h_0 replaced by c (Theorem 4.20), and let $\Phi_{\mathrm{win}}(c) := x^Q(c)$ be its final value within the window (the finer time scale). Since $S_N(t_q - t_r) = (S_N(\Delta t))^{q-r}$, the fixed point solves the recursion

$$x^0(c) = c, \qquad x^{q+1}(c) = S_N(\Delta t)\,(x^q(c) + \Delta t\, G_N(x^q(c)))\,. \tag{19}$$

Defining the discrete inner product by $\langle a, b\rangle_s := \sum_k \lambda_k^s a_k b_k$, the spectral theory gives

$$\|S_N(t)c\|_{\ell^2_s} \le e^{-\lambda_1 t}\|c\|_{\ell^2_s}, \qquad \|S_N(t)c\|_{\ell^2_{s+r}} \le c_\alpha\, t^{-r/2}\|c\|_{\ell^2_s}, \tag{20}$$

for $0 \le r \le \alpha$, where $c_\alpha := \max\{1, (\alpha/2e)^{\alpha/2}\}$. Also, for any grid function $v \in \mathbb{R}^M$ and $z \in \mathbb{R}^N$,

$$\langle \mathcal{C}_N^h v, z\rangle_0 = Q_{N,h}(v, \mathcal{S}_N z), \tag{21}$$

which maps point-wise inequalities on f (evaluated at the spatial grid points) into inequalities for G_N. For two initial vectors a, b we can write the differences as $z^q := x^q(a) - x^q(b)$ and $d^q := G_N(x^q(a)) - G_N(x^q(b))$.

Proposition 5.9 (Continuation Over Windows)**.** *Set $T \le T_0$, $(N,h) \in \mathcal{A}$, Q, n and $\widehat{f_\varepsilon}$ to have same values over all of the windows. Let u be the mild solution on $[0, JT]$, and set $u_j := u(jT)$ and $\beta_j := \mathcal{C}_N^h \mathcal{I}_h u_j$. Assume that the following holds.*

(i) Each of the functions u_j satisfies the conditions from Theorem 5.4 and Assumption 6.3 with the same constants, so the bound from Theorem 5.4 with data u_j is at most η_{win} (uniformly).

(ii) There is $\delta > 0$ such that the conditions from Theorem 5.4 on the initial coefficients hold for every vector c with $\|c - \beta_j\|_{\ell^2_\alpha} \le \delta$ for some j.

(iii) For some $s \in \{0, \alpha\}$ there are $\Theta_{\text{win}}, C_{\text{lift}} \ge 0$ such that, for coefficients c, c' δ close to β_j in ℓ^2_α

$$\|\Phi_{\text{win}}(c) - \Phi_{\text{win}}(c')\|_{\ell^2_s} \le \Theta_{\text{win}}\|c - c'\|_{\ell^2_s}, \qquad \|\Phi_{\text{win}}(c) - \Phi_{\text{win}}(c')\|_{\ell^2_\alpha} \le C_{\text{lift}}\|c - c'\|_{\ell^2_s}.$$

Let $b_0 := \beta_0$, let b_{j+1} be the terminal value of the n-th neural Picard iterate on window j started from b_j, and set $\tilde{u}_j := \mathcal{S}_N b_j$ and $\bar{\eta} := \eta_{\text{win}} + \max_j \varepsilon^{\text{enc}}_{N,h}(u_j)$. If $\bar{\eta}\left(1 + C_{\text{lift}}\lambda_1^{-(\alpha-s)/2}\sum_{i<J}\Theta^i_{\text{win}}\right) \le \delta$, then for $1 \le j \le J$,

$$\|\tilde{u}_j - u_j\|_{X^\alpha} \le \eta_{\text{win}} + C_{\text{lift}}\lambda_1^{-(\alpha-s)/2}\,\bar{\eta}\sum_{i<j-1}\Theta^i_{\text{win}}.$$

The proof is given in Appendix A.3. Condition (iii) always holds with $s = \alpha$ and $\Theta_{\text{win}} = C_{\text{lift}} = 1/(1-\kappa)$. When the governing law has an energy estimate it holds with $s = 0$ and $\Theta_{\text{win}} = e^{\psi T}$ (see for example Lemma 5.10 and Appendix A.6 for general laws).

Lemma 5.10 (Windows for the Allen-Cahn System)**.** *Let $\Omega = \mathbb{T}$, $L = -\partial_{xx} + I$, $f(u) = 2u - u^3$, $\alpha = \frac{7}{4}$, and assume the quadrature is the standard trapezoidal rule. Suppose the grid values of $\mathcal{S}_N x^q(a)$ and $\mathcal{S}_N x^q(b)$ are in the interval $[-m_K, m_K]$ for all q, the Lipschitz constant of f on $[-m_K, m_K]$ is $\Gamma_0 := \max\{2, 3m_K^2 - 2\}$, and assume that $\Delta t\,\Gamma_0^2 \le 1$. Then the following holds.*

$$\|z^q\|_{\ell^2_0} \le e^{3t_q/2}\|a - b\|_{\ell^2_0}, \qquad \|\Phi_{\text{win}}(a) - \Phi_{\text{win}}(b)\|_{\ell^2_\alpha} \le \left(T^{-7/8} + 8\Gamma_0 e^{3T/2}T^{1/8}\right)\|a - b\|_{\ell^2_0}.$$

Proof. Fix the (finer) time-stamp q. Let v_i and w_i be the grid values of $\mathcal{S}_N x^q(a)$ and $\mathcal{S}_N x^q(b)$, so that $v_i - w_i$ are the grid values of $\mathcal{S}_N z^q$. Since the trapezoidal rule is exact on products of retained modes, we have $\sum_i \omega_i |v_i - w_i|^2 = \|z^q\|^2_{\ell^2_0}$ and $\|\mathcal{C}_N^h g\|_{\ell^2_0} \le \left(\sum_i \omega_i |g_i|^2\right)^{1/2}$ for every grid function g.

By (21), we have

$$\langle d^q, z^q\rangle_0 = \sum_i \omega_i \left(f(v_i) - f(w_i)\right)(v_i - w_i) = \sum_i \omega_i \left(2|v_i - w_i|^2 - (v_i^3 - w_i^3)(v_i - w_i)\right) \le 2\|z^q\|^2_{\ell^2_0},$$

using monotonicity $(v^3 - w^3)(v - w) \ge 0$. Since $|f(v_i) - f(w_i)| \le \Gamma_0|v_i - w_i|$, we have $\|d^q\|_{\ell^2_0} \le \Gamma_0\|z^q\|_{\ell^2_0}$. Using the assumption $\Gamma_0^2\Delta t \le 1$,

$$\|z^q + \Delta t\, d^q\|^2_{\ell^2_0} \le \left(1 + 4\Delta t + \Gamma_0^2\Delta t^2\right)\|z^q\|^2_{\ell^2_0} \le e^{5\Delta t}\|z^q\|^2_{\ell^2_0}.$$

Using (19) and (20) with $\lambda_1 = 1$, $\|z^{q+1}\|_{\ell_0^2} \le e^{-\Delta t} e^{5\Delta t/2} \|z^q\|_{\ell_0^2} = e^{3\Delta t/2} \|z^q\|_{\ell_0^2}$, and induction gives the first bound.

Iterating the stepper forward (19) gives $z^Q = S_N(T) z^0 + \Delta t \sum_{p<Q} S_N(T - t_p) d^p$. By (20) with $r = \alpha = \frac{7}{4}$, where $c_\alpha = 1$ since $\alpha < 2e$, and by $\|d^p\|_{\ell_0^2} \le \Gamma_0 \|z^p\|_{\ell_0^2} \le \Gamma_0 e^{3T/2} \|z^0\|_{\ell_0^2}$,

$$\|z^Q\|_{\ell_\alpha^2} \le \left(T^{-7/8} + \Gamma_0 e^{3T/2} \, \Delta t \sum_{p<Q} (T - t_p)^{-7/8} \right) \|z^0\|_{\ell_0^2} \le \left(T^{-7/8} + 8\Gamma_0 e^{3T/2} T^{1/8} \right) \|z^0\|_{\ell_0^2},$$

using the integral bound $\Delta t \sum_{p<Q} (T - t_p)^{-7/8} \le \int_0^T s^{-7/8} \, ds = 8T^{1/8}$.

□

The window length is set by the contraction, so $J \propto T_{\text{final}} (C_{\text{stab}} L_R)^{1/\theta}$, which is independent of N, h, Q and η. The final accuracy η requires $\bar{\eta} \le \eta / A_J$ with $A_J := 1 + C_{\text{lift}} \lambda_1^{-(\alpha - s)/2} \sum_{i<J} \Theta_{\text{win}}^i$, so T_{final} changes the constant but not the exponent of η. The contraction gives $\Theta_{\text{win}} = 1/(1 - \kappa) \le 2$ for $\kappa \le \frac{1}{2}$, so $A_J \lesssim J 2^J = J e^{(\log 2/T) T_{\text{final}}}$. An energy estimate gives $\Theta_{\text{win}} = e^{\psi T}$, so $A_J \lesssim J e^{\psi_+ T_{\text{final}}}$. Both bounds can be written as the form $A_J \lesssim J e^{\rho T_{\text{final}}}$, but the contraction rate $\rho = \log 2/T$ grows as the windows shrink, while the energy rate $\rho = \psi_+$ is set by the law and is independent of the window length. For Allen-Cahn, this is a difference of $\psi = \frac{3}{2}$ versus $\log 2/T \approx 6 \times 10^3$, for a single-window.

5.4 Realization of the Picard Sweep

While Theorem 5.6 counts the number of trainable parameters for the neural Picard map, here we count every parameter for the entire architecture. This includes the fixed transforms, which are assumed to be fast (Assumption 5.11), and the Duhamel step as a linear recurrence with diagonal coefficient. Specifically, since $S_N(t_q - t_r) = S_N(\Delta t)^{q-r}$, the iterates $v^q := \left(\Psi^h_{N,Q,\varepsilon}(a) \right)^q$ satisfy

$$v^0 = a_0^h, \qquad v^{q+1} = S_N(\Delta t) \left(v^q + \Delta t \, \mathcal{C}_N^h g^q \right), \qquad q = 0, \dots, Q - 1, \tag{22}$$

where $g^q := \widehat{\mathcal{F}}_\varepsilon^h \left(\mathcal{S}_N^h a^q, \nabla_h \mathcal{S}_N^h a^q \right)$. Since the multipliers $e^{-\lambda_k \Delta t}$ are fixed, each step of the recurrence is an affine map with fixed coefficients, and composition is associative. So a parallel scan [5] computes all $v^0, \dots, v^Q$ in depth $O(\log Q)$ using $O(Q \log Q)$ weights per mode, instead of the naive depth Q.

Assumption 5.11 (Fast transforms)**.** *There is a function $\mathcal{T}(M)$ and a constant C_{ft} such that each fixed transform $\mathcal{T}_{\text{syn}}$, $\mathcal{T}_{\text{syn},j}$ and $\mathcal{T}_{\text{an}}$ of Assumption 4.5 factors into $C_{\text{ft}} \log M$ sparse layers with at most $\mathcal{T}(M)$ nonzero weights in total.*

If we can use the FFT then the assumption holds with $\mathcal{T}(M) = CM \log M$ for the trigonometric basis and with $\mathcal{T}(M) = CM \log^2 M$ for classical orthogonal polynomial and spherical harmonic bases [20, 79]. It holds trivially with $\mathcal{T}(M) = NM$ and one layer, which recovers the dense transform case.

Definition 5.12 (Realization of Picard Sweeps)**.** *Fix the parameters $(N, h) \in \mathcal{A}$, Q, n and $\widehat{f}_\varepsilon$. For $w \in \mathbb{R}^{dM}$, the sampled data $w = \mathcal{I}_h u_0$, set $a_0^h(w) := \mathcal{C}_N^h w$ and $A^0(w) := \left(a_0^h(w), \dots, a_0^h(w) \right) \in (\mathbb{R}^{dN})^{Q+1}$, and let $A^k(w) := \Psi^h_{N,Q,\varepsilon} \left(A^{k-1}(w) \right)$ for $k = 1, \dots, n$, with initial data $a_0^h(w)$. The n-sweep network can be realized by $\mathcal{A}_n(w) := A^n(w)$, the loop of Definition 4.7 repeated n times.*

Theorem 5.13 (Size and depth of the n-sweep Network)**.** *Assume the conditions from Theorem 5.6 and Assumption 5.11, pick $(N, h, Q, n, \varepsilon)$ and $\widehat{f}_\varepsilon$ accordingly, and assume that the sampled input satisfies*

$$\left\| \mathcal{C}_N^h \mathcal{I}_h u_0 \right\|_{\ell_\alpha^2} \le R_0, \tag{23}$$

which holds when $C_{\text{stab}}^\alpha \|u_0\|_{X^\alpha} \le R_0$ (Lemma 4.17). Then $\mathcal{A}_n$ is a ReLU network with the following properties:

1. *The accuracy is given by* $\left\| \mathcal{L}^S_{N,Q}\, \mathcal{A}_n\, (\mathcal{I}_h u_0) - u \right\|_{X^\alpha_T} \le \eta$.

2. *The depth is given by* $\operatorname{dep}(\mathcal{A}_n) \le n \left(\operatorname{dep}(\widehat{f}_\varepsilon) + C \log(MQ) \right) \le C \left(\log \eta^{-1} \right)^2$.

3. *The size is*

$$W(\mathcal{A}_n) \le C\, n \left[(2 + d_x) d\, Q\, \mathcal{T}(M) + d\, N\, Q \log(2Q) + MQ\, W(\widehat{f}_\varepsilon) \right] + C\, \mathcal{T}(M),$$

where $M = M(h)$. *If* $N \le CM$ *and* $\mathcal{T}(M) \le CM \log M$, *then we can write it as*

$$W(\mathcal{A}_n) \le C\, n \underbrace{MQ}_{\text{space–time points}} \Big[\underbrace{(2 + d_x)\, d \log(MQ)}_{\text{transforms and scan}} + \underbrace{W(\widehat{f}_\varepsilon)}_{\text{block}} \Big]; \tag{24}$$

4. *The number of trainable parameters is given by* $W(\widehat{f}_\varepsilon) + W_{\text{spec}}$ *which are the distinct and trainable weights.*

The proof is given in Appendix A.4. The next theorem obtains the explicit rates by combining the results from Theorems 5.6 and 5.13.

Theorem 5.14 (Rates and Total Cost)**.** *Assume the conditions from Theorem 5.13,* $c_W k^{2/d_x} \le \lambda_k \le C_W k^{2/d_x}$ *for all* k, $M(h) \asymp h^{-d_x}$, $\mathcal{T}(M) \le CM \log M$, *and the resolution condition* $(N, h) \in \mathcal{A}$ *if* $h \lambda_N^{1/2} \le c_{\text{res}}$. *Let* $u_0 \in X^{\gamma_s}$. *Using Theorem 5.6 and a quadrature of order* $r \le \gamma_s - \alpha$, *set*

$$v := \frac{d_x}{r} + \frac{1}{\zeta_t} + p_{\max} \chi_s.$$

Then the architecture from Definition 4.7 approximates the solution up to accuracy η *in* $C([0, T]; X^\alpha)$ *with the following cost (see Remark 5.15 for higher order operators), where* C *is independent of* η, *for* $\alpha \ne 2$ *(for* $\alpha = 2$, *every bound involving the block includes a factor* $(\log \eta^{-1})^{p_{\max}}$*):*

Quantity	*Upper Bound*
discretization	$N \le C\eta^{-d_x/(\gamma_s - \alpha)}, \quad M \le C\eta^{-d_x/r}, \quad Q \le C\eta^{-1/\zeta_t}, \quad n \le C \log \eta^{-1}$
trainable block	$W(\widehat{f}_\varepsilon) \le C\eta^{-p_{\max}\chi_s} \log \eta^{-1}$
trainable, spectral	$W_{\text{spec}} \le C\eta^{-d_x/(\gamma_s - \alpha)}$
All Weights (per eval.)	$\le C\eta^{-v} (\log \eta^{-1})^2$
Total Depth	$\operatorname{dep}(\mathcal{A}_n) \le C (\log \eta^{-1})^2$

The trainable parameter count is independent of $M(h)$, Q *and* n.

Note that when $r = \gamma_s - \alpha$ we have $M \asymp N$.

Remark 5.15 (Higher-order Operators)**.** *Theorem 5.14 assumes Weyl's law and the resolution condition associated with second-order linear operators. Suppose* L *has order* $2m$. *If* $c_W k^{2m/d_x} \le \lambda_k \le C_W k^{2m/d_x}$ *and the resolution condition is* $h \lambda_N^{1/(2m)} \le c_{\text{res}}$, *then the same proof gives* $N \asymp \eta^{-d_x/(m(\gamma_s - \alpha))}$. *With an exact quadrature* $M \asymp N$, *so the spatial part of the exponent* v *becomes* $\frac{d_x}{m(\gamma_s - \alpha)}$. *Thus* m *is included in the rates.*

Proof of Theorem 5.14. The proof collects all the bounds together. The accuracy and the block-wise bound from Theorem 5.6 (where we absorb m_0 into the constant) and the trainable parameters come from $\widehat{f}_\varepsilon$ and the spectral weights $W_{\text{spec}} \le C_{\text{spec}} N$ (Proposition 4.10).

The additional assumptions regarding the parameters make certain terms explicit. Following the proof of Theorem 5.6, we have N'' and $\Lambda := (m_0/(c_{\text{err}}\eta))^{1/\zeta_s}$, where $N \le N''$. If $N'' \ge 2$, the lower Weyl bound in the assumptions gives $c_W (N'')^{2/d_x} \le \lambda_{N''} < \Lambda$, so $N'' < (\Lambda/c_W)^{d_x/2}$ and $\Lambda^{d_x/2} = (m_0/(c_{\text{err}}\eta))^{d_x/(\gamma_s - \alpha)}$. By Theorem 5.6 (iv) we have $Q \le 2T(c_{\text{err}}\eta)^{-1/\zeta_t}$ and $n \le C \log \eta^{-1}$ by (i).

Next we look at M. All conditions in Theorem 5.6(iii) hold for every $h \le h_* := c\,(c_{\mathrm{err}}\eta)^{1/r}$, with $c \in (0,1]$ depending only on m_0, λ_1, c_{res} and C_{quad}. Specifically, $h^r \le c_{\mathrm{err}}\eta$ since $c \le 1$. Next, $\varepsilon^{\mathrm{enc}}_{N,h}(u_0) \le C_{\mathrm{quad}}h^r\|u_0\|_{X^{\alpha+r}} \le Cm_0h^r \le c_{\mathrm{err}}\eta$ for $c \le (Cm_0)^{-1/r}$, since $u_0 \in X^{\gamma_{\mathrm{s}}} \subset X^{\alpha+r}$. Lastly, $(N,h) \in \mathcal{A}$ when the resolution condition holds $h\lambda_N^{1/2} \le c_{\mathrm{res}}$, and since $\lambda_N \le \max\{1,\lambda_1\}\Lambda$ gives $\lambda_N^{1/2} \le C(c_{\mathrm{err}}\eta)^{-1/(\gamma_{\mathrm{s}}-\alpha)}$, this holds for $h \le h_*$ because $r \le \gamma_{\mathrm{s}} - \alpha$ and $c_{\mathrm{err}}\eta < 1$. Since the choice $h = 2^{-k}$ in (iii) stops no later than the first power of two under h_*, we have $h > h_*/2$, and thus $M \le Ch^{-d_x} \le C\eta^{-d_x/r}$. Conversely, $h \le (c_{\mathrm{err}}\eta)^{1/r}$ gives $M \ge c\eta^{-d_x/r} \ge cN$, since $N \le C\eta^{-d_x/(\gamma_{\mathrm{s}}-\alpha)} \le C\eta^{-d_x/r}$, so the size from (24) holds.

Putting these bounds together in (24) yields

$$W(\mathcal{A}_n) \le C\log\eta^{-1}\cdot\eta^{-d_x/r-1/\zeta_{\mathrm{t}}}\left(\log\eta^{-1} + \eta^{-p_{\max}\chi_{\mathrm{s}}}\log\eta^{-1}\right) \le C\eta^{-v}(\log\eta^{-1})^2,$$

and the depth comes from Theorem 5.13(ii). The cost of evaluating a ReLU network is a finite number of operations per nonzero weight and per nodes, and there are at most as many units as nodes. With $r = \gamma_{\mathrm{s}} - \alpha$, the upper bounds on N and M have matching rates, and the upper Weyl bound assumption gives the matching lower bound on N, so $M \asymp N$. □

Remark 5.16 (Error in L^2)**.** *Since $\|v\|_{L^2} \le \lambda_1^{-\alpha/2}\|v\|_{X^\alpha}$, all of our bounds in $C([0,T];X^\alpha)$ give a bound in $C([0,T];L^2)$ directly, that is, with the same exponents and a constant multiplied by $\lambda_1^{-\alpha/2}$. The lower bounds of Section 7 allow any $\alpha \ge 0$, so they hold for L^2. Repeating the proofs, using the weaker L^2 norm should remove most of the amplification effects, since $\xi_{\alpha'} = 1$ for $\alpha' < 2$, and change the mode count to $\eta^{-d_x/\gamma_{\mathrm{s}}}$, matching Proposition 7.6 at $\alpha = 0$. We leave this to a future work.*

5.5 Rate Comparisons with Other Approximations

We can compare these rates to two other approximations directly. For simplicity and fairness, the only difference between the approximations is in the nonlinear terms and we assume $\alpha > 2$, so that $\varepsilon \asymp \eta^\chi$ is needed in our approach (Table 2). First, if we consider the Picard map as a generic Lipschitz operator [52, 45, 36, 81], then resolving the trajectories from Lemma 2.7 requires a latent space with $n(\eta) \to \infty$, and so the exponential cost cannot be improved over the Lipschitz class [42]. To improve the rate, one would need holomorphic or structured operators, which is not given by the derived regularity of the system Lemma 2.7. In another approach, one can consider a standard approximation to f on $\mathbb{R}^D$, $D = d(1+d_x)$ [86]. When $s \le s_{\min}$, the common setting, our DAG formulation improves the exponent by a factor of at least D (e.g., 6 for the 2D Navier-Stokes, 2 for a one-dimensional conservation law), and for polynomial f it leads to a larger improvement, replacing inverse power laws on η by $\log\eta^{-1}$ only. The intrinsic cost of resolving the state variable (the factor $\frac{d_x}{r} + \frac{1}{\zeta_{\mathrm{t}}}$) is used in all three approaches, but our formulation reduces the trainable components and leverages the parabolic structure directly.

Method	Dimension of nonlinear approx.	Trainable size for accuracy η
1. Black-box learning of Φ	$n(\eta) \asymp \eta^{-\frac{d_x+2}{\gamma-\alpha}} \to \infty$	$\exp\left(c\,\eta^{-\frac{d_x+2}{\gamma-\alpha}}\log\eta^{-1}\right)$
2. Locality + direct ReLU for f	$D = d(1+d_x)$, fixed	$C\,\eta^{-\frac{D}{s}\chi}\log\eta^{-1}$
3. Locality + DAG (Thm. 5.6)	1, at each nonlinear node	$C\,\eta^{-\frac{1}{s_{\min}}\chi}\log\eta^{-1}$ ($C\log\eta^{-1}$ if multilinear)

Table 2: Comparison between general and neural Picard theory. First, using the knowledge that the system is a semilinear parabolic PDE reduces the dimension of the nonlinear approximation from $n(\eta)$ to the state dimension $D = d(1+d_x)$. Then our DAG approximation reduces it from D to one variable per node. The discretization and the spectral weights are the same in all three rows.

5.6 Fixed Linear Operations

In some settings, the governing law f involves a nonlocal but fixed, linear operation. We show that such systems do not change the overall costs. This was referenced in Section 3 and used in Section 7.1 and in the

explicit PDE examples of Section 8. Since the linear term may change the spaces, we need to introduce an extension.

Lemma 5.17 (Fixed Linear Operations). *Let $\widetilde{H} \supseteq H$ be a Hilbert space associated with a self-adjoint extension of L that has the e_k as its eigenfunctions and let $\widetilde{X}^s$ be the induced spaces with $X^s = \widetilde{X}^s \cap H$. Let $\mathcal{M} : \widetilde{X}^\beta \to X^\beta$ be a linear operator such that*

$$\mathcal{M}^* e_k = \mu_k e_k, \qquad C_{\mathcal{M}} := \sup_{k \geq 1} |\mu_k| < \infty, \tag{25}$$

and let $f = \mathcal{M} \circ f_{\mathrm{loc}}$. We assume that f_{loc} is local, satisfies Assumption 3.5, and is Lipschitz on bounded sets from X^α into $\widetilde{X}^\beta$ with constant L_R^{loc}. If $D_\mu := \mathrm{diag}(\mu_1, \ldots, \mu_N)$, then the following hold.

1. *$\mathcal{C}_N \mathcal{M} = D_\mu \mathcal{C}_N$ on $\widetilde{X}^\beta$, D_μ commutes with $S_N(t)$, and $\|D_\mu\|_{\ell^2_s \to \ell^2_s} \leq C_{\mathcal{M}}$ for every s;*
2. *If $\mathcal{C}_N^h$ is replaced by $D_\mu \mathcal{C}_N^h$ in Definition 4.4, and Assumption 4.16 is replaced by $\widetilde{X}^s$, then Theorems 4.20, 4.21, 4.22, 5.4, and 5.6 hold but with $L_R = C_{\mathcal{M}} L_R^{\mathrm{loc}}$ and $F_R = C_{\mathcal{M}} F_R^{\mathrm{loc}}$.*
3. *The trainable parameters are from approximating f_{loc} and the spectral weights Definition 4.6, which includes the entries of D_μ (the $\mathbf{1}_{\mathcal{M}}$ term). In particular, if $\mathcal{M}$ is the orthogonal projection of $\widetilde{H}$ onto H, then $\mu_k \equiv 1$, $D_\mu = I$, and $\mathcal{M}$ requires no layers.*

The proof is given in Appendix A.4.

6 The Solution Operator on Compact Sets of Initial Data

Thus far, Theorems 5.6 and 5.13 are results concerning the approximation of a single trajectory of the governing system starting from a fixed u_0. The data (initial condition) is input to the architecture through the encoder $w \mapsto \mathcal{C}_N^h w$ on the sampled function (vector) $w = \mathcal{I}_h u_0$. In operator learning, one aims to construct an approximation over a set of inputs and obtain a uniform accuracy for all inputs over, say, set $\mathcal{U}$. In this section, we show that the architecture with the same trainable parameters achieves this.

On bounded subsets of X^γ the regularity from prior sections is sufficient to prove that the system approximates the general solution operator. Every constant in the analysis depends only on u_0 via $\|u_0\|_{X^\gamma}$, so Theorem 5.6 is already uniform and the rates are unchanged (summarized in Theorem 6.5). On the other hand, if we just have sets that are compact in X^α, we get a uniform approximation but without an explicit rate, see also [13, 40].

6.1 Uniform Properties

We denote the solution operator by $G_T : u_0 \mapsto u$, see Definition 2.4, and denote the set of initial conditions by $\mathcal{U}$.

Lemma 6.1 (Uniform Well-posedness on Bounded Sets). *For $R_0 > 0$, we set $R := 1 + 2R_0$. Using the conditions from Proposition 2.5, there exists $T_0 > 0$, depending only on R, on L_R, and on the semigroup constant, such that for every $T \in (0, T_0]$ and every $u_0 \in X^\alpha$ with $\|u_0\|_{X^\alpha} \leq R_0$:*

1. *Φ maps B_R^α into itself and is a contraction with constant $\kappa_0 = CT^\theta L_R \leq \frac{1}{2}$;*
2. *$G_T[u_0]$ is well defined on B_R^α.*

Both R and κ_0 are independent of u_0.

The proof is given in Appendix A.5.

Lemma 6.2 (Compactness of the regularity ball). *For $\gamma > \alpha$ the embedding $X^\gamma \hookrightarrow X^\alpha$ is compact, in particular, the ball $\mathcal{U}_{R_0} := \{u_0 \in X^\gamma : \|u_0\|_{X^\gamma} \leq R_0\}$ is a compact subset of X^α.*

The proof is given in Appendix A.5.

To extend the results from Section 5, we need the conditions to hold uniformly over the input data set. The parabolic estimates must be uniform on bounded sets of data.

Assumption 6.3 (Uniform Estimates)**.** *For every $\rho_\gamma > 0$ there are constants R_γ and C_R, only depending on ρ_γ, T, and the constants from Propositions 2.5 and 2.6, such that every u_0 with $\|u_0\|_{X^\gamma} \le \rho_\gamma$ satisfies the same result from Lemma 2.7 with*

$$\sup_{0<t\le T} t^\zeta \|u(t)\|_{X^\gamma} \le R_\gamma, \qquad \|F\|_{\infty,\beta} + [F]_\zeta \le C_R,$$

and with the bound from Theorem 4.22 on $\|f(P_N u, \nabla P_N u)\|_{X^{\beta+r}}$ holding with the same C_R.

The assumption is a summary of previous bounds. The first two bounds follow from Lemma 2.7, whose constants $R_\gamma = C(R + T^\theta F_R)$ and $[F]_\zeta \le C L_R(R_\gamma + T^\theta F_R)$ depend on u_0 only through the radius R of Proposition 2.5. Thus it holds by a uniform bound on $\|u_0\|_{X^\alpha}$. The bound on $f(P_N u, \nabla P_N u)$ follows from Assumption 2.8 with $s = \alpha + r$, with constants only depending on K_R, R and R_γ. The bootstrap shows that the same holds when we replace X^γ by $X^{\gamma'}$, the constants then also depending on the radius $R_{\gamma'}$. The radius ρ_γ is only needed for E^h_{init} and $\|(I - P_N)u_0\|_{X^\alpha}$ to be uniformly controlled over the data set.

The architecture sees the data only by its sampled values $\mathcal{I}_h u_0$ and reconstructs $\Pi_{N,h} u_0 = \mathcal{S}_N \mathcal{C}_N^h \mathcal{I}_h u_0$ in E_N. Over a set $\mathcal{U}$, the errors are $\varepsilon_N^{\text{tail}}(\mathcal{U})$, which measures how well the eigenbasis resolves $\mathcal{U}$, and $\varepsilon_{N,h}^{\text{enc}}(\mathcal{U})$, which measures the error of using quadrature instead of inner products (see Definition 5.2). For regular data, we will get a rate. Using just compactness, we can only show they tend to zero. In this subsection we assume, in addition to Assumptions 4.16 and 4.18, the admissible refinement condition:

$$\text{for every } N \in \mathbb{N} \text{ there is } h_N > 0 \text{ with } (N,h) \in \mathcal{A} \text{ for all } h \in (0, h_N]. \tag{26}$$

For a set $\mathcal{U} \subset X^\alpha$ of data, we extend the previous tail and encoding definitions to $\varepsilon_N^{\text{tail}}(\mathcal{U}) := \sup_{u_0\in\mathcal{U}} \varepsilon_N^{\text{tail}}(u_0)$ and $\varepsilon_{N,h}^{\text{enc}}(\mathcal{U}) := \sup_{u_0\in\mathcal{U}} \varepsilon_{N,h}^{\text{enc}}(u_0)$.

Lemma 6.4 (Uniform Encoding)**.** *Let $\mathcal{V} \subset X^\alpha$ be compact and let N be fixed. Then, by* (26)*,*

$$\sup_{v\in\mathcal{V}} \left\| \left(\mathcal{C}_N^h \mathcal{I}_h - \mathcal{C}_N \right) v \right\|_{\ell^2_\alpha} \to 0 \qquad \text{as } h \to 0,$$

for $(N,h) \in \mathcal{A}$. In particular, $\varepsilon_{N,h}^{\text{enc}}(\mathcal{U}) \to 0$ for every compact $\mathcal{U} \subset X^\alpha$, where we set $\mathcal{V} = \{S(t)u_0 : u_0 \in \mathcal{U},\ 0 \le t \le T\}$.

The set $\mathcal{V} = \{S(t)u_0 : u_0 \in \mathcal{U},\ 0 \le t \le T\}$ is compact since it is the continuous image of $\mathcal{U} \times [0,T]$. The proof is given in Appendix A.5.

6.2 Uniform Rates from Regularity

Theorem 6.5 (Operator Learning Error Rate)**.** *Let $R_0, \rho_\gamma > 0$ and take the initial data to be in*

$$\mathcal{U} := \{u_0 \in X^\gamma :\ \|u_0\|_{X^\alpha} \le R_0,\ \|u_0\|_{X^\gamma} \le \rho_\gamma\}.$$

Assume the conditions from Theorem 5.6 together with Assumption 6.3 hold. Let $T \le T_0$ be from Lemma 6.1 with the radius $\max\{1, C^\alpha_{\text{stab}}\} R_0$.

Then for every sufficiently small $\eta > 0$ there is a (N,h,Q,n,ε) and a block $\widehat{f}_\varepsilon$ with

$$\operatorname{dep}(\widehat{f}_\varepsilon) \le C \log \eta^{-1}, \qquad W(\widehat{f}_\varepsilon) \le C\, \eta^{-p_{\max}\chi} \log \eta^{-1},$$

(if $\alpha = 2$, include the factor $(\log \eta^{-1})^{p_{\max}}$) such that the full neural network $\mathcal{A}_n$ from Definition 5.12 satisfies

$$\sup_{u_0\in\mathcal{U}} \left\| \mathcal{L}^S_{N,Q}\, \mathcal{A}_n\, (\mathcal{I}_h u_0) - G_T[u_0] \right\|_{X^\alpha_T} \le \eta,$$

where the size and depth bounds from Theorem 5.13 hold. The constants depend on R_0 and ρ_γ and not on η.

The proof is given in Appendix A.5.

Theorem 6.5 can be directly compared to the operator-learning literature, where one architecture, with $C\eta^{-p_{\max}\chi} \log \eta^{-1}$ block parameters and $C_{\text{spec}} N$ spectral weights is used for all initial data in a bounded set of X^γ. Note that X^γ-boundedness is the only assumption used for the initial data, in particular, we do not require existence of a low-dimensional manifold or the span of finitely many terms.

7 Sharpness and Lower Bounds

We will consider several results regarding the sharpness of previous results.

7.1 Sharpness of Amplification Factor

First, we will address the neural network amplification factor ξ_α, which exceeds 1 only when $\alpha > 2$. That setting is needed for the embedding $X^\alpha \hookrightarrow C^1$ and thus $\alpha > 1 + \frac{d_x}{2}$ (so $\alpha > 2$ when $d_x \geq 2$). We show that the estimate is sharp, that smoother data reduce it, and that it is absent for gradient-free laws when $d_x \leq 3$.

The problem comes from the following balance. The Duhamel operator can only improve smoothness by fewer than two derivatives. The network error is controlled in ℓ^∞, and so after applying $\mathcal{C}_N^h$, the integrand input to the Duhamel operator is only in X^0. An output in X^α with $\alpha \geq 2$ therefore requires more smoothing than the Duhamel integral can produce. What appears is the Galerkin truncation giving the finite factor ξ_α. We could fix it by asking for $(f - \widehat{f}_\varepsilon)(u, \nabla u)$ in X^β, but that changes how learning is typically performed and would not be consistent with the practice. The following result shows that this is a sharp obstacle.

Proposition 7.1 (The Amplification Estimate is Sharp)**.** *For $\alpha > 2$, suppose the quadrature is exact, i.e. $\mathcal{C}_N^h \mathcal{S}_N^h = I_{\mathbb{R}^N}$, with $\|e_k\|_{L^\infty} \leq \sqrt{2}$ (see Remark 4.19). Set $\Lambda_0 := \min\{\lambda_N, \Delta t^{-1}\}$, let $T \geq 2\Lambda_0^{-1}$, and suppose some eigenvalue is in the interval $[c_{\mathrm{sp}}\Lambda_0, \Lambda_0]$ for a fixed $c_{\mathrm{sp}} \in (0,1)$ (holds by Weyl). Then there exist grid functions $g^0, \dots, g^{Q-1}$ with $\max_r \|g^r\|_{\ell^\infty} \leq \varepsilon$ such that*

$$\left\| \Delta t \sum_{r=0}^{Q-1} S_N(t_Q - t_r)\, \mathcal{C}_N^h g^r \right\|_{\ell^2_{\vec{a}}} \geq c\, \varepsilon\, \Lambda_0^{\alpha/2-1},$$

where $c > 0$ depends only on α and c_{sp}. Therefore, no estimate that uses only the ℓ^∞-error of the neural network can remove ξ_α from Theorem 4.21.

Proof of Proposition 7.1. Pick a $k_0 \leq N$ so that $\Lambda := \lambda_{k_0} \in [c_{\mathrm{sp}}\Lambda_0, \Lambda_0]$, and then $\Lambda\Delta t \leq 1$ and $\Lambda T \geq 2c_{\mathrm{sp}}$. We can construct the grid functions by $g_i^r := \frac{\varepsilon}{\sqrt{2}} e_{k_0}(x_i)$ for every r (using the eigenfunctions). Then $\|g^r\|_{\ell^\infty} \leq \varepsilon$ and $\mathcal{C}_N^h g^r = \frac{\varepsilon}{\sqrt{2}} \mathbf{1}_{k_0}$ since the quadrature is exact on the eigenfunction. Recall that $S_N(t)$ is diagonal, which makes the LHS in the proposition statement equal to

$$\frac{\varepsilon}{\sqrt{2}}\, \Lambda^{\alpha/2}\, \Delta t \sum_{j=1}^{Q} e^{-\Lambda j \Delta t} = \frac{\varepsilon}{\sqrt{2}} \Lambda^{\alpha/2} \Delta t \frac{e^{-\Lambda\Delta t}\left(1 - e^{-\Lambda T}\right)}{1 - e^{-\Lambda\Delta t}} \geq \frac{\varepsilon}{\sqrt{2}}\, e^{-1} \left(1 - e^{-2c_{\mathrm{sp}}}\right) \Lambda^{\alpha/2-1},$$

where we used $1 - e^{-\Lambda\Delta t} \leq \Lambda\Delta t$, $e^{-\Lambda\Delta t} \geq e^{-1}$, and $\Lambda T \geq 2c_{\mathrm{sp}}$. The rest follows from $\Lambda \geq c_{\mathrm{sp}}\Lambda_0$. □

Improved regularity can help remove this factor. Since ξ_α is controlled by $\min\{\lambda_N, \Delta t^{-1}\}$, smoother data makes the exponent go from χ to χ_{s} by lowering the spatial scale $\lambda_N \asymp \eta^{-1/\zeta_{\mathrm{s}}}$ (see Theorem 5.6), and by Remark 4.2 it lowers χ to $1 + \frac{(\alpha-2)_+}{2\theta}$. Also, the condition $\alpha > 2$ comes from the gradient input term for f. For a gradient-free law $f = f(u)$, e.g. reaction-diffusion system, Assumption 2.2 can be weakened to $X^\alpha \hookrightarrow C^0(\overline{\Omega})$, so in dimensions $d_x \leq 3$ we can take $\alpha < 2$ and $\xi_\alpha = 1$.

7.2 Divergence Forms

In some settings, we can rewrite the convective terms in divergence form (see also [10]):

$$-(u \cdot \nabla)u = -\nabla \cdot (u \otimes u) \quad \text{for } \nabla \cdot u = 0, \qquad -F'(u)\partial_x u = -\partial_x F(u),$$

so that $f = \mathcal{M} f_{\mathrm{loc}}(u)$ where f_{loc} only depends on u and where $\mathcal{M}$ is linear of order one. The derivative then becomes a diagonal spectral layer, the block no longer needs to compute ∇u directly, and so we can weaken the embedding requirements to $X^\alpha \hookrightarrow C^0$. The additional cost is that the network error now propagates through that layer.

Proposition 7.2 (Divergence Form)**.** *Let $\widetilde{X}^s$ be the ambient space from Lemma 5.17, and let $\{\tilde{e}_k\}$ be an orthonormal family of eigenfunctions from the extended L with the same eigenvalues λ_k, i.e. $\sum_k \lambda_k^s |\langle g, \tilde{e}_k\rangle|^2 \le \|g\|^2_{\widetilde{X}^s}$. Let $\mathcal{M}$ be linear with $\mathcal{M}^* e_k = \mu_k \tilde{e}_k$ and $|\mu_k| \le C_{\mathcal{M}} \lambda_k^{1/2}$, and assume that $f = \mathcal{M} \circ f_{\text{loc}}$, where f_{loc} depends only on u, satisfies Assumption 3.5 with $K \subset \mathbb{R}^d$, and is Lipschitz on bounded sets from X^α into $\widetilde{X}^\alpha$ with Lipschitz constant L_R^{loc}. Set $\beta = \alpha - 1$ (so $\sigma = \theta = \frac{1}{2}$), and replace $\mathcal{C}_N^h$ by $D_\mu \widetilde{\mathcal{C}}_N^h$ where $D_\mu = \operatorname{diag}(\mu_1, \dots, \mu_N)$. Then the following hold.*

1. *f is Lipschitz on bounded sets from X^α into X^β with constant $C_{\mathcal{M}} L_R^{\text{loc}}$, so Proposition 2.5 applies.*
2. *The discrete smoothing estimate from Theorems 4.20, 4.21, and 4.22 holds with the updated form*

$$\left\| S_N(\tau) D_\mu \widetilde{\mathcal{C}}_N^h v \right\|_{\ell^2_\alpha} \le C\, C_{\mathcal{M}}\, C^\alpha_{\text{stab}} \tau^{-1/2} \|v\|_{\widetilde{X}^\alpha},$$

   where the quadrature assumptions are used only at $s = \alpha$ for functions in $\widetilde{X}^\alpha$.
3. *Assumption 2.2 can be weakened to $X^\alpha \hookrightarrow C^0$, only requiring that $\alpha > \frac{d_x}{2}$. The entries of D_μ are included in the spectral weights.*
4. *Theorem 4.21 holds with $\xi_\alpha(N, \Delta t)$ replaced by $\xi_{\alpha+1}(N, \Delta t)$, where the amplification is at most logarithmic when $\alpha = 1$.*

The proof is given in Appendix A.5. On the torus, $\mathcal{M} = -\nabla\cdot$ and so $\mathcal{M}^* e_k = \nabla e_k$, which is $|k|$ times a normalized mode, and thus the multiplier is $|\mu_k| = |k| \asymp \lambda_k^{1/2}$. Including this with the Leray projection does not change the result since $\mathbb{P}$ is an orthogonal projection and commutes with L. Since the amplification is $\xi_{\alpha+1}$, the best choice is to pick an α close to 1 within what the embedding $X^\alpha \hookrightarrow C^0$ allows. For the 2D Navier-Stokes equation, this is $\alpha = 1 + \iota$, with $\xi \asymp \lambda_N^{\iota/2}$, the same as in velocity form. The improvement is that the block only needs to compute u and the gradient transforms are no longer needed. For a viscous conservation law in one dimension, $\alpha = 1$ gives only a logarithmic amplification, and since $f_{\text{loc}} = F$ is a single component-wise node in the DAG, $p_{\max}$ improves from $\frac{1}{s-1}$ to $\frac{1}{s}$.

In the divergence form, the quadrature assumptions only appear when $s = \alpha$ and only through f_{loc}. The condition $\beta > \frac{d_x}{2}$ in Assumption 4.16(ii) is replaced by $\alpha > \frac{d_x}{2}$, and $\beta = \alpha - 1$ may be as small as 0. In Theorems 4.20 and 4.21 this corresponds to part (2) and part (4). In Theorem 4.22, the quadrature term becomes $d_l = D_\mu(\widetilde{\mathcal{C}}_N^h \mathcal{I}_h - \widetilde{\mathcal{C}}_N) f_{\text{loc}}(P_N u(t_l))$, and

$$\|d_l\|_{\ell^2_\beta} \le C_{\mathcal{M}} \left\| (\widetilde{\mathcal{C}}_N^h \mathcal{I}_h - \widetilde{\mathcal{C}}_N) f_{\text{loc}}(P_N u(t_l)) \right\|_{\ell^2_\alpha} \le C_{\mathcal{M}} C_{\text{quad}} h^r \, \|f_{\text{loc}}(P_N u(t_l))\|_{\widetilde{X}^{\alpha+r}},$$

so Assumption 4.18 is needed only when $s = \alpha$, and the weighted condition of that theorem applies to $f_{\text{loc}}(P_N u)$ in $\widetilde{X}^{\alpha+r}$. Similarly, Assumption 2.8 applies to the local part, $\|f_{\text{loc}}(v)\|_{\widetilde{X}^s} \le K_R(1 + \|v\|_{X^s})$ for $s \in [\alpha, \gamma']$, which implies it for f since $\|\mathcal{M} g\|_{X^{s-1}} \le C_{\mathcal{M}} \|g\|_{\widetilde{X}^s}$. In Lemma A.2, condition (2) holds with $\Gamma_\alpha = C_{\mathcal{M}} C^\alpha_{\text{stab}} L_R^{\text{loc}}$ since $\|D_\mu \widetilde{\mathcal{C}}_N^h v\|_{\ell^2_\beta} \le C_{\mathcal{M}} \|\widetilde{\mathcal{C}}_N^h v\|_{\ell^2_\alpha} \le C_{\mathcal{M}} C^\alpha_{\text{stab}} \|v\|_{\widetilde{X}^\alpha}$. Lastly, Theorem 5.6 can be applied where ξ_α is replaced by $\xi_{\alpha+1}$, so χ and χ_s are computed with $\alpha + 1$ instead of α. In particular $\chi_{\text{div}} = 1 + \frac{(\alpha-1)_+}{2\zeta}$ and there is an extra log term when $\alpha = 1$.

The divergence form now gives us that the block only needs to compute u, so its input dimension drops from $d(1 + d_x)$ to d and the gradient transforms are not needed. For non-polynomial fluxes the node is F rather than F' which improves $p_{\max}$ from $\frac{1}{s-1}$ to $\frac{1}{s}$ where $F \in C^2$ is sufficient. The embedding requirement goes to $X^\alpha \hookrightarrow C^0$, which allows for a smaller α and thus fewer modes need to be retained.

7.3 Lower Operator Bounds

We have constructed upper bound rates in Theorems 5.6 and 6.5, which count the trainable parameters that are sufficient to approximate the solution operator with one neural network. In this section, we derive lower bound rates that hold generically for the class of governing laws. In particular, we fix the class of governing

Class of constitutive law	Embedding needed	Admissible α	Amplification
$f = f(u)$	$X^\alpha \hookrightarrow C^0$	$\alpha > \frac{d_x}{2}$	none, if $d_x \leq 3$
$f = \mathcal{M} f_{\text{loc}}(u)$, ord $\mathcal{M} = 1$	$X^\alpha \hookrightarrow C^0$	$\alpha \geq 1, \alpha > \frac{d_x}{2}$	$\xi_{\alpha+1}$; logarithmic if $\alpha = 1$
$f = f(u, \nabla u)$, genuinely nonlinear in ∇u	$X^\alpha \hookrightarrow C^1$	$\alpha > 1 + \frac{d_x}{2}$	$\chi = 1 + \frac{(\alpha-2)_+}{2\xi}$, sharp

Table 3: Embedding and Amplification: Gradient-free laws avoid the amplification when $d_x \leq 3$. The divergence form removes the gradient input but amplifies the network error by $\xi_{\alpha+1}$, which is only logarithmic at $\alpha = 1$. The third row is sharp (Proposition 7.1), and χ can be improved to $1 + \frac{(\alpha-2)_+}{2\theta}$ (see Remark 4.2).

equations, allow the part of the architecture that does not depend directly on the governing equation to be arbitrary, and count the parameters needed to resolve the dynamical system.

The upper bounds thus far were formulated so that for governing equations that share the same properties, e.g. Lipschitz constants, growth, and other constants, the proof of Theorem 6.5 provides the same discretization (see also [10] for uniform bounds over classes of equations). Therefore, in that class we can reuse the same transformations and spectral weights. Following the theory of continuous nonlinear widths [19], we will assume that if a parameter or constant depends on the governing equation its dependence is continuous. Without this restriction, a direct bound is difficult to obtain.

We provide three cases, a lower bound for gradient-free laws $f = f(u)$, a lower bound for gradient-dependent laws $f = c_0 u - H(u_x)$, and a lower bound on the spectral dependencies. The proofs follow the same style where we show that the architecture has accuracy η with W parameters chosen by a continuous map w. The W parameters depend on the governing equation. For clarity, we summarize the process as follows. First, we construct governing equations f_c indexed by vectors $c \in \mathbb{R}^{W+1}$, with $|c|_\infty = 1$, so it has one coordinate for each independent feature and so that f_c and f_{-c} differ on each feature. The set $\{|c|_\infty = 1\}$ is homeomorphic to the W-dimensional sphere by an odd map, so the Borsuk-Ulam theorem states that every continuous map from it to $\mathbb{R}^W$ maps some pair of opposite points to the same value [8]. This was used to bound the continuous nonlinear widths from below in [19]. Our argument is related to the sphere embedding widths of [71], except that the two opposite points are separated by the PDE solution rather than by a norm. By the theorem, there is a vector c with $w(f_c) = w(f_{-c})$, that is, two opposite governing equations with the same parameters and therefore the same output. Since that output is within η of each true solution, the two solutions are within 2η of each other. Then we can use the PDE to measure how close the solutions of f_c and f_{-c} are as a function of the parameters and accuracy. In particular, this gives a lower bound relationship between η and W. For each case, we need to construct a solution operator that forces this to occur explicitly.

Theorem 7.3 (Lower Bound for Gradient-Free Laws)**.** *Assume that $\Omega = \mathbb{T}^{d_x}$, $L = -\nu\Delta + c_0 I$ with $\nu, c_0 > 0$, $T > 0$, $\alpha \geq 0$ and $s \geq 1$. Set $\psi_0 := T/6$ to be a constant function. For $0 < R_f \leq \psi_0/2$, let $\mathcal{C}_{R_f}$ be the class of governing equations described by the functions*

$$f(u) = c_0 u + \frac{1}{\psi(u)}, \qquad \operatorname{supp}(\psi - \psi_0) \subset [0,1], \quad \|\psi - \psi_0\|_{C^s} \leq R_f,$$

equipped with the distance $\|\psi - \psi'\|_{C^s}$, and let G_T^f be the solution operator of (2) with f. Suppose that $u_0 \equiv -1$, $(\mathcal{A}_w)_{w \in \mathbb{R}^W}$ is any family of maps from a set of data containing u_0 into $C([0,T]; X^\alpha)$, and let $w : \mathcal{C}_{R_f} \to \mathbb{R}^W$ be continuous. If

$$\left\| \mathcal{A}_{w(f)}(u_0) - G_T^f(u_0) \right\|_{C([0,T];X^\alpha)} \leq \eta,$$

holds for every $f \in \mathcal{C}_{R_f}$, then

$$W \;\geq\; c'\, \eta^{-1/(s+1)}$$

for sufficiently small η and where c' depends only on c_0, α, d_x, T, s, R_f and a smooth bump function ϕ.

The idea for the proof is to reduce the problem to a scalar ODE $y' = 1/\psi(y)$, whose solution reaches the value b at time $\int_{-1}^{b} \psi(y)\, dy$. If the dynamics are accurate, then they also determine the integrals of ψ. We

perturb ψ_0 by $W+1$ smooth bumps with width $1/(W+1)$ with signs determined by c. Since we measure it in C^s, the height of the bumps need scale like $\epsilon_W \sim (W+1)^{-s}$ and so two opposite governing functions differ in the integral over each bump by a value scaling like $(W+1)^{-s-1}$. Relating to the accuracy η then gives $W \gtrsim \eta^{-1/(s+1)}$.

Proof of Theorem 7.3. First we will reduce the PDE to scalar ODEs. Since $\|\psi - \psi_0\|_\infty \le R_f \le \psi_0/2$, every ψ takes values in the interval $[\psi_0/2, 3\psi_0/2]$ and so $1/\psi$ is a bounded and Lipschitz function. Therefore the scalar problem $y' = 1/\psi(y)$, $y(0) = -1$ is well-posed and y is increasing with speed bounded below by $2/(3\psi_0) = 4/T$. For the constant function case, the linear operator L multiplies it by c_0. Since we included c_0 in both L and f, the problem reduces to ψ. In fact, the constant function $y(t)$ solves (2), and by uniqueness of the solution operator we have $G_T^f(u_0) = y$. The trajectory passes through every value in $[-1, 3]$ before time T. The trajectory reaches the value b at time

$$t_\psi(b) = \int_{-1}^{b} \psi(y)\, dy.$$

The norm reduces to $\|y(t)\, 1\|_{X^\alpha} = m_\alpha |y(t)|$, where $m_\alpha := \|1\|_{X^\alpha} = c_0^{\alpha/2}(2\pi)^{d_x/2}$ is the X^α norm of the constant (spatial) function 1, which is an eigenfunction of L with eigenvalue c_0. So we have mapped the system into simpler ODE.

We can show that trajectories being close implies that the integrals of ψ are sufficiently close. Take two function $\psi, \tilde{\psi}$ that are within the class, with corresponding trajectories $y, \tilde{y}$, and assume that $\delta := \sup_{[0,T]} |y - \tilde{y}| \le 1$. For a fixed value $b \in [0, 1]$, we can look at the time it takes y to reach b, which is given by $t := t_\psi(b)$. At that same time t, $\tilde{y}$ reaches some value $\tilde{b} := \tilde{y}(t)$, which is close $|\tilde{b} - b| \le \delta$. By construction we have $t_{\tilde{\psi}}(\tilde{b}) = t = t_\psi(b)$. Therefore, the two times of b differ by the time $\tilde{y}$ needs to travel from the value $\tilde{b}$ to the value b:

$$\left| \int_{-1}^{b} (\psi(z) - \tilde{\psi}(z))\, dz \right| = \left| t_{\tilde{\psi}}(\tilde{b}) - t_{\tilde{\psi}}(b) \right| = \left| \int_{b}^{\tilde{b}} \tilde{\psi}(z)\, dz \right| \le \frac{3\psi_0}{2}\, \delta,$$

since they start from the same initial state -1. The difference between two values, by subtracting the bound above, gives us

$$\left| \int_{b_1}^{b_2} (\psi(z) - \tilde{\psi}(z))\, dz \right| \le 3\psi_0 \delta,$$

for $0 \le b_1 < b_2 \le 1$. Thus accuracy in trajectories leads to accuracy in integrals of ψ.

Next, we perturb the trajectories by bump functions. This is a standard construction of disjoint bump functions in a C^s ball [19]. Let $\phi_i(y) := \phi((W+1)y - i + 1)$, for $1 \le i \le W+1$, be bump functions with disjoint supports in $[0, 1]$, where $\phi \ge 0$ is smooth with $\operatorname{supp} \phi \subset [0, 1]$ and $\int \phi = 1$. Each bump has width $1/(W+1)$, so $\int_0^1 \phi_i(y)\, dy = 1/(W+1)$. Since the input to ϕ is multiplied by $W+1$, the k-th derivative is multiplied by $(W+1)^k$, and since the supports are disjoint, $\|\sum_i c_i \phi_i\|_{C^s} \le C_\phi (W+1)^s$ for $|c|_\infty \le 1$, where $C_\phi := 2\|\phi\|_{C^s}$. The largest allowable height is given by

$$\epsilon_W := \frac{R_f}{C_\phi (W+1)^s},$$

and the functions $\psi_c := \psi_0 + \epsilon_W \sum_i c_i \phi_i$ with $|c|_\infty = 1$ define the governing equation $f_c \in \mathcal{C}_{R_f}$. The map $c \to w(f_c)$ is continuous from $\{|c|_\infty = 1\}$ to $\mathbb{R}^W$, thus by the Borsuk-Ulam theorem there is a c such that $w(f_c) = w(f_{-c})$. Both governing equations are approximated by the same model $\mathcal{A}_{w(f_c)}(u_0)$, so their trajectories are within 2η distance (measured in X^α), which implies that $m_\alpha \delta \le 2\eta$.

We compare the errors and the sizes. Note that if $2\eta > m_\alpha$, then $\frac{m_\alpha R_f}{3C_\phi \psi_0 \eta} < \frac{2R_f}{3C_\phi \psi_0} \le 1$, since $R_f \le \psi_0/2$ and

$$C_\phi = 2\|\phi\|_{C^s} \ge 2 \sup \phi \ge 2.$$

In that case, the bound $(W+1)^{s+1} \geq m_\alpha R_f/(3C_\phi\psi_0\eta)$ holds trivially. If that does not occur, then $\delta \leq 1$ and we can pick i and $|c_i| = 1$. Set $[b_1, b_2]$ to be exactly the support of ϕ_i, and thus on this interval $|\psi_c - \psi_{-c}| = 2\epsilon_W\phi_i$. This gives

$$\frac{2\epsilon_W}{W+1} = \left|\int_{b_1}^{b_2} (\psi_c(z) - \psi_{-c}(z))\, dz\right| \leq 3\psi_0\delta \leq \frac{6\psi_0\,\eta}{m_\alpha}.$$

Since $\epsilon_W = R_f/(C_\phi(W+1)^s)$, the bound gives $(W+1)^{s+1} \geq m_\alpha R_f/(3C_\phi\psi_0\eta)$, which completes the proof. □

Remark 7.4 (Lower and Upper Bounds on the Operator)**.** *The class $\mathcal{C}_{R_f}$ has just one component-wise node defined by the function $1/\psi$. For integer s, we can apply Remark B.2 to approximate this node with the bound from Theorem B.1 and with weights that depend continuously on ψ. Since $\psi_0 = T/6$, the governing law scales like T^{-1}, which means that its Lipschitz constant is on the order of T^{-1}. One consequence is that the time window used in Proposition 2.5 may be shorter than T so multiple windows may be needed. The upper bound on $[0, T]$ can be continued using Proposition 5.9, where the same block is applied on J windows. Since we are looking at a class of functions, J depends on the class and T, but it does not depend on η. So we only adjust constants and not the powers of η. For integer s, the upper and lower bounds are thus*

$$c\,\eta^{-1/(s+1)} \leq W \leq C\,\eta^{-\chi/s}\log\eta^{-1},$$

where $\chi = 1$ when $d_x \leq 3$, both bounds holding uniformly for $\mathcal{C}_{R_f}$. Note that there is an addition of 1 in the lower exponent due to the construction of the proof via integrals; however, we removed this gap in Theorem 7.5 for Hamilton-Jacobi problems. The lower bound does not have a χ since the trajectories are constant in space, so no norm mismatch occurs. This bound complements the exponential lower bounds from [42].

We next consider gradient dependence, specifically, a lower bound for Hamilton-Jacobi PDE in the form $u_t + H(u_x) = \nu u_{xx}$.

Theorem 7.5 (Lower Bound for Gradient Nonlinearities)**.** *Assume that $\Omega = \mathbb{T}$, $L = -\nu\partial_{xx} + c_0 I$ with $\nu, c_0 > 0$, $T > 0$, $\alpha \geq 0$, an integer $s \geq 1$ and $0 < R_f \leq 1$. Let $\mathcal{H}_{R_f}$ be the class of governing equations with $f_H(u, u_x) = c_0 u - H(u_x)$ with $H \in C^\infty(\mathbb{R})$, $\operatorname{supp} H \subset [0,1]$ and $\|H\|_{C^s} \leq R_f$, where the set is equipped with the distance $\|H - H'\|_{C^s}$. Let v_0 be a smooth 2π-periodic function with zero mean. Assume that $v_0 \leq 1$, and $v_0 = 1$ on an interval $I = [x_L, x_R]$ of length $\pi < |x_R - x_L| < 2\pi$. Its integral $V_0(x) := \int_0^x v_0(y)\,dy$ is periodic since v_0 has mean zero. The data set is $\mathcal{U} := \{SV_0 : 0 \leq S \leq 1\}$, where the slopes are exactly S on I. Let $(\mathcal{A}_w)_{w\in\mathbb{R}^W}$ be maps from $\mathcal{U}$ into $C([0,T]; X^\alpha)$, and assume that $w : \mathcal{H}_{R_f} \to \mathbb{R}^W$ is continuous.*

If the accuracy bound

$$\sup_{u_0\in\mathcal{U}} \left\|\mathcal{A}_{w(H)}(u_0) - G_T^{f_H}(u_0)\right\|_{C([0,T];X^\alpha)} \leq \eta,$$

holds for every $H \in \mathcal{H}_{R_f}$, then

$$W \;\geq\; c\,\left(\eta\log\eta^{-1}\right)^{-1/s},$$

for sufficiently small η, where $c > 0$ only depends on ν, c_0, α, T, s, R_f and v_0.

Proof of Theorem 7.5. We are working on $\mathbb{T}$ with $[0, 2\pi)$ and periodic endpoints. We can translate x if needed so that the set where $v_0 = 1$ contains an interval $I = [x_L, x_R] \subset (0, 2\pi)$ whose length is $|I| = x_R - x_L > \pi$. Pick $a \in \left(0, \frac{|I|-\pi}{2}\right)$ and set

$$I_a := [x_L + a,\, x_R - a],$$

to be the interval obtained by shrinking the interval by a at each end. The length is given by $|I_a| = |I| - 2a > \pi$, so I_a is longer than $\mathbb{T} \setminus I_a$, and

$$\varsigma := |I_a| - (2\pi - |I_a|) > 0.$$

Next, consider the bump function $\phi \in C^\infty(\mathbb{R})$ which satisfies the following properties: $\operatorname{supp}\phi \subset [0,1]$, $0 \leq \phi \leq \phi_*$ and $\phi = \phi_*$ on $[\frac{1}{4}, \frac{3}{4}]$. The bumps are flat at maxs and mins. We define the values $C_\phi := \|\phi\|_{C^s}$ and $B_0 := 1 + \|v_0\|_\infty$.

As before, we construct two functions with the same parameters to get the bound. For $c \in \mathbb{R}^{W+1}$ with $|c|_\infty = 1$ let

$$H_c(y) := \epsilon_W \sum_{j=1}^{W+1} c_j \phi\big((W+1)y - j + 1\big), \qquad \epsilon_W := \frac{R_f}{C_\phi (W+1)^s}.$$

The terms have disjoint supports in $[0,1]$, so we have that $H_c \in \mathcal{H}_{R_f}$, $\|H_c\|_\infty \le \epsilon_W \phi_*$ and $\|H_c'\|_\infty \le R_f \le 1$. The set $\{|c|_\infty = 1\}$ is homeomorphic to S^W by an odd map, we can apply Borsuk-Ulam theorem to get a coefficient vector c such that $w(H_c) = w(H_{-c})$. The two Hamiltonians are approximated by the same parameters and thus output the same values, which must be within η of each other. For every $u_0 \in \mathcal{U}$, denote the true solutions by u_+ and u_- associated with H_c and H_{-c}, and so the two true solutions are within 2η to each other with respect to the $C([0,T]; X^\alpha)$. Their means $m_\pm(t) := \frac{1}{2\pi}\int_{\mathbb{T}} u_\pm(t,x)\,dx$ are close. Recall that constants are eigenfunctions of L with eigenvalue c_0 and so $\|v\|_{X^\alpha} \ge c_0^{\alpha/2}\sqrt{2\pi}\,\left|\frac{1}{2\pi}\int_{\mathbb{T}} v\right|$. Therefore, the difference is given by:

$$|m_+(t) - m_-(t)| \le \frac{2\eta}{c_0^{\alpha/2}\sqrt{2\pi}}, \tag{27}$$

for $0 \le t \le T$.

Pick the index i with $|c_i| = 1$ and define

$$S := \frac{i - 1/2}{W+1}$$

for $u_0 := SV_0 = S\int_0^x v_0 \in \mathcal{U}$. Then S is the center of the ith bump, and spatial derivative is $\partial_x u_0 = S v_0 = S$ on I. The solutions are smooth [22], so we can differentiate the equation in x to get the slopes $z_\pm := \partial_x u_\pm$ by solving:

$$\partial_t z_\pm + H'_{\pm c}(z_\pm)\partial_x z_\pm = \nu\,\partial_{xx} z_\pm, \qquad z_\pm(0) = S v_0.$$

The maximum principle gives $|z_\pm - S| \le B_0$.

We can find a dynamical system for the mean value. If we integrate the PDE for $u_\pm$, using $\int_{\mathbb{T}} \partial_{xx} u_\pm\,dx = 0$, then

$$m'_\pm(t) = -\frac{1}{2\pi}\int_{\mathbb{T}} H_{\pm c}\left(z_\pm(t,x)\right)\,dx, \tag{28}$$

which is controlled by the average slope. Over a short time interval, the slopes are close to S. At $t = 0$ the slope equals S on all of I, but diffusion will eventually change it. We bound this change on the inner interval I_a. We can determine a $t_1 \in (0,T]$, and set $\omega := a/(2\nu t_1)$. The perturbation $\tilde{z} := z_\pm - S$ solves the linear PDE

$$\tilde{z}_t + b\tilde{z}_x - \nu\tilde{z}_{xx} = 0 \quad \text{on } [0,t_1]\times I,$$

where $b := H'_{\pm c}(z_\pm)$ satisfies $|b| \le 1$, $\tilde{z}(0,\cdot) = 0$ on I, and $|\tilde{z}| \le B_0$ at endpoints $x = x_L$ and $x = x_R$. We can form a bound using weak maximum principle. Consider a comparison with the function

$$Z(t,x) := B_0 e^{(\omega + \nu\omega^2)t}\left(e^{\omega(x - x_R)} + e^{\omega(x_L - x)}\right)$$

The first term is at least B_0 at $x = x_R$ and similarly with the second at left endpoint $x = x_L$, and both decay towards the interior of I. Applying the linear PDE to Z to check the residual yields

$$Z_t + bZ_x - \nu Z_{xx} = \omega B_0 e^{(\omega+\nu\omega^2)t}\left((1+b)e^{\omega(x-x_R)} + (1-b)e^{\omega(x_L - x)}\right) \ge 0,$$

and noting that $Z \ge |\tilde{z}|$ at $t = 0$ and $x = x_L, x_R$. The weak maximum principle applied to $Z - \tilde{z}$ and $Z + \tilde{z}$ implies that $|\tilde{z}| \le Z$ on $[0,t_1]\times I$. For $x \in I_a$, both $x - x_R$ and $x_L - x$ are at most $-a$, so by replacing $\omega t_1 = a/(2\nu)$, we have the bound

$$|z_\pm - S| \le 2B_0\, e^{a/(2\nu)} e^{-a^2/(4\nu t_1)},$$

on $[0, t_1] \times I_a$. By choosing the constants to be

$$C_2 := 8B_0 e^{a/(2\nu)}, \qquad \ell := \log\big(C_2(W+1)\big) > 1, \qquad t_1 := \min\left\{T, \frac{a^2}{4\nu\ell}\right\},$$

we have $e^{-a^2/(4\nu t_1)} \le e^{-\ell} = 1/(C_2(W+1))$, and

$$|z_\pm - S| \le \frac{1}{4(W+1)}, \tag{29}$$

on $[0, t_1] \times I_a$. This shows that the two slopes are close to S, and improves as W increases.

We next relate the slopes to the means. By (29) on $[0, t_1] \times I_a$, the slopes are within a quarter bump, i.e. $(W+1)z_\pm - i + 1 \in [\frac{1}{4}, \frac{3}{4}]$. In this subinterval, the only bump that is nonzero is the ith one (by construction) and it equals ϕ_* (flat), which yields

$$H_{\pm c}(z_\pm) = \pm\epsilon_W c_i \phi_*, \qquad H_c(z_+) - H_{-c}(z_-) = 2\epsilon_W c_i \phi_*.$$

Taking the differences of (28) and splitting the integral into

$$m'_-(t) - m'_+(t) = \frac{1}{2\pi}\int_{I_a} \left(H_c(z_+) - H_{-c}(z_-)\right)\, dx + \frac{1}{2\pi}\int_{\mathbb{T}\setminus I_a} \left(H_c(z_+) - H_{-c}(z_-)\right)\, dx,$$

we will bound each term. The first integral equals $\frac{1}{\pi}\epsilon_W c_i \phi_* |I_a|$, and the second is at most $\frac{1}{\pi}\epsilon_W \phi_* (2\pi - |I_a|)$. We have that $|I_a| > 2\pi - |I_a|$, so the first integral is larger in absolute value than the second integral. Thus by the reverse triangle inequality $|m'_+(t) - m'_-(t)| \ge \frac{\varsigma\phi_*}{\pi}\epsilon_W$ on $[0, t_1]$. Since $m_+(0) = m_-(0)$, integrating gives the bound

$$|m_+(t_1) - m_-(t_1)| = \int_0^{t_1} |m'_+(t) - m'_-(t)|\, dt \ge \frac{\varsigma\phi_*}{\pi}\, t_1\, \epsilon_W.$$

Lastly, we compare the bounds. By (27), this differences between the means are at most $2\eta/(c_0^{\alpha/2}\sqrt{2\pi})$. Since $\ell > 1$, we have $t_1 \ge \min\{T, a^2/(4\nu)\}/\ell$, and replacing ϵ_W yields the inequality:

$$(W+1)^s \log\big(C_2(W+1)\big) \ge \frac{C_1 R_f}{\eta}, \qquad C_1 := \frac{c_0^{\alpha/2}\varsigma\phi_* \min\{T, a^2/(4\nu)\}}{\sqrt{2\pi}\, C_\phi}.$$

Choose $c > 0$ with $(2c)^s \frac{2}{s} < C_1 R_f$. If $W + 1 < 2c\,(\eta \log \eta^{-1})^{-1/s}$, then for sufficiently small η we would have $\log(C_2(W+1)) \le \frac{2}{s}\log\eta^{-1}$, so the LHS is smaller than $(2c)^s \frac{2}{s}\,\eta^{-1}$, which is a contradiction. Therefore, the inequality should be $W + 1 \ge 2c\,(\eta \log \eta^{-1})^{-1/s}$, and when the RHS is larger than 1, the result is $W \ge c\,(\eta \log \eta^{-1})^{-1/s}$. □

For $s \ge 2$, the exponent $1/s$ from Theorem 7.5 agrees (up to logs) with the upper bound uniform over this class (Section 8). For first-order Hamilton-Jacobi equations, upper bounds that avoid the curse of parametric complexity were derived in [42]. Our lower bound is over a class of Hamiltonians and is complementary. This is a fairly simple PDE and therefore indicates that genuine nonlinearities in u_x will require such a rate. We extend the lower bounds from Theorems 7.3 and 7.5 to include the effect of learning the spectrum, which matches the spectral term of Theorem 5.14 up to the constant. We can show this by considering just the linear part, setting f to zero.

Proposition 7.6 (Spectral Projection is Sharp)**.** *Let $\tilde{L}$ satisfy the prior assumptions and have eigenpairs $(\tilde{\lambda}_k, e_k)$. The norms are those induced by $\tilde{L}$. Set $T > 0$, $\alpha < \gamma_s$, $\rho_s > 0$, the governing law $f \equiv 0$, and a finite set $\mathcal{K}$ of indices with*

$$\frac{1}{T} \le \tilde{\lambda}_k < \left(\frac{\rho_s}{9\eta}\right)^{1/\zeta_s},$$

for $k \in \mathcal{K}$. For vector $m \in [1,2]^{\mathcal{K}}$, suppose L_m has eigenvalues $m_k\tilde{\lambda}_k$ for $k \in \mathcal{K}$ and $\tilde{\lambda}_k$ otherwise, with the same corresponding eigenfunctions. Let $(\mathcal{A}_w)_{w\in\mathbb{R}^W}$ be maps from $B := \{u_0 : \|u_0\|_{X^{\gamma_s}} \le \rho_s\}$ to $C([0,T]; X^\alpha)$, and let $w : [1,2]^{\mathcal{K}} \to \mathbb{R}^W$ be continuous weights. If, for every $m \in [1,2]^{\mathcal{K}}$, the following holds

$$\sup_{u_0\in B} \left\| \mathcal{A}_{w(m)}(u_0) - e^{-tL_m}u_0 \right\|_{X_T^\alpha} \le \eta,$$

then $W \ge |\mathcal{K}|$. Assuming the two-sided Weyl bound $c_W k^{2/d_x} \le \tilde{\lambda}_k \le C_W k^{2/d_x}$, we have $W \ge c\eta^{-d_x/(\gamma_s-\alpha)}$ for sufficiently small η. More generally, for an operator of order 2ℓ, with $k^{2\ell/d_x}$ in the Weyl bound, the same argument yields $W \ge c\,\eta^{-d_x/(\ell(\gamma_s-\alpha))}$.

Proof of Proposition 7.6. We follow the same proof style as before. Let $K := |\mathcal{K}|$ and $W \le K-1$. The boundary of $[-1,1]^K$ is homeomorphic to S^{K-1} by a homeomorphism that commutes with $c \mapsto -c$. Applying the Borsuk-Ulam theorem to $c \mapsto w(m(c))$, with $m(c) := \frac{1}{2}(3+c) \in [1,2]^K$, gives c on that boundary with $w(m(c)) = w(m(-c))$. On the boundary, some coordinates satisfy $|c_k| = 1$, so there is a mode whose multiplier is 1 for one of the two vectors and 2 for the other, i.e., $\{m_k(c), m_k(-c)\} = \{1,2\}$.

Next we examine the separation on the mode. We can choose initial data to have all its mass on that mode, e.g. $u_0 := \rho_s(\tilde{\lambda}_k)^{-\gamma_s/2}e_k \in B$. Since $f \equiv 0$, we can explicitly solve the PDE and get two solutions: $e^{-\tilde{\lambda}_k t}u_0$ and $e^{-2\tilde{\lambda}_k t}u_0$. At the time $t = (\tilde{\lambda}_k)^{-1} \le T$, their difference is given by

$$\left(e^{-1} - e^{-2}\right) u_0 = \left(e^{-1} - e^{-2}\right) \rho_s(\tilde{\lambda}_k)^{-\gamma_s/2}e_k.$$

By assumption, e_k is an eigenfunction of $\tilde{L}$ with eigenvalue $\tilde{\lambda}_k$, and by construction we have $\|e_k\|_{X^\alpha} = (\tilde{\lambda}_k)^{\alpha/2}$. Therefore, their difference in the X^α norm induced by $\tilde{L}$ is given by:

$$\left(e^{-1} - e^{-2}\right) \rho_s(\tilde{\lambda}_k)^{(\alpha-\gamma_s)/2} = \left(e^{-1} - e^{-2}\right) \rho_s(\tilde{\lambda}_k)^{-\zeta_s},$$

recalling that $\zeta_s = \frac{\gamma_s-\alpha}{2}$. The condition that $\tilde{\lambda}_k < (\rho_s/(9\eta))^{1/\zeta_s}$ on $\mathcal{K}$ gives $\rho_s(\tilde{\lambda}_k)^{-\zeta_s} > 9\eta$. Using $(e^{-1} - e^{-2})9 > 2$, the difference has X^α norm greater than 2η. But both solutions are within η of the same trajectory $\mathcal{A}_{w(m(c))}(u_0)$, which is a contradiction by triangle inequality. Therefore, $W \ge K$.

The Weyl bounds give $\#\{k : \tilde{\lambda}_k < \Lambda\} \ge (\Lambda/C_W)^{d_x/2} - 1$ and $\#\{k : \tilde{\lambda}_k < 1/T\} \le (c_W T)^{-d_x/2}$. With $\Lambda := (\rho_s/(9\eta))^{1/\zeta_s}$, the number of modes with $T^{-1} \le \tilde{\lambda}_k < \Lambda$ is at least $c(\rho_s/\eta)^{d_x/(2\zeta_s)}$ for sufficiently small η, and $d_x/(2\zeta_s) = d_x/(\gamma_s - \alpha)$. This completes the proof. □

The bound is sharp, since for $f \equiv 0$, exact spectral weights $e^{-m_k\tilde{\lambda}_k\Delta t}$ produce e^{-tL_m} on the first N modes, so the only error is from truncating and encoding. Since e^{-tL_m} commutes with P_N and is a contraction on X^α,

$$\left\|(I - P_N)e^{-tL_m}u_0\right\|_{X^\alpha} \le \rho_s(\tilde{\lambda}_{N+1})^{-\zeta_s} \le \frac{\eta}{2}$$

when $\tilde{\lambda}_{N+1} \ge (2\rho_s/\eta)^{1/\zeta_s}$. By Weyl's law, this holds for some $N \le C(\rho_s/\eta)^{d_x/(\gamma_s-\alpha)}$, and choosing h so that the encoding error is at most $\eta/2$ gives accuracy η. So the spectral term, specifically the exponent in Theorem 5.14, is sharp when the spectrum is unknown. Learning a self-adjoint linear operator with a known eigenspace was investigated in [16]. Proposition 7.6 is a parametric version, where the number of unknowns is quantified by Weyl's law.

8 Explicit Examples

In this section, we apply the main theorems to several popular equations. The results are summarized in Table 4. The key is that the structural exponent does not depend on the dimension, and even for complicated dynamics considered here, the nonlinear governing equations can be associated with fairly simple DAGs. Upper bounds for the terminal time solutions for Allen-Cahn, Cahn-Hilliard, and the two-dimensional Navier-Stokes were shown through stable spectral schemes emulation in [10]. In [35], they showed bounds

for Navier-Stokes by emulating a pseudo-spectral method. We show that our bounds hold along the whole trajectory with shared weights and that we have lower bounds that match or nearly match.

We consider the torus $\Omega = \mathbb{T}^{d_x}$ with $H = L^2(\mathbb{T}^{d_x}; \mathbb{R}^d)$ and $L = -\nu\Delta + c_0 I$, where $\nu, c_0 > 0$. In this setting, L is strictly positive, $X^s \simeq H^s(\mathbb{T}^{d_x}; \mathbb{R}^d)$, the eigenfunctions are trigonometric, and Weyl's gives $\lambda_N \asymp N^{2/d_x}$. Given the equation $u_t - \nu\Delta u = g(u, \nabla u)$, we write it in the form (2) by shifting $f = g + c_0 u$. This adds one affine node to the DAG, but does not change the structural exponent. On the torus the trapezoidal rule satisfies the quadrature assumptions (see Remark 4.19), the FFT gives the fast transforms (see Assumption 4.5), and the Moser estimates give Assumption 2.8 for smooth governing laws. This reduces the overhead from several assumptions. Unless otherwise stated, we set

$$\beta = \alpha - 1, \qquad \alpha < \gamma < \alpha + 1, \qquad \theta = \frac{1}{2}, \qquad \zeta = \frac{\gamma - \alpha}{2} \in \left(0, \frac{1}{2}\right).$$

8.1 Allen-Cahn with Explicit Constants

To show that the settings of the main theorems are obtainable, we show one example explicitly. Let $\Omega = \mathbb{T}^1$, $L = -\partial_{xx} + I$ with $\lambda_k = 1 + k^2$, and the Allen-Cahn governing law $f(u) = 2u - u^3$. Since it is polynomial, $p_f = 0$ is attained.

The law is gradient-free, so we just need u and thus the embedding simplifies to $X^\alpha \hookrightarrow C^0$, that is, $\alpha > \frac{1}{2}$. We take $\alpha = \frac{7}{4} < 2$, so there is no amplification. Since f maps H^α into H^1, we may set $\beta = 1$, which satisfies $\beta > \frac{1}{2}$ (needed for Assumption 4.16(ii)) and gives $\theta = \frac{5}{8}$. Setting $\gamma = \frac{11}{4}$, gives us $\zeta = \frac{1}{2}$. The Lipschitz estimate of f into $X^1 = H^1$ involves u', so there we use $X^\alpha \hookrightarrow C^1$, which holds since $\alpha > \frac{3}{2}$. With $R_0 = 1$ and $R = 3$, one can show that $\|u\|_{C^1} \le m_K = C_{\text{emb}} R \approx 4$ on the ball, and f has Lipschitz constant $L_R \le 207$ from X^α to X^1. The contraction condition of Proposition 2.5 then yields $T_0 \approx 10^{-4}$. The horizon is short because L_R grows like m_K^2 and T_0 scales like $L_R^{-1/\theta} = L_R^{-8/5}$. Longer times can be reached by the continuation arguments and energy estimate.

For the nonlinear part, since $\alpha < 2$, we have $\varepsilon \asymp \eta$. Approximating the cubic term through $u^3 = \frac{1}{4}\big((u + u^2)^2 - (u - u^2)^2\big)$ with Yarotsky's construction of the square [86] gives a block $\widehat{f}_\varepsilon$ with 206, 244, and 290 weights at $\varepsilon = 10^{-2}, 10^{-3}, 10^{-4}$. By Lemma 3.4, this one block is shared across every grid point, time level and sweep, and the spectral layers add $3N$ shared weights, with $N \asymp \eta^{-1}$ by Theorem 5.6. The trainable count $W_{\text{train}} = W(\widehat{f}_\varepsilon) + 3N$ is therefore

η	N	$W(\widehat{f}_\varepsilon)$	$3N$	W_{train}
10^{-2}	10^2	206	3×10^2	5×10^2
10^{-3}	10^3	254	3×10^3	3×10^3
10^{-4}	10^4	300	3×10^4	3×10^4

independently of the number of grid points, time levels and sweeps. The block grows by less than 100 weights over two orders of magnitude in accuracy, so the trainable count is dominated by the spectral weights, which reflect the regularity of the data rather than the law.

The values above use $R_0 = 1$, which is valid for one window. However, to extend to longer times we need the solution to remain in the ball. The solutions tend to the states ± 1 where $\|\pm 1\|_{X^\alpha} = \sqrt{2\pi}$. Since $\|u\|_{L^\infty} \le 0.54\|u\|_{X^{7/4}}$, if we start with $\|u_0\|_{X^\alpha} \le 1$ then by the maximum principle it will satisfy $|u| \le 1$ for all time, and using mild form on the unit interval gives the bound $\|u(t)\|_{X^{7/4}} \le 9.1$ for all t and a uniform bound in X^γ for $t \ge 1$. Using this value, we have that $R_0 \approx 9.1$ implies $m_K \approx 25$, $L_R \approx 8 \times 10^3$ and $T_0 \approx 3 \times 10^{-7}$. Therefore, to move through a unit interval, the method needs about 3×10^6 windows by Proposition 5.9. If we just used contraction, we would need to rescale per-window error by $2^{-3\times 10^6}$. However, using the energy bounds from Lemma 5.10 gives $\Theta_{\text{win}} = e^{3T/2}$ and reduces the accuracy factor to $A_J \approx 4 \times 10^{12}$ for $T_{\text{final}} = 1$, provided each window contains at least two time steps so that $\Delta t\, \Gamma_0^2 \le 1$. This means that we need less than 650 more parameters for the nonlinear block, due to the log scaling.

8.2 Reaction-Diffusion Equations and Viscous Conservation Laws

First we look at reaction-diffusion equations. Consider $u_t - \nu\Delta u = g(u)$ with $g : \mathbb{R}^d \to \mathbb{R}^d$ applied component-wise and $g(0) = 0$. Since the law is gradient-free, the network needs only $X^\alpha \hookrightarrow C^0$, that is, $\alpha > \frac{d_x}{2}$. We take $\frac{d_x}{2} < \beta < \alpha$ and $g \in C^s$ with $s \geq \lceil\beta\rceil + 1$. We can apply Moser estimates [2, 66] to show that $u \mapsto g(u)$ is Lipschitz on bounded sets from X^α to X^β, and thus the contraction theory Proposition 2.5 can be applied. The DAG is simply a single component-wise C^s node, so $p_{\max} = 1/s$, and Theorem 5.6 gives

$$W(\widehat{f}_\varepsilon) \leq C\eta^{-\chi/s}\log\eta^{-1}.$$

For $d_x \leq 3$ the exponents can be chosen so that $\frac{d_x}{2} < \beta < \alpha < 2$, for example, $\alpha = \frac{7}{4}$ and $\beta = \frac{13}{8}$ in three dimensions, and so the factor is $\chi = 1$. The same exponent s yields well-posedness and approximation bounds, so $p_{\max} \leq 1/(\lceil\beta\rceil + 1)$ in this setting.

Example 8.1 (Allen-Cahn). *For $g(u) = u - u^3$ the graph consists of two multilinear nodes, $u \odot u$ and $(u \odot u) \odot u$, therefore we have $p_f = 0$, and using Corollary 5.7, $W(\widehat{f}_\varepsilon) \leq C\log\eta^{-1}$. The same holds for any polynomial reaction with diffusion in every variable, including the diffusive FitzHugh-Nagumo system.*

Example 8.2 (Arrhenius Equation). *For a non-polynomial reaction term, consider $g(u) = A\,e^{-E/(u+a)}$ with $u + a$ bounded away from zero on K. The single node is C^∞ near K, so $p_f = 0$ but it cannot be attained. Thus we have $W(\widehat{f}_\varepsilon) \leq C_p\eta^{-p}\log\eta^{-1}$ for every $p > 0$.*

Viscous Conservation Laws. Consider $u_t + \partial_x F(u) = \nu u_{xx}$ on $\mathbb{T}$ with $F \in C^s$, $s \geq 3$ and $F(0) = F'(0) = 0$. The governing law can be written as $f(u, u_x) = -F'(u)\,u_x + c_0 u$. In one dimension, we can pick $\alpha \in (\frac{3}{2}, 2)$, so $X^\alpha \hookrightarrow C^1$ and $\chi = 1$. With $\beta = \alpha - 1 \in (\frac{1}{2}, 1)$ and $s \geq 3$, f is Lipschitz from X^α to X^β, where $s \geq 3$ is needed to control the difference $F'(u) - F'(v)$ in H^β. The graph (Figure 2(b)) has a component-wise node $F' \in C^{s-1}$ and a bilinear node, so $p_{\max} = \frac{1}{s-1}$ and

$$W(\widehat{f}_\varepsilon) \leq C\,\eta^{-1/(s-1)}\log\eta^{-1}.$$

In divergence form, we have $f = -\partial_x F(u)$, and the derivative can be moved to a spectral layer (Proposition 7.2). With $\alpha = 1$ the amplification is only logarithmic, and the exponent improves to $\frac{1}{s}$.

Example 8.3 (Viscous Burgers). *For $F(u) = \frac{1}{2}u^2$, the node F' is the identity and the graph reduces to one bilinear node (Figure 2(a)). Therefore, $p_f = 0$ and $W(\widehat{f}_\varepsilon) \leq C\log\eta^{-1}$. The same holds for $f = -(u \cdot \nabla)u$ in any dimension.*

Viscous Hamilton-Jacobi equations. Consider $u_t + H(u_x) = \nu u_{xx}$ on $\mathbb{T}$ with $H \in C^s$, $s \geq 2$, and $H(0) = 0$, so that $f(u, u_x) = c_0 u - H(u_x)$. Similar to the conservation laws, we take $\alpha \in (\frac{3}{2}, 2)$ and $\beta = \alpha - 1$, so $X^\alpha \hookrightarrow C^1$ and $\chi = 1$. If we write $H(u_x) - H(v_x) = (u_x - v_x)\int_0^1 H'(v_x + \tau(u_x - v_x))\,d\tau$ and use the properties of $H^\beta(\mathbb{T})$, the condition $s \geq 2$ leads to a Lipschitz bound for f from X^α to X^β. The governing law written as a DAG with a single component-wise C^s node applied to u_x gives $p_{\max} = \frac{1}{s}$ and

$$W(\widehat{f}_\varepsilon) \leq C\,\eta^{-1/s}\log\eta^{-1},$$

where C depends on H only through the norm $\|H\|_{C^s}$. For s, the Taylor construction in Remark B.2 attains this bound with continuous weights. Thus, the bound is uniform over the class $\mathcal{H}_{R_f}$ from Theorem 7.5, where $\|H\|_{C^s} \leq R_f$ makes the constants in the discretization estimates uniform. In this setting, we have

$$c\,\left(\eta\log\eta^{-1}\right)^{-1/s} \;\leq\; W \;\leq\; C\,\eta^{-1/s}\log\eta^{-1}.$$

This result is over the class $H \in \mathcal{H}_{R_f}$, which is smooth with $p_f = 0$ for each element. No uniform bound over this class can improve the $\frac{1}{s}$ exponent.

8.3 Two-Dimensional Navier-Stokes

Consider $u_t - \nu\Delta u + \mathbb{P}[(u\cdot\nabla)u] = 0$, $\nabla\cdot u = 0$, on $\mathbb{T}^2$, where $\mathbb{P}$ is the Leray projection. The space is set to $H = L^2_{0,\sigma}(\mathbb{T}^2;\mathbb{R}^2)$, the mean-zero divergence-free fields, and $L = -\nu\Delta$ is the Stokes operator. The linear operator is strictly positive on H and commutes with $\mathbb{P}$. The governing equation is $f = \mathbb{P}f_{\text{loc}}$ with $f_{\text{loc}}(u,\nabla u) = -(u\cdot\nabla)u$, so Lemma 5.17 can be applied with $\widetilde{H} = L^2_0(\mathbb{T}^2;\mathbb{R}^2)$. Since $\mathbb{P}$ is the orthogonal projection onto H, $\mu_k \equiv 1$ (from Lemma 5.17) and the Leray projection does not add additional costs. This also shows that the network can approximate the system without seeing the pressure.

First, we consider the velocity form. The embedding into C^1 requires $\alpha > 2$, so we take $\alpha = 2+\iota$ and $\beta = 1+\iota$ for small $\iota > 0$. The properties of $H^{1+\iota}(\mathbb{T}^2)$ allow f_{loc} to be Lipschitz from X^α to $\widetilde{X}^\beta$. The DAG is simply one bilinear node (Figure 2(a)), so $p_f = 0$. The neural network error is amplified with factor $\xi_\alpha \asymp \lambda_N^{\iota/2}$, so the block needs accuracy $\varepsilon \asymp \eta^\chi$ with $\chi = \frac{\gamma-2}{\gamma-2-\iota}$. However, because $p_f = 0$, this just changes the constant, i.e.,

$$W(\widehat{f}_\varepsilon) \le C\log\varepsilon^{-1} \le C\chi\log\eta^{-1} + C,$$

so the block has $C\log\eta^{-1}$ parameters.

If we use the divergence form, we can reduce the embedding requirement. The nonlinearity can be written as $f = -\mathbb{P}\nabla\cdot(u\otimes u)$, where the local law $f_{\text{loc}}(u) = u\otimes u$ only sees u, and the divergence part becomes a diagonal spectral layer with $|\mu_k| \asymp |k|$ (see Proposition 7.2). The embedding only requires $\alpha > 1$, so we can set $\alpha = 1+\iota$ and $\beta = \iota$. This removes the gradient transforms, but the network error goes through the derivative layer and is amplified by $\xi_{\alpha+1} \asymp \lambda_N^{\iota/2}$, as in velocity form. Since $p_f = 0$, the block count again is $C\log\eta^{-1}$.

For the long horizon times, recall that the two-dimensional equations are globally well-posed for smooth data and have X^γ bounds on every finite horizon. Thus the continuation theory, Proposition 5.9, extends the result to any T_{final} with the same block. As discussed in Section 5.3, the constant depends on T_{final}, and exponentially on the number of windows if only the contraction is used.

Cahn-Hilliard. We consider a fourth order equation $u_t + \epsilon^2\Delta^2 u = \Delta(u^3 - u)$ on $\mathbb{T}^{d_x}$ with $d_x \le 3$. Since the mean of u is conserved, we can work with mean-zero fields, i.e., a nonzero mean m_0 is removed by writing $u = m_0 + w$, and this does not change the law being polynomial. Set $H = L^2_0(\mathbb{T}^{d_x})$ and $L = \epsilon^2\Delta^2$, which is strictly positive on H with eigenvalues $\lambda_k = \epsilon^2|k|^4$, so that $X^s \simeq H^{2s}_0$. The governing equation is $f = \mathcal{M}f_{\text{loc}}$ with $\mathcal{M} = \Delta$ and $f_{\text{loc}}(u) = u^3 - u$. Since $\mathcal{M}^* e_k = -|k|^2 e_k$, the spectral multipliers satisfy $|\mu_k| = \epsilon^{-1}\lambda_k^{1/2}$, so we can apply Proposition 7.2 with $\beta = \alpha - 1$. Since we use $s = \alpha$ in the quadrature, we need $X^\alpha \hookrightarrow C(\mathbb{T}^{d_x})$, i.e. $\alpha > \frac{d_x}{4}$, which holds for $\alpha = 1$ when $d_x \le 3$. Then the local part f_{loc} is Lipschitz from $X^1 = H^2$ to $\widetilde{X}^1 = H^2$. The DAG for f_{loc} has two multilinear nodes so $p_f = 0$, and the spectral layers need $4N$ weights (for D_{an}, D_S, D_W and D_μ). The network error goes through the layer defined by D_μ so it gets amplified by the factor $\xi_{\alpha+1} = \xi_2$, but it is only logarithmic for $\alpha = 1$. Since $p_f = 0$, the only change is to constants and we get

$$W(\widehat{f}_\varepsilon) \le C\log\eta^{-1}.$$

Since L is a fourth order operator, by Remark 5.15 with $m = 2$ we have $N \asymp M \asymp \eta^{-d_x/(2(\gamma_s-\alpha))}$. The example shows that the theory applies to higher order L.

8.4 Summary

Table 4 summarizes the two complexity measures used in this work, i.e., the trainable parameters (Theorem 5.6) and the size of the full network (Theorem 5.14). Since we can loop the blocks and apply the network to space-time points using parameter sharing, the complexity is mostly on the block and spectral terms. Let

$$v_0 := \frac{d_x}{r} + \frac{1}{\zeta_t}$$

denote the exponent of the number MQ of space-time points, thus the full network has $W(\mathcal{A}_n) \le C\eta^{-v}(\log\eta^{-1})^2$ with $v = v_0 + p_{\max}\chi_s$. The best choice for r is $r = \gamma_s - \alpha$, which leads to $M \asymp N$.

Equation	Local law f	Nodes	d_x	$p_{\max}$	Block $W(\widehat{f_\varepsilon})$	Spectral	Full network v
Allen-Cahn	$u - u^3$	2 mult.	≥ 1	0	$C \log \eta^{-1}$	$3N$	v_0
Reaction-diffusion, $g \in C^s$	$g(u)$	1 comp.	≤ 3	$\frac{1}{s}$	$C\eta^{-1/s} \log \eta^{-1}$	$3dN$	$v_0 + \frac{1}{s}$
Viscous Burgers	$-u\,\partial_x u$	1 mult.	1	0	$C \log \eta^{-1}$	$4N$	v_0
Conservation law, $F \in C^s$	$-F'(u)\,\partial_x u$	1 c. + 1 m.	1	$\frac{1}{s-1}$	$C\eta^{-\frac{1}{s-1}} \log \eta^{-1}$	$4N$	$v_0 + \frac{1}{s-1}$
Viscous Hamilton–Jacobi, $H \in C^s$	$-H(\partial_x u)$	1 comp.	1	$\frac{1}{s}$	$C\eta^{-1/s} \log \eta^{-1}$	$4N$	$v_0 + \frac{1}{s}$
Cahn–Hilliard	$\Delta(u^3 - u)$	2 mult.	≤ 3	0	$C \log \eta^{-1}$	$4N$	v_0
2D Navier-Stokes	$-\mathbb{P}[(u \cdot \nabla)u]$	1 mult.	2	0	$C \log \eta^{-1}$	$10N$	v_0

Table 4: Trainable and Total Complexity for accuracy η in $C([0,T]; X^\alpha)$, with $v_0 = \frac{d_x}{r} + \frac{1}{\zeta_t}$ and $N \asymp \eta^{-d_x/(\gamma_s - \alpha)}$, except for the (4th order) Cahn-Hilliard where $N \asymp \eta^{-d_x/(2(\gamma_s - \alpha))}$.

9 Conclusion

In empirical tests, neural networks of moderate size obtain small errors on the standard nonlinear PDE benchmarks. We showed that for semilinear parabolic equations $u_t + Lu = f(u, \nabla u)$ with $-L$ generating an analytic semigroup, the structure of the equations gives an explanation for this. The trainable cost of approximating the solution operator splits into two terms. The first is the governing law, which appears through a single point-wise block shared across grid points, time points, and Picard sweeps. This block needs $C\eta^{-p\chi_s} \log \eta^{-1}$ parameters for every p above the structural exponent p_f. The exponent is determined by the computation graph of f, applies to laws that are nonlinear in ∇u, and does not depend on the spatial dimension or the number of state components. In particular, for the two-dimensional Navier-Stokes equations, the nonlinear block costs only $C \log \eta^{-1}$ parameters, where the Leray projection is absorbed into the spectral layers. The second term is data-based and appears in the learned spectral layers with a bounded number of weights per mode, i.e., $N \asymp \eta^{-d_x/(\gamma_s - \alpha)}$ total. Since the weights are tied, neither term grows with the grid, the time step or the number of sweeps, which is consistent with the ability of these networks to transfer across resolutions. Since all time levels are updated at once via the Picard iteration, the depth is polylogarithmic in η^{-1}.

The analysis only uses the max-norm error of the block on the states observed by the iteration, and its Lipschitz constant is not needed. The error is controlled in the stronger norm of $C([0,T]; X^\alpha)$ along the whole trajectory, since only the exact discrete map needs to contract and parabolic smoothing improves point-wise errors to X^α. All of the other sources of error are derived from the mild formulation and bounded separately. The cost is the factor $\chi_s \geq 1$ in the exponent, which exceeds 1 only when $\alpha > 2$. It is sharp as an estimate, but it reflects the formulation rather than the method, since it disappears for gradient-free laws when $d_x \leq 3$.

Both terms have lower bounds for architectures whose law-dependent parameters are continuous. Over a class of viscous Hamilton-Jacobi laws, at least $c(\eta \log \eta^{-1})^{-1/s}$ parameters are needed, which for $s \geq 2$ matches the upper bound up to logarithmic factors. Over a gradient-free class, at least $c\,\eta^{-1/(s+1)}$ are needed. When the spectrum is not known, one weight per mode is needed, and thus the spectral error term is sharp. For Allen-Cahn the constants are explicitly computed. The block has a few hundred parameters on a single time window, so the trainable count is controlled by the spectral weights. Energy estimates can be used to extend the result to long horizons with growth set by the one-sided rate of the law rather than by the contraction.

The results focus on the approximation rate and not generalization, which is left to future work. In particular, the algebraic decay of the errors in the size of the training set is consistent with the algebraic parameter counts shown here, but proving it requires additional statistical analysis [75, 81, 10]. The error is measured in X^α, which forces $\alpha > 2$ for gradient laws when $d_x \geq 2$. Bounds in weaker norms follow directly and measuring the error in such norms from start to finish should remove most of the amplification. The proofs use a first-order Lawson-Euler step, which limits the temporal accuracy. The analysis looks at one operator at a time so extending it to families of equations, as in multiple-operator learning [83, 81, 82], would need additional structural conditions on the family of graphs, as was done for classes of stable spectral schemes in [10].

Acknowledgments

This work was supported in part by the National Science Foundation under grant NSF 2606034.

References

[1] Ben Adcock, Gregor Maier, and Rahul Parhi. Towards sharp minimax risk bounds for operator learning. *arXiv preprint arXiv:2512.17805*, 2025.

[2] Hajer Bahouri, Jean-Yves Chemin, and Raphaël Danchin. *Fourier analysis and nonlinear partial differential equations*, volume 343. Springer, 2011.

[3] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. Deep equilibrium models. *Advances in neural information processing systems*, 32, 2019.

[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] Guy E Blelloch. Prefix sums and their applications. 1990.

[6] Jonas Blessing, Philipp Schmocker, and Alessandro Sgarabottolo. Neural operators approximate strongly continuous convex monotone semigroups. *arXiv preprint arXiv:2609.02727*, 2026.

[7] Helmut Bolcskei, Philipp Grohs, Gitta Kutyniok, and Philipp Petersen. Optimal approximation with sparsely connected deep neural networks. *SIAM Journal on Mathematics of Data Science*, 1(1):8–45, 2019.

[8] Karol Borsuk. Drei sätze über die n-dimensionale euklidische sphäre. *Fundamenta Mathematicae*, 20(1):177–190, 1933.

[9] 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.

[10] Nisha Chandramoorthy, Daniel Sanz-Alonso, and Nathan Waniorek. From spectral methods to sample complexity bounds for fourier neural operators. *arXiv preprint arXiv:2607.00320*, 2026.

[11] Ke Chen, Meenakshi Krishnan, and Haizhao Yang. Error analysis for learning the time-stepping operator of evolutionary pdes. *arXiv preprint arXiv:2509.04256*, 2025.

[12] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

[13] 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.

[14] Steven M Cox and Paul C Matthews. Exponential time differencing for stiff systems. *Journal of Computational Physics*, 176(2):430–455, 2002.

[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. *arXiv preprint arXiv:2203.13181*, 2022.

[16] Maarten V de Hoop, Nikola B Kovachki, Nicholas H Nelsen, and Andrew M Stuart. Convergence rates for learning linear operators from noisy data. *SIAM/ASA Journal on Uncertainty Quantification*, 11(2):480–513, 2023.

[17] Tim De Ryck and Siddhartha Mishra. Generic bounds on the approximation error for physics-informed (and) operator learning. *Advances in Neural Information Processing Systems*, 35:10945–10958, 2022.

[18] Ronald DeVore, Boris Hanin, and Guergana Petrova. Neural network approximation. *Acta Numerica*, 30:327–444, 2021.

[19] Ronald A DeVore, Ralph Howard, and Charles Micchelli. Optimal nonlinear approximation. *Manuscripta mathematica*, 63(4):469–478, 1989.

[20] James R Driscoll and Dennis M Healy. Computing fourier transforms and convolutions on the 2-sphere. *Advances in applied mathematics*, 15(2):202–250, 1994.

[21] Weinan E, Martin Hutzenthaler, Arnulf Jentzen, Thomas Kruse, et al. Multilevel picard iterations for solving smooth semilinear parabolic heat equations. *Partial Differential Equations and Applications*, 2(6):80, 2021.

[22] Lawrence C Evans. *Partial differential equations*, volume 19. American mathematical society, 2022.

[23] Michael Feischl, Christoph Schwab, and F Zehetgruber. Neural general operator networks via banach fixed point iterations. *SAM Research Report*, 2025, 2025.

[24] Takashi Furuya, Ryo Ozawa, and Jenn-Nan Wang. Approximation theory of laplacian-based neural operators for reaction-diffusion system. *arXiv preprint arXiv:2605.12025*, 2026.

[25] Takashi Furuya, Michael Puthawala, Maarten V de Hoop, and Matti Lassas. Can neural operators always be continuously discretized? *Advances in Neural Information Processing Systems*, 37:98936–98993, 2024.

[26] Takashi Furuya, Koichi Taniguchi, and Satoshi Okuda. Quantitative approximation for neural operators in nonlinear parabolic equations. In *International Conference on Learning Representations*, volume 2025, pages 85528–85556, 2025.

[27] Karol Gregor and Yann LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th international conference on international conference on machine learning*, pages 399–406, 2010.

[28] Zhongkai Hao, Chang Su, Songming Liu, Julius Berner, Chengyang Ying, Hang Su, Anima Anandkumar, Jian Song, and Jun Zhu. Dpot: Auto-regressive denoising operator transformer for large-scale pde pre-training. *arXiv preprint arXiv:2403.03542*, 2024.

[29] Daniel Henry. *Geometric theory of semilinear parabolic equations*. Springer, 2006.

[30] Maximilian Herde, Bogdan Raonić, Tobias Rohner, Roger Käppeli, Roberto Molinaro, Emmanuel De Bezenac, and Siddhartha Mishra. Poseidon: Efficient foundation models for pdes. *Advances in Neural Information Processing Systems*, 37:72525–72624, 2024.

[31] 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.

[32] Marlis Hochbruck and Alexander Ostermann. Exponential integrators. *Acta Numerica*, 19:209–286, 2010.

[33] Martin Hutzenthaler, Arnulf Jentzen, Thomas Kruse, and Tuan Anh Nguyen. A proof that rectified deep neural networks overcome the curse of dimensionality in the numerical approximation of semilinear heat equations. *SN partial differential equations and applications*, 1(2):10, 2020.

[34] 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.

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

[36] Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to pdes. *Journal of Machine Learning Research*, 24(89):1–97, 2023.

[37] Nikola B Kovachki, Samuel Lanthaler, and Hrushikesh Mhaskar. Data complexity estimates for operator learning. *arXiv preprint arXiv:2405.15992*, 2024.

[38] Samuel Lanthaler. Operator learning with pca-net: upper and lower complexity bounds. *Journal of Machine Learning Research*, 24(318):1–67, 2023.

[39] Samuel Lanthaler. Operator learning of lipschitz operators: An information-theoretic perspective. *arXiv preprint arXiv:2406.18794*, 2024.

[40] 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), 2022.

[41] Samuel Lanthaler, Roberto Molinaro, Patrik Hadorn, and Siddhartha Mishra. Nonlinear reconstruction for operator learning of pdes with discontinuities. *arXiv preprint arXiv:2210.01074*, 2022.

[42] Samuel Lanthaler and Andrew M Stuart. The parametric complexity of operator learning. *IMA Journal of Numerical Analysis*, 46(2):647–712, 2026.

[43] Samuel Lanthaler, Andrew M Stuart, and Margaret Trautner. Discretization error of fourier neural operators. *arXiv preprint arXiv:2405.02221*, 2024.

[44] J Douglas Lawson. Generalized runge-kutta processes for stable systems with large lipschitz constants. *SIAM Journal on Numerical Analysis*, 4(3):372–380, 1967.

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

[46] J-L Lions. Quelques méthodes de résolution des problèmes aux limites non-linéaires. *Dunod*, 1969.

[47] Hao Liu, Zecheng Zhang, Wenjing Liao, and Hayden Schaeffer. Neural scaling laws of deep relu and deep operator network: A theoretical study. *arXiv preprint arXiv:2410.00357*, 2024.

[48] 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. *arXiv preprint arXiv:2409.09811*, 2024.

[49] 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.

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

[51] Gabriel J Lord, Catherine E Powell, and Tony Shardlow. An introduction to computational stochastic pdes. *Cambridge Texts Appl. Math*, 50, 2014.

[52] 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.

[53] Alessandra Lunardi. *Interpolation theory*, volume 16. Edizioni della normale, Pisa, 2018.

[54] 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.

[55] Tanya Marwah, Zachary Lipton, and Andrej Risteski. Parametric complexity bounds for approximating pdes with neural networks. *Advances in Neural Information Processing Systems*, 34:15044–15055, 2021.

[56] Tanya Marwah, Zachary Chase Lipton, Jianfeng Lu, and Andrej Risteski. Neural network approximations of pdes beyond linearity: A representational perspective. In *international conference on machine learning*, pages 24139–24172. PMLR, 2023.

[57] Tanya Marwah, Ashwini Pokle, J Zico Kolter, Zachary Lipton, Jianfeng Lu, and Andrej Risteski. Deep equilibrium based neural operators for steady-state pdes. *Advances in Neural Information Processing Systems*, 36:15716–15737, 2023.

[58] Michael McCabe, Bruno Régaldo-Saint Blancard, Liam Parker, Ruben Ohana, Miles Cranmer, Alberto Bietti, Michael Eickenberg, Siavash Golkar, Geraud Krawezik, Francois Lanusse, et al. Multiple physics pretraining for spatiotemporal surrogate models. *Advances in Neural Information Processing Systems*, 37:119301–119335, 2024.

[59] Vishal Monga, Yuelong Li, and Yonina C Eldar. Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Processing Magazine*, 38(2):18–44, 2021.

[60] Ariel Neufeld and Tuan Anh Nguyen. Rectified deep neural networks overcome the curse of dimensionality in the numerical approximation of gradient-dependent semilinear heat equations. *arXiv preprint arXiv:2403.09200*, 2024.

[61] Joost AA Opschoor, Christoph Schwab, and Jakob Zech. Exponential relu dnn expression of holomorphic maps in high dimension. *Constructive Approximation*, 55(1):537–582, 2022.

[62] Amnon Pazy. *Semigroups of linear operators and applications to partial differential equations*. Springer Science & Business Media, 2012.

[63] Philipp Petersen and Felix Voigtlaender. Optimal approximation of piecewise smooth functions using deep relu neural networks. *Neural Networks*, 108:296–330, 2018.

[64] Tomaso Poggio and Maia Fraser. Compositional sparsity of learnable functions. *Bulletin of the American Mathematical Society*, 61(3), 2024.

[65] Niklas Reinhardt, Sven Wang, and Jakob Zech. Statistical learning theory for neural operators. *Journal of Machine Learning Research*, 27(102):1–82, 2026.

[66] Thomas Runst and Winfried Sickel. *Sobolev spaces of fractional order, Nemytskij operators, and nonlinear partial differential equations*, volume 3. Walter de Gruyter, 2011.

[67] Hayden Schaeffer. Learning partial differential equations via data discovery and sparse optimization. *Proceedings. Mathematical, physical, and engineering sciences*, 473(2197):20160446, 2017.

[68] Johannes Schmidt-Hieber. Nonparametric regression using deep neural networks with ReLU activation function. *Annals of Statistics*, 48:1875–1897, 2020.

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

[70] Christoph Schwab and Jakob Zech. Deep learning in high dimension: Neural network expression rates for generalized polynomial chaos expansions in uq. *Analysis and Applications*, 17(01):19–55, 2019.

[71] Jonathan W Siegel. Sharp lower bounds on the manifold widths of sobolev and besov spaces. *Journal of Complexity*, 85:101884, 2024.

[72] Shashank Subramanian, Peter Harrington, Kurt Keutzer, Wahid Bhimji, Dmitriy Morozov, Michael W Mahoney, and Amir Gholami. Towards foundation models for scientific machine learning: Characterizing scaling and transfer behavior. *Advances in neural information processing systems*, 36:71242–71262, 2023.

[73] 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.

[74] Yifan Sun, Linan Zhang, and Hayden Schaeffer. Neupde: Neural network based ordinary and partial differential equations for modeling time-dependent data. In *Mathematical and Scientific Machine Learning*, pages 352–372. PMLR, 2020.

[75] Koichi Taniguchi and Sho Sonoda. Generalization error bounds for picard-type operator learning in nonlinear parabolic pdes. *arXiv preprint arXiv:2605.10277*, 2026.

[76] Roger Temam. *Infinite-dimensional dynamical systems in mechanics and physics*. Springer Science & Business Media, 2012.

[77] Roger Temam. *Navier–Stokes equations: theory and numerical analysis*, volume 343. American Mathematical Society, 2024.

[78] Vidar Thomée. *Galerkin finite element methods for parabolic problems*. Springer Science & Business Media, 2007.

[79] Alex Townsend, Marcus Webb, and Sheehan Olver. Fast polynomial transforms based on toeplitz and hankel matrices. *Mathematics of Computation*, 87(312):1913–1934, 2018.

[80] Adrien Weihs, Chunyang Liao, Jingmin Sun, and Hayden Schaeffer. Kernel methods for learning operators with multiple inputs and outputs. *arXiv preprint arXiv:2608.11831*, 2026.

[81] Adrien Weihs and Hayden Schaeffer. Generalization bounds and statistical guarantees for multi-task and multiple operator learning with mno networks. *arXiv preprint arXiv:2604.01961*, 2026.

[82] Adrien Weihs and Hayden Schaeffer. Multiple neural operators achieve near-optimal rates for multi-task learning. *arXiv preprint arXiv:2605.22724*, 2026.

[83] Adrien Weihs, Jingmin Sun, Zecheng Zhang, and Hayden Schaeffer. A deep learning framework for multi-operator learning: Architectures and approximation theory. *Applied and Computational Harmonic Analysis*, page 101925, 2026.

[84] Hermann Weyl. Das asymptotische verteilungsgesetz der eigenwerte linearer partieller differentialgleichungen (mit einer anwendung auf die theorie der hohlraumstrahlung). *Mathematische Annalen*, 71(4):441–479, 1912.

[85] Jia-Qi Yang and Lei Shi. Efficient approximation for encoder–decoder neural operators via variation spaces. *arXiv preprint arXiv:2606.01244*, 2026.

[86] Dmitry Yarotsky. Error bounds for approximations with deep relu networks. *Neural networks*, 94:103–114, 2017.

[87] Dmitry Yarotsky. Optimal approximation of continuous functions by very deep relu networks. In *Conference on learning theory*, pages 639–649. PMLR, 2018.

[88] Dmitry Yarotsky. Elementary superexpressive activations. In *International conference on machine learning*, pages 11932–11940. PMLR, 2021.

[89] 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.

[90] 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.

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

A Proofs

The proofs are deferred here for readability. The sections are grouped by similar techniques or results.

A.1 Functional Analysis and Spaces

Proof of Proposition 2.5. By the assumptions in the proposition and Assumption 2.3, we have the semigroup bound $\|S(\cdot)u_0\|_{X_T^\alpha} \le \|u_0\|_{X^\alpha}$, and the Duhamel bound

$$\|\mathcal{D}[g]\|_{X_T^\alpha} \le CT^\theta \|g\|_{X_T^\beta},$$

where $\theta := 1 - \frac{\alpha-\beta}{2} > 0$ and C depends on $\alpha - \beta$ but is independent of T.

For a fixed initial function $u_0 \in X^\alpha$, we set the radius of the ball to $R := 1 + 2\|u_0\|_{X^\alpha}$. Then the semigroup assumption yields

$$\|S(\cdot)u_0\|_{X_T^\alpha} \le \|u_0\|_{X^\alpha} < \frac{R}{2}.$$

By the local Lipschitz assumption on f with $v = 0 \in B_R^\alpha$ and using that $f(0,0) = 0$, we can bound the force term by:

$$\|f(u, \nabla u)\|_{X_T^\beta} = \|f(u, \nabla u) - f(0,0)\|_{X_T^\beta} \le L_R \|u\|_{X_T^\alpha},$$

for every $u \in B_R^\alpha$. Therefore, $\|f(u, \nabla u)\|_{X_T^\beta} \le L_R R$.

To prove that the ball is invariant, i.e., $\Phi(B_R^\alpha) \subseteq B_R^\alpha$, take any $u \in B_R^\alpha$, then

$$\begin{aligned}
\|\Phi[u]\|_{X_T^\alpha} &\le \|S(\cdot)u_0\|_{X_T^\alpha} + \|\mathcal{D}[f(u, \nabla u)]\|_{X_T^\alpha} \\
&\le \|u_0\|_{X^\alpha} + CT^\theta \|f(u, \nabla u)\|_{X_T^\beta} \\
&\le \|u_0\|_{X^\alpha} + CT^\theta L_R R \\
&\le \frac{R}{2} + \frac{R}{2} = R,
\end{aligned}$$

holds when $T_0 > 0$ is chosen so that $CT^\theta L_R \le \frac{1}{2}$ for every $T \in (0, T_0]$.

Next, to show that the map is contractive, take any $u, v \in B_R^\alpha$. Then by the assumptions on the Duhamel operator and by its linearity

$$\begin{aligned}
\|\Phi[u] - \Phi[v]\|_{X_T^\alpha} = \|\mathcal{D}[f(u, \nabla u) - f(v, \nabla v)]\|_{X_T^\alpha} &\le CT^\theta \|f(u, \nabla u) - f(v, \nabla v)\|_{X_T^\beta} \\
&\le CT^\theta L_R \|u - v\|_{X_T^\alpha}.
\end{aligned}$$

Let $\kappa_0 = CT^\theta L_R$. Since $T \le T_0$, by construction we have $\kappa_0 \le \frac{1}{2} < 1$. Thus the map is a (strict) contraction on B_R^α,

$$\|\Phi[u] - \Phi[v]\|_{X_T^\alpha} \le \kappa_0 \|u - v\|_{X_T^\alpha}.$$

Since B_R^α is a closed subset of the Banach space X_T^α, it is complete and thus by the Banach fixed point theorem there exists a unique fixed function $u \in B_R^\alpha$ of Φ which satisfies $u = \Phi[u]$. By Definition 2.4, u is the unique mild solution on $[0, T]$ in B_R^α. □

Proof of Proposition 2.6. In the mild formulation, using Assumption 2.3 and the assumptions in the proposition, we have

$$\begin{aligned}
\|u(t)\|_{X^\gamma} &\le \|S(t)u_0\|_{X^\gamma} + \left\| \int_0^t S(t-s) f(u(s), \nabla u(s))\, ds \right\|_{X^\gamma} \\
&\le Ct^{-(\gamma-\alpha)/2} \|u_0\|_{X^\alpha} + Ct^{1-(\gamma-\beta)/2} \|f(u, \nabla u)\|_{X_T^\beta},
\end{aligned}$$

for $t > 0$. Since $u \in X_T^\alpha$, by assumption we have $f(u, \nabla u) \in X_T^\beta$. Since $\gamma - \beta < 2$, the right-hand side is finite for every $t > 0$ which implies that $u(t) \in X^\gamma$ for every $t \in (0, T]$. To show that $u \in C((0, T]; X^\gamma)$,

note that for any interval $[\delta, T]$ the first term of the mild solution will have the term $S(\cdot - \delta)S(\delta)u_0$ with $S(\delta)u_0 \in X^\gamma$. This is continuous by using the continuity of the semigroup on X^γ. The second term is Duhamel component which lies in $C([0,T]; X^\gamma)$ by the continuity argument following Assumption 2.3 with $s = \beta$ and $r = \gamma$. □

Proof of Lemma 2.7. The result of (i) is direct, it follows from $u(t) \in B_R^\alpha$ for every t and the definition of F_R.

For (ii), if $t > 0$ we can apply Proposition 2.6 to get

$$\|u(t)\|_{X^\gamma} \le Ct^{-\zeta}\|u_0\|_{X^\alpha} + Ct^{1-\bar\sigma}\|g\|_{X_T^\beta} \le Ct^{-\zeta}R + Ct^{\theta-\zeta}F_R,$$

using $1 - \bar\sigma = \theta - \zeta > 0$, which is the same as $\gamma - \beta < 2$. Then, we multiply by t^ζ and take the supremum which gives (ii) for $t \le T$.

Lastly, for (iii) we can set $\tau = t - s$ and write the mild form as:

$$u(t) - u(s) = [S(\tau) - I]\, u(s) + \int_s^t S(t-r)g(r)\, dr.$$

Using the spectral representation and applying the inequality $|e^{-x} - 1| \le x^\zeta$ for $x \ge 0$ and $\zeta \in [0,1]$, the first term is bounded by:

$$\|[S(\tau) - I]\, u(s)\|_{X^\alpha}^2 = \sum_k \lambda_k^\alpha \left|e^{-\lambda_k\tau} - 1\right|^2 |u_k(s)|^2 \le \tau^{2\zeta} \sum_k \lambda_k^{\alpha+2\zeta}|u_k(s)|^2 = \tau^{2\zeta}\|u\|_{X^\gamma}^2,$$

since $\alpha + 2\zeta = \gamma$. Using the bound (ii), the first term is controlled by $\tau^\zeta R_\gamma s^{-\zeta} = (t-s)^\zeta s^{-\zeta} R_\gamma$. For the second term, applying Assumption 2.3 and (i) yields:

$$\left\|\int_s^t S(t-r)g(r)\, dr\right\|_{X^\alpha} \le CF_R \int_s^t (t-r)^{-\sigma}\, dr = \frac{CF_R}{\theta}\tau^\theta \le CF_R\tau^\zeta T^{\theta-\zeta},$$

where $\theta > \zeta$. Since $s \le T$, we have that $T^{\theta-\zeta} \le s^{-\zeta}T^\theta$, and adding the two bounds (with $C \ge 1$) yields the results. The bound on $[g]_\zeta$ follows from the (point-wise) Lipschitz inequality (3). □

Proof of Lemma 2.9. We iterate over θ sized steps by setting $s_0 := \alpha$, $s_{j+1} := \min\{s_j + \theta, \gamma'\}$ and $J := \lceil(\gamma' - \alpha)/\theta\rceil$, so $s_J = \gamma'$. Let $B_j := \sup_{0\le t\le T}\|u(t)\|_{X^{s_j}}$ with $B_0 \le R$ and assume $B_j < \infty$.

By Assumption 2.8 with $s = s_j$, the force $g = f(u, \nabla u)$ satisfies the bound $\sup_t \|g(t)\|_{X^{s_j-2\sigma}} \le K_R(1 + B_j)$, where $s_j - 2\sigma \ge \beta \ge 0$. The mild form gives $u(t) = S(t)u_0 + \int_0^t S(t-\tau)g(\tau)\, d\tau$, the integral converges in X^β because g is continuous with values in X^β. The difference between the target exponent and the force's exponent is $s_{j+1} - (s_j - 2\sigma) \le \theta + 2\sigma = 1 + \sigma < 2$, so applying the smoothing from Assumption 2.3 yields $\|S(t-\tau)g(\tau)\|_{X^{s_{j+1}}} \le C(t-\tau)^{-(s_{j+1}-s_j+2\sigma)/2}\|g(\tau)\|_{X^{s_j-2\sigma}}$. This has an integrable singularity, so after applying it to each truncation P_K of the integrand, integrating over time, and sending $K \to \infty$ (which holds by continuity in $\|\cdot\|_{X^s}$) we get

$$\sup_{0\le t\le T}\left\|\int_0^t S(t-\tau)g(\tau)\, d\tau\right\|_{X^{s_{j+1}}} \le CT^{1-(s_{j+1}-s_j+2\sigma)/2}K_R(1 + B_j).$$

The first term in the mild form is controlled by

$$\|S(t)u_0\|_{X^{s_{j+1}}} \le \|u_0\|_{X^{s_{j+1}}} \le \lambda_1^{-(\gamma'-s_{j+1})/2}\|u_0\|_{X^{\gamma'}}.$$

Thus $B_{j+1} \le C(1 + \|u_0\|_{X^{\gamma'}} + B_j)$, and after J steps $B_J \le C'(1 + \|u_0\|_{X^{\gamma'}})$ (where C' depends on J). The bound on the force is from Assumption 2.8 with $s = \gamma'$. □

Proof of Lemma 2.10. Let $g(\tau) := f(u(\tau), \nabla u(\tau))$, $\kappa = \min\{1, \zeta'\}$ with $\zeta' = (\gamma' - \alpha)/2$, and fix $0 \le s < t \le T$ with $\delta := t - s$. Using the mild form at time s, we have the difference

$$u(t) - u(s) = (S(\delta) - I)\, u(s) + \int_s^t S(t-\tau) g(\tau)\, d\tau.$$

For $\lambda > 0$ and $\kappa \in [0, 1]$ we have the simple inequality $|e^{-\lambda\delta} - 1| \le \min\{1, \lambda\delta\} \le (\lambda\delta)^\kappa$, so using the spectral representation, we bound:

$$\|(S(\delta) - I)u(s)\|_{X^\alpha}^2 = \sum_k \lambda_k^\alpha \left| e^{-\lambda_k \delta} - 1 \right|^2 |\langle u(s), e_k \rangle|^2 \le \delta^{2\kappa} \|u(s)\|_{X^{\alpha+2\kappa}}^2.$$

Since $\kappa \le \zeta'$, we have $\alpha + 2\kappa \le \gamma'$. The Hilbert space embedding and Lemma 2.9 combine to give

$$\|(S(\delta) - I)u(s)\|_{X^\alpha} \le \delta^\kappa \lambda_1^{-(\gamma' - \alpha - 2\kappa)/2} \|u(s)\|_{X^{\gamma'}} \le \lambda_1^{-(\gamma' - \alpha - 2\kappa)/2} R_{\gamma'} (t-s)^\kappa.$$

To bound the other term, we apply Lemma 2.9 to get $\sup_\tau \|g(\tau)\|_{X^{\gamma' - 2\sigma}} \le K_R(1 + R_{\gamma'})$. Since $\alpha - (\gamma' - 2\sigma) = 2(\sigma - \zeta')$, the smoothing from Assumption 2.3 gives

$$\|S(t-\tau) g(\tau)\|_{X^\alpha} \le C(t-\tau)^{-(\sigma - \zeta')} K_R(1 + R_{\gamma'}),$$

for $\sigma > \zeta'$.

When $\sigma \le \zeta'$, the index α is not larger than the index $\gamma' - 2\sigma$ so we can control the term using contractivity of S and the first eigenvalue. Specifically, it has the same form but with power 0 and constant $\lambda_1^{-(\zeta' - \sigma)}$. In both cases the exponent is $a' := (\sigma - \zeta')_+ \le \sigma < 1$, and thus integration in $\tau \in (s, t)$ yields

$$\left\| \int_s^t S(t-\tau) g(\tau)\, d\tau \right\|_{X^\alpha} \le \frac{C}{1 - a'} (t-s)^{1-a'} K_R(1 + R_{\gamma'}).$$

The exponent is $1 - a' = \min\{1, \theta + \zeta'\} \ge \kappa$, since $\theta > 0$. Also, the time difference is no more than T, i.e. $t - s \le T$, and thus $(t-s)^{1-a'} \le T^{1-a'-\kappa}(t-s)^\kappa$, which allows us to bound the second term by at most $CT^{\min\{1, \theta+\zeta'\} - \kappa} K_R(1 + R_{\gamma'})(t-s)^\kappa$.

Adding the two bounds gives $\|u(t) - u(s)\|_{X^\alpha} \le C_t (t-s)^\kappa$ with

$$C_t := \lambda_1^{-(\gamma' - \alpha - 2\kappa)/2} R_{\gamma'} + CT^{\min\{1, \theta+\zeta'\} - \kappa} K_R(1 + R_{\gamma'}).$$

For the second inequality in the lemma, $u(t)$ and $u(s)$ are both in ball of radius R in X^α, so by the Lipschitz bound (3) we have

$$\|g(t) - g(s)\|_{X^\beta} \le L_R \|u(t) - u(s)\|_{X^\alpha} \le L_R C_t (t-s)^\kappa.$$

□

Proof of Proposition 2.12. The proof is direct. Let $u_k := \langle u, e_k \rangle_H$, then

$$\|u - P_N u\|_{X^\beta}^2 = \sum_{k=N+1}^\infty \lambda_k^{-(\alpha-\beta)} \lambda_k^\alpha |u_k|^2 \le \lambda_{N+1}^{-(\alpha-\beta)} \sum_{k=N+1}^\infty \lambda_k^\alpha |u_k|^2 \le \lambda_{N+1}^{-(\alpha-\beta)} \|u\|_{X^\alpha}^2,$$

which proves the bound. □

A.2 Neural Networks and DAG approximation

Proof of Lemma 3.3. Both part (1) and the equal depth case in part (3) are implied directly from equation (5), since the first layers can be stacked on the common input and all other layers can be written block-wise. If the depths differ, the affine output layer $y = Az + b$ for the shallower network is replaced by the ReLU layer $(\sigma(Az + b), \sigma(-Az - b))$, which at most doubles the weights for that layer, and the $2m_i$ nonnegative

terms are moved through $\Delta - 1$ more layers by diagonal identity layer ($2m_i$ per layer). At the output layer, the terms are mapped by using $y = \sigma(y) - \sigma(-y)$ with $2m_i$ weights. This adds up to no more than $W(\mathcal{N}_i) + 2m_i\Delta$ additional weights.

For part (2), since the product of sparse matrices is not necessarily sparse, we combine the output layer of $\mathcal{N}_1$ with the input layer of $\mathcal{N}_2$ with the identity layer used in part (4). That is, the output $y = Az + b$ of $\mathcal{N}_1$ is replaced by a split ReLU layer $(\sigma(Az + b), \sigma(-Az - b))$, and the first layer $A'y + b'$ of $\mathcal{N}_2$ uses

$$A' \left(\sigma(Az + b) - \sigma(-Az - b)\right) + b'.$$

This at most doubles each of the modified steps' number of nonzero entries, where the depth is at most $\operatorname{dep}(\mathcal{N}_1) + \operatorname{dep}(\mathcal{N}_2)$, and $W \leq 2W(\mathcal{N}_1) + 2W(\mathcal{N}_2)$.

For part (4), we have the identity $y = \sigma(y) - \sigma(-y)$ for all real y, and together with the recombination has four nonzero weights per coordinate. For part (5), the hidden layer has weights $(1, 1)$ and biases $(\mathrm{a}, -\mathrm{a})$. The output layer has weights $(1, -1)$ and bias $-\mathrm{a}$, which totals seven nonzero terms. Note that if $|y| \leq \mathrm{a}$ then $\sigma(y + \mathrm{a}) - \sigma(y - \mathrm{a}) - \mathrm{a} = (y + \mathrm{a}) - 0 - \mathrm{a} = y$, and if $y > \mathrm{a}$ we have $(y + \mathrm{a}) - (y - \mathrm{a}) - \mathrm{a} = \mathrm{a}$. By symmetry it holds for $y < -\mathrm{a}$. □

Proof of Lemma 3.4. Both versions map X to the array whose i-th entry is obtained through the same composition of affine maps and activations to X_i. Since σ acts element-wise and (4) applies (A_l, b_l) at each point, the two formulation agrees at each point. The depth does not change since the layers are a one-to-one correspondence and the number of distinct parameters are the same since it is the same action. When written as dense layers, the convolution stores one copy of each layer per grid point. □

Proof of Theorem 3.7. We construct the approximating network by going through each node and creating a local approximation. Then we control the total error by propagating approximation error inductively through the topological ordering of the graph.

First, we derive the relevant constants. Since the graph $G = (\mathcal{V}, \mathcal{E})$ is finite and acyclic, there exists a topological ordering of its vertices. We relabel the vertices so that

$$(i, j) \in \mathcal{E} \quad \Longrightarrow \quad i < j.$$

Let $\mathcal{V}_{\text{in}}$ denote the input nodes, and let $\mathcal{V}_{\text{nl}}$ denote the set of non-affine nodes. For every node $j \in \mathcal{V}$, let Ψ_j be the map computed at node j, and let $z_j(x)$ for $x \in K$ be the exact intermediate variable produced during the evaluation of the DAG.

To bound the error propagation through the DAG, we need to define the Lipschitz constant and the corresponding input/output spaces for each node. Since the intermediate variables are continuous, the input space K is compact, and the graph contains finitely many nodes, every intermediate variable's range

$$K_j := \{z_j(x) : x \in K\}$$

is compact. For every node j which is not an input node, let $\operatorname{pa}(j)$ denote the set of parent nodes of j, and define the set of admissible inputs as

$$K_j^{\text{in}} := \prod_{i \in \operatorname{pa}(j)} K_i,$$

which is compact by construction. The product is equipped with the maximum norm. By Assumption 3.5, every nodal map is either affine, multilinear, or component-wise C^{s_j} on an open neighborhood of K_j^{in}. Thus, each map Ψ_j is continuously differentiable on some open neighborhood $U_j \supset K_j^{\text{in}}$. Since K_j^{in} is compact, we can choose a compact set $\overline{U}_j$ so that

$$K_j^{\text{in}} \subset \operatorname{int}(\overline{U}_j) \subset \overline{U}_j \subset U_j \subset \operatorname{dom}(\Psi_j).$$

The set $\overline{U}_j$ is needed as a buffer to ensure that perturbation can extend outside of the exact range while remaining valid. The map Ψ_j is C^1, so it is locally Lipschitz on the open set U_j and a locally Lipschitz map is Lipschitz on every compact subset. Therefore, Ψ_j is Lipschitz on $\overline{U}_j$ with Lipschitz constant $L_j := \operatorname{Lip}(\Psi_j, \overline{U}_j) < \infty$ with respect to the maximum norm. For the affine nodes, L_j can be set to operator norm.

Since the graph is finite, the maximum Lipschitz constant of a node map

$$L_{\max} := \max_{j \in \mathcal{V} \setminus \mathcal{V}_{\text{in}}} L_j$$

is finite. Once every buffer set is fixed, let $\Lambda_1 := \max\{1, L_{\max}\}$ and let ρ be the minimal distance between the admissible inputs and the boundary of the buffer sets, i.e.,

$$\rho := \min_{j \in \mathcal{V} \setminus \mathcal{V}_{\text{in}}} \operatorname{dist}\left(K_j^{\text{in}}, \partial \overline{U}_j\right).$$

Since $K_j^{\text{in}} \subset \operatorname{int}(\overline{U}_j)$ and both sets are compact, the distance for each j is positive, and since there are only finitely many nodes $\rho > 0$.

This construction shows that if every parent variable at a node is perturbed by at most ρ in the maximum norm, then the perturbation does not leave $\overline{U}_j$, where the Lipschitz estimate for Ψ_j remains valid. This is needed for propagation and evaluation.

Next we construct a deep ReLU approximation of each node. Let $\delta \in (0,1)$ be the error parameter for the nodes. For each $j \in \mathcal{V}$, the ReLU realization $\widetilde{\Psi}_{j,\delta}$ is determined by the type of node as follows. If j is affine, then Ψ_j is exactly realizable by linear layers, and thus we set $\widetilde{\Psi}_{j,\delta} := \Psi_j$, and incur no error

$$\sup_{y \in \overline{U}_j} \|\Psi_j(y) - \widetilde{\Psi}_{j,\delta}(y)\| = 0.$$

The depth and number of weights are independent of δ and so we set $p_j = q_j = 0$. If there is at least one nonlinear node, then we can neglect any fixed parameter costs associated with the finitely many linear nodes.

If node j is multilinear, then since K_j^{in} is compact, there exists $b_j > 0$ with $K_j^{\text{in}} \subset \left([-b_j, b_j]^n\right)^{r_j}$, where r_j is the number of input variables associated with node j. The buffer set is the cube,

$$\overline{U}_j := \left([-2b_j, 2b_j]^n\right)^{r_j}, \qquad U_j := \left((-3b_j, 3b_j)^n\right)^{r_j},$$

and thus $\operatorname{dist}\left(K_j^{\text{in}}, \partial \overline{U}_j\right) \geq b_j > 0$. Using Theorem B.3, Ψ_j can be approximated on this cube by a ReLU network $\widetilde{\Psi}_{j,\delta}$ with

$$\sup_{y \in \overline{U}_j} \|\Psi_j(y) - \widetilde{\Psi}_{j,\delta}(y)\| \leq \delta,$$

and with parameter counts

$$\operatorname{dep}(\widetilde{\Psi}_{j,\delta}) \leq c_j \log(\delta^{-1}), \qquad W(\widetilde{\Psi}_{j,\delta}) \leq c_j \log(\delta^{-1}),$$

where $c_j > 0$ depends only on Ψ_j, $\overline{U}_j$, and the node dimensions. Thus for each multilinear node we set $p_j = 0$ and $q_j = 1$.

Suppose that node j is a component-wise nonlinear node induced by a scalar function $g_j \in C^{s_j}$. Let d_j denote the dimension of the input space of node j, so that $\overline{U}_j \subset \mathbb{R}^{d_j}$ and $\Psi_j(y_1, \ldots, y_{d_j}) = \left(g_j(y_1), \ldots, g_j(y_{d_j})\right)$. Since $\overline{U}_j \subset \mathbb{R}^{d_j}$ is compact, its coordinate-wise projections are compact subsets of $\mathbb{R}$ and so there exists a compact interval $[a_j, b_j] \subset \mathbb{R}$ containing every coordinate of every point in $\overline{U}_j$. We denote $[a_j, b_j] =: [b_j^-, b_j^+]$ to avoid confusion. We can apply Theorem B.1 to g_j on $[b_j^-, b_j^+]$, to get that for every $\delta \in (0,1)$, there is a ReLU network $N_{j,\delta}$ such that

$$\|g_j - N_{j,\delta}\|_{L^\infty([b_j^-, b_j^+])} \leq \frac{\delta}{\sqrt{d_j}},$$

with parameter counts

$$\operatorname{dep}(N_{j,\delta}) \leq C_j \log(\delta^{-1}), \qquad W(N_{j,\delta}) \leq C_j\, \delta^{-1/s_j} \log(\delta^{-1}),$$

where $C_j > 0$ depends only on g_j, s_j, and $[b_j^-, b_j^+]$. The full node approximation is given by

$$\widetilde{\Psi}_{j,\delta}(y_1, \ldots, y_{d_j}) := \big(N_{j,\delta}(y_1), \ldots, N_{j,\delta}(y_{d_j})\big).$$

For every $y \in \overline{U}_j$, the difference is

$$\|\Psi_j(y) - \widetilde{\Psi}_{j,\delta}(y)\|^2 = \sum_{i=1}^{d_j} |g_j(y_i) - N_{j,\delta}(y_i)|^2 \le d_j \left(\frac{\delta}{\sqrt{d_j}}\right)^2 = \delta^2,$$

yielding

$$\sup_{y \in \overline{U}_j} \|\Psi_j(y) - \widetilde{\Psi}_{j,\delta}(y)\| \le \delta.$$

Since $\widetilde{\Psi}_{j,\delta}$ contains d_j parallel copies of $N_{j,\delta}$, the depth is the same, while the number of weights increases by at most a factor of d_j. Therefore, there exists a constant $c_j > 0$, depending only on g_j, $\overline{U}_j$, and the node dimension d_j, such that

$$\operatorname{dep}(\widetilde{\Psi}_{j,\delta}) \le c_j \log(\delta^{-1}), \qquad W(\widetilde{\Psi}_{j,\delta}) \le c_j\, \delta^{-1/s_j} \log(\delta^{-1}),$$

where we can set $p_j = {s_j}^{-1}$ and $q_j = 1$.

Next we estimate the propagation of error through the DAG. Let $\mathcal{V} = \{j_1, \ldots, j_M\}$ be the the topological ordering of the graph so that every parent of j_m is in $\{j_1, \ldots, j_{m-1}\}$. For each $m = 1, \ldots, M$, we define the node-wise error by

$$e_m := \sup_{x \in K} \left\| z_{j_m}(x) - \widetilde{z}_{j_m}(x) \right\|,$$

and the cumulative error (with $E_0 = 0$) by

$$E_m := \max_{1 \le l \le m} e_l.$$

If j_m is an input node, then $e_m = 0$. If j_m is not an input node, we can write

$$\mathbf{z}_{\mathrm{pa}(j_m)}(x) := (z_i(x))_{i \in \mathrm{pa}(j_m)}, \qquad \widetilde{\mathbf{z}}_{\mathrm{pa}(j_m)}(x) := (\widetilde{z}_i(x))_{i \in \mathrm{pa}(j_m)},$$

and so

$$z_{j_m}(x) = \Psi_{j_m}\left(\mathbf{z}_{\mathrm{pa}(j_m)}(x)\right), \qquad \widetilde{z}_{j_m}(x) = \widetilde{\Psi}_{j_m,\delta}\left(\widetilde{\mathbf{z}}_{\mathrm{pa}(j_m)}(x)\right).$$

We use induction to derive the error bound. Assuming that $E_{m-1} \le \rho$, we have

$$\left\| \mathbf{z}_{\mathrm{pa}(j_m)}(x) - \widetilde{\mathbf{z}}_{\mathrm{pa}(j_m)}(x) \right\|_\infty \le E_{m-1} \le \rho$$

for $x \in K$. Since

$$\mathbf{z}_{\mathrm{pa}(j_m)}(x) \in K_{j_m}^{\mathrm{in}} \subset \operatorname{int}(\overline{U}_{j_m})$$

and

$$\operatorname{dist}\left(K_{j_m}^{\mathrm{in}}, \partial \overline{U}_{j_m}\right) \ge \rho,$$

it follows that $\widetilde{\mathbf{z}}_{\mathrm{pa}(j_m)}(x) \in \overline{U}_{j_m}$ for all $x \in K$. In addition, the Lipschitz bound for Ψ_{j_m} and the (local) approximation error for $\widetilde{\Psi}_{j_m,\delta}$ hold. The node-wise error is bounded by

$$\begin{aligned}
e_m &= \sup_{x \in K} \left\| \Psi_{j_m}\left(\mathbf{z}_{\mathrm{pa}(j_m)}(x)\right) - \widetilde{\Psi}_{j_m,\delta}\left(\widetilde{\mathbf{z}}_{\mathrm{pa}(j_m)}(x)\right) \right\| \\
&\le \sup_{x \in K} \left\| \Psi_{j_m}\left(\mathbf{z}_{\mathrm{pa}(j_m)}(x)\right) - \Psi_{j_m}\left(\widetilde{\mathbf{z}}_{\mathrm{pa}(j_m)}(x)\right) \right\| + \sup_{x \in K} \left\| \Psi_{j_m}\left(\widetilde{\mathbf{z}}_{\mathrm{pa}(j_m)}(x)\right) - \widetilde{\Psi}_{j_m,\delta}\left(\widetilde{\mathbf{z}}_{\mathrm{pa}(j_m)}(x)\right) \right\| \\
&\le L_{j_m} E_{m-1} + \sup_{x \in K} \left\| \Psi_{j_m}\left(\widetilde{\mathbf{z}}_{\mathrm{pa}(j_m)}(x)\right) - \widetilde{\Psi}_{j_m,\delta}\left(\widetilde{\mathbf{z}}_{\mathrm{pa}(j_m)}(x)\right) \right\|,
\end{aligned}$$

using the Lipschitz constant for the exact node function Ψ_{j_m}. If j_m is an affine node, then the second term is zero. Otherwise, by the approximation bounds we have

$$\sup_{x\in K}\left\|\Psi_{j_m}\left(\widetilde{\mathbf{z}}_{\mathrm{pa}(j_m)}(x)\right)-\widetilde{\Psi}_{j_m,\delta}\left(\widetilde{\mathbf{z}}_{\mathrm{pa}(j_m)}(x)\right)\right\|\le\sup_{y\in\overline{U}_{j_m}}\left\|\Psi_{j_m}(y)-\widetilde{\Psi}_{j_m,\delta}(y)\right\|\le\delta,$$

and thus $e_m \le L_{j_m}E_{m-1}+\delta$. Since we set $\Lambda_1 = \max\{1, L_{\max}\} \ge 1$, either case gives the bound $e_m \le \Lambda_1 E_{m-1}+\delta$. The cumulative error is bounded by

$$E_m = \max\{E_{m-1}, e_m\} \le \max\{E_{m-1}, \Lambda_1 E_{m-1}+\delta\} = \Lambda_1 E_{m-1}+\delta.$$

Iterating the bound yields

$$E_m \le \delta\sum_{l=0}^{m-1}\Lambda_1^l,$$

and the largest value of the sum is defined as C_1, given by

$$C_1 := \sum_{l=0}^{|\mathcal{V}|-1}\Lambda_1^l.$$

Since $m \le |\mathcal{V}|$ the error is bounded by $E_m \le C_1\delta$, uniformly in m. In particular, if j_{out} is the output node, then the error is

$$\sup_{x\in K}\|f(x)-\mathcal{N}_\delta(x)\| = e_{j_{\text{out}}} \le E_{|\mathcal{V}|} \le C_1\delta.$$

We need to choose the parameters so that the induction and bounds hold. We can set

$$\varepsilon_0 := \min\left\{1, \frac{\rho}{C_1}\right\}, \qquad \delta := \frac{\varepsilon}{C_1}, \qquad \varepsilon\in(0,\varepsilon_0],$$

then $C_1\delta = \varepsilon \le \varepsilon_0 \le \rho$ and the bound $E_m \le C_1\delta \le \rho$ thus closes the induction step and error step, and we have

$$\sup_{x\in K}\|f(x)-\mathcal{N}_\delta(x)\| \le \varepsilon$$

where we set $\mathcal{N}_\varepsilon := \mathcal{N}_\delta$ to finish the construction.

Lastly, we need to show that the network $\mathcal{N}_\varepsilon$ is obtained through realization of the graph and through $\widetilde{\Psi}_{j,\delta}$. Since the DAG is finite and acyclic, the nodes can be evaluated in the topological order. Along every directed path, the network depth is bounded by the sum of the depths of the nodewise networks, up to a fixed constant taking into account any affine (routing) layers. The total number of nonzero weights is at most twice the sum of the node-wise weight counts (Lemma 3.3, Part 2) plus the weights for any of the identity layers (Lemma 3.3, Part 4). Each identity step costs at most four weights per coordinate per layer, and every node-wise network has depth at most $c_j\log(\delta^{-1})$, so the total cost is at most $C\log(\delta^{-1})$ weights, with C depending only on the DAG and the node dimensions. Since every nonlinear node adds at least $c_j\log(\delta^{-1})$ weights, we can include both together by redefining c_j. Therefore, there exist constants $C_D, C_W > 0$, depending only on the DAG and the node dimensions, such that

$$\mathrm{dep}(\mathcal{N}_\varepsilon) \le C_D + \sum_{j\in\mathcal{V}_{\text{nl}}} c_j\log(\delta^{-1}), \qquad W(\mathcal{N}_\varepsilon) \le C_W + \sum_{j\in\mathcal{V}_{\text{nl}}} c_j\,\delta^{-p_j}\left(\log(\delta^{-1})\right)^{q_j}.$$

Since $\delta = \varepsilon/C_1$, we have that $\log(\delta^{-1}) = \log(\varepsilon^{-1}) + \log(C_1)$ and so $\delta^{-p_j} = C_1^{p_j}\varepsilon^{-p_j}$. Again, by redefining of the constants c_j, we obtain

$$\mathrm{dep}(\mathcal{N}_\varepsilon) \le C_D + \sum_{j\in\mathcal{V}_{\text{nl}}} c_j\log(\varepsilon^{-1}), \qquad W(\mathcal{N}_\varepsilon) \le C_W + \sum_{j\in\mathcal{V}_{\text{nl}}} c_j\,\varepsilon^{-p_j}\left(\log(\varepsilon^{-1})\right)^{q_j}.$$

We can remove the constant terms by noting that if $\varepsilon \in (0,1)$, then $\log(\varepsilon^{-1}) \geq 0$ and by replacing ε_0 with $\min\{\varepsilon_0, e^{-1}\}$ we can assume $\log(\varepsilon^{-1}) \geq 1$. Therefore, $C_D \leq C_D \log(\varepsilon^{-1})$ and $\varepsilon^{-p_j} \left(\log(\varepsilon^{-1})\right)^{q_j} \geq 1$, so we can combine the terms into

$$\operatorname{dep}(\mathcal{N}_\varepsilon) \leq \sum_{j \in \mathcal{V}_{\mathrm{nl}}} c_j \log(\varepsilon^{-1}), \qquad W(\mathcal{N}_\varepsilon) \leq \sum_{j \in \mathcal{V}_{\mathrm{nl}}} c_j \, \varepsilon^{-p_j} \left(\log(\varepsilon^{-1})\right)^{q_j}.$$

We note that this bound assumes that there is at least one non-affine node. This completes the proof. □

A.3 Discretization and Complexity

Proof of Proposition 4.1. Fix the time index $q \in \{1, \dots, Q\}$ and decompose the difference over the subintervals:

$$\begin{aligned}
\mathcal{D}[g](t_q) - (\mathcal{D}_Q[g])^q &= \sum_{r=0}^{q-1} \int_{t_r}^{t_{r+1}} \left[S(t_q - s)g(s) - S(t_q - t_r)g(t_r)\right] ds \\
&= \underbrace{\sum_{r=0}^{q-1} \int_{t_r}^{t_{r+1}} S(t_q - s)\left(g(s) - g(t_r)\right) ds}_{=:E_1} + \underbrace{\sum_{r=0}^{q-1} \left(\int_{t_r}^{t_{r+1}} \left[S(t_q - s) - S(t_q - t_r)\right] ds \right) g(t_r)}_{=:E_2}.
\end{aligned} \tag{30}$$

We will repeatedly use the smoothing estimates from Assumption 2.3 in the two forms

$$\|S(v)w\|_{X^\alpha} \leq C v^{-\sigma} \|w\|_{X^\beta}, \qquad \|LS(v)w\|_{X^\alpha} = \|S(v)w\|_{X^{\alpha+2}} \leq C v^{-1-\sigma} \|w\|_{X^\beta}, \tag{31}$$

noting that $\frac{(\alpha+2)-\beta}{2} = 1 + \sigma$.

To bound E_1, we separate the first subinterval from the rest since the weighted modulus degenerates there. For $r = 0$, we use only boundedness so if $q = 1$

$$\left\| \int_0^{\Delta t} S(t_1 - s)\left(g(s) - g(0)\right) ds \right\|_{X^\alpha} \leq 2\|g\|_{\infty,\beta} \int_0^{\Delta t} C(\Delta t - s)^{-\sigma} ds = \frac{2C}{\theta} \|g\|_{\infty,\beta} (\Delta t)^\theta,$$

and if $q \geq 2$ then $t_q - s \geq t_{q-1} \geq \Delta t$ for $s \leq \Delta t$, so the same term is at most $2C\|g\|_{\infty,\beta} \Delta t \, (\Delta t)^{-\sigma} = 2C\|g\|_{\infty,\beta} (\Delta t)^\theta$. For $r \geq 1$ and $s \in [t_r, t_{r+1}]$ the assumptions give $\|g(s) - g(t_r)\|_{X^\beta} \leq [g]_\zeta (\Delta t)^\zeta t_r^{-\zeta}$, thus

$$\begin{aligned}
\left\| \sum_{r=1}^{q-2} \int_{t_r}^{t_{r+1}} S(t_q - s)\left(g(s) - g(t_r)\right) ds \right\|_{X^\alpha} &\leq C[g]_\zeta (\Delta t)^\zeta \sum_{r=1}^{q-2} t_r^{-\zeta} \int_{t_r}^{t_{r+1}} (t_q - s)^{-\sigma} ds \\
&\leq C[g]_\zeta (\Delta t)^{1+\zeta} \sum_{r=1}^{q-2} t_r^{-\zeta} \left(t_q - t_{r+1}\right)^{-\sigma},
\end{aligned} \tag{32}$$

where in the last step we bounded the integrand by its supremum on the subinterval, which is finite because $t_{r+1} < t_q$ for $r \leq q-2$. The last subinterval, $r = q-1$, is estimated directly by $\int_{t_{q-1}}^{t_q} (t_q - s)^{-\sigma} ds = \theta^{-1} (\Delta t)^\theta$, so it adds at most $C[g]_\zeta (\Delta t)^{\zeta+\theta} t_{q-1}^{-\zeta} \leq C[g]_\zeta (\Delta t)^\theta$, since $t_{q-1} \geq \Delta t$ for $q \geq 2$. Substituting $t_r = r\Delta t$ in (32) yields

$$(\Delta t)^{1+\zeta} \sum_{r=1}^{q-2} (r\Delta t)^{-\zeta} \left((q - r - 1)\Delta t\right)^{-\sigma} = (\Delta t)^{1-\sigma} \sum_{r=1}^{q-2} r^{-\zeta} (q - r - 1)^{-\sigma} \leq C (\Delta t)^{1-\sigma} q^{1-\zeta-\sigma},$$

the last inequality uses the discrete Beta function bound which is valid for $\zeta < 1$, $\sigma < 1$ and $\zeta + \sigma = \bar{\sigma} < 1$. Since $q\Delta t = t_q \leq T$,

$$(\Delta t)^{1-\sigma} q^{1-\zeta-\sigma} = t_q^{1-\zeta-\sigma} (\Delta t)^{1-\sigma-(1-\zeta-\sigma)} = t_q^{\theta-\zeta} (\Delta t)^\zeta \leq T^{\theta-\zeta} (\Delta t)^\zeta.$$

Adding up all the terms and using $(\Delta t)^\theta = (\Delta t)^{\theta-\zeta}(\Delta t)^\zeta \le T^{\theta-\zeta}(\Delta t)^\zeta$ for the contribution $C[g]_\zeta(\Delta t)^\theta$ of the term $r = q-1$, we obtain $\|E_1\|_{X^\alpha} \le C\|g\|_{\infty,\beta}(\Delta t)^\theta + C[g]_\zeta T^{\theta-\zeta}(\Delta t)^\zeta$.

Next, for E_2, boundedness is enough. For indices $r \le q-2$, the interval $[t_r, t_{r+1}]$ is at least Δt away from the singularity. We can use that $S(t_q - s) - S(t_q - t_r) = \int_{t_q-s}^{t_q-t_r} LS(v)\,dv$ and the second estimate in (31) to get

$$\left\| \left(\int_{t_r}^{t_{r+1}} \left[S(t_q - s) - S(t_q - t_r) \right] ds \right) g(t_r) \right\|_{X^\alpha} \le C\|g\|_{\infty,\beta} \int_{t_r}^{t_{r+1}} \int_{t_q-t_{r+1}}^{t_q-t_r} v^{-1-\sigma}\,dv\,ds.$$

The inner integral does not depend on s, so the outer integral becomes Δt. As r goes from 0 to $q-2$ the intervals $[t_q - t_{r+1}, t_q - t_r]$ partition $[\Delta t, t_q]$ and so

$$\sum_{r=0}^{q-2} \|\cdots\|_{X^\alpha} \le C\|g\|_{\infty,\beta}\,\Delta t \int_{\Delta t}^{t_q} v^{-1-\sigma}\,dv \le \frac{C}{\sigma}\|g\|_{\infty,\beta}\,\Delta t\,(\Delta t)^{-\sigma} = \frac{C}{\sigma}\|g\|_{\infty,\beta}(\Delta t)^\theta.$$

For the last subinterval where $r = q-1$, we estimate the two terms in E_2 separately, using the first bound in (31):

$$\left\| \int_{t_{q-1}}^{t_q} S(t_q - s)ds\, g(t_{q-1}) \right\|_{X^\alpha} \le C\|g\|_{\infty,\beta} \int_0^{\Delta t} v^{-\sigma} dv = \frac{C}{\theta}\|g\|_{\infty,\beta}(\Delta t)^\theta,$$

and

$$\left\|\Delta t\, S(\Delta t) g(t_{q-1})\right\|_{X^\alpha} \le C\|g\|_{\infty,\beta}\Delta t\,(\Delta t)^{-\sigma} = C\|g\|_{\infty,\beta}(\Delta t)^\theta.$$

Thus $\|E_2\|_{X^\alpha} \le C\|g\|_{\infty,\beta}(\Delta t)^\theta$, with C depending only on σ.

Adding the two bounds together and taking the maximum completes the first part, where both sides are zero when $q = 0$.

For the second part of the proposition, write the error as $E_1 + E_2$ as in (30), with integration variable τ. Since the modulus of g does not use the weight, the subintervals do not need to be separated. For E_1, the assumed bound $\|g(\tau) - g(t_r)\|_{X^\beta} \le H(\Delta t)^\kappa$ and the smoothing gives

$$\|E_1\|_{X^\alpha} \le CH(\Delta t)^\kappa \int_0^{t_q} (t_q - \tau)^{-\sigma} d\tau \le CT^\theta H(\Delta t)^\kappa.$$

For E_2, write $S(t_q - \tau) - S(t_q - t_r) = S(t_q - \tau)(I - S(\delta))$ with $\delta = \tau - t_r \le \Delta t$. Let $s := \beta' - 2\kappa$. A spectral argument yields

$$\|(I - S(\delta))w\|_{X^s} \le \delta^\kappa \|w\|_{X^{s+2\kappa}} = \delta^\kappa \|w\|_{X^{\beta'}},$$

and the smoothing bound from X^s to X^α includes a $(t_q - \tau)^{-a}$ term, where $a = (\alpha - s)_+/2 = \left(\sigma - \frac{\beta'-\beta}{2} + \kappa\right)_+$. Therefore,

$$\|E_2\|_{X^\alpha} \le CG(\Delta t)^\kappa \int_0^{t_q} (t_q - \tau)^{-a} d\tau \le \frac{C}{1-a} T^{1-a} G(\Delta t)^\kappa,$$

and the integral converging because $a < 1$, which is the condition $\kappa < \theta + \frac{\beta'-\beta}{2}$.

If $s < 0$, X^s is the space from Definition 4.15, and both spectral bounds still hold. If $s > \alpha$, then $a = 0$ and the bound uses $\|w\|_{X^\alpha} \le \lambda_1^{-(s-\alpha)/2}\|w\|_{X^s}$, so C also depends on λ_1. □

Proof of Proposition 4.10. With the exact values, the encoder gives $D_{\text{an}}^{\text{ex}}\mathcal{T}_{\text{an}} w = \mathcal{C}_N^h w$, the synthesis step gives $\mathcal{S}_N^h a^r$ and $\nabla_h \mathcal{S}_N^h a^r$, and the analysis step gives $\mathcal{C}_N^h G^r$, using Assumption 4.5. Writing out the recurrence steps yields $v^q = S_N(\Delta t)^q a_0^h + \Delta t \sum_{r<q} S_N(\Delta t)^{q-r} \hat{g}^r$, which is step (2d) since $S_N(t_q - t_r) = S_N(\Delta t)^{q-r}$. Lastly, $S_N(\Delta t)^{\tau/\Delta t} = S_N(\tau)$, since $S_N(\Delta t)$ is diagonal with positive entries, so the output agrees with Definition 4.12. □

Proof of Proposition 4.13. Take $t \in [t_q, t_{q+1}]$, and set the interpolation weight to $s_0 := \frac{t-t_q}{t_{q+1}-t_q} \in [0,1]$ which yields the piecewise linear interpolant in time:

$$\mathcal{L}_{N,Q} a(t) = (1 - s_0)\mathcal{S}_N a^q + s_0 \mathcal{S}_N a^{q+1}.$$

By the triangle inequality and homogeneity of the X^α-norm,

$$\|\mathcal{L}_{N,Q}a(t)\|_{X^\alpha} \le \max_{0\le j\le Q} \|\mathcal{S}_N a^j\|_{X^\alpha}.$$

The supremum over $t \in [0,T]$ yields the bound

$$\|\mathcal{L}_{N,Q}a\|_{X_T^\alpha} \le \max_{0\le j\le Q} \|\mathcal{S}_N a^j\|_{X^\alpha} = \max_{0\le j\le Q} \|a^j\|_{\ell^2_\alpha},$$

where we used that the discrete norm and function norms are equal.

Next, since $\mathcal{L}_{N,Q}a(t_j) = \mathcal{S}_N a^j$, we have the lower bound

$$\|\mathcal{L}_{N,Q}a\|_{X_T^\alpha} = \sup_{t\in[0,T]} \|\mathcal{L}_{N,Q}a(t)\|_{X^\alpha} \ge \max_{0\le j\le Q} \|\mathcal{S}_N a^j\|_{X^\alpha}.$$

Using both inequalities shows that the norms are equal, and the rest of the claim follows. □

Proof of Corollary 4.14. By Proposition 4.13, the lifting $u_a := \mathcal{L}_{N,Q}a$ is in B_R^α and $\mathcal{S}_N a^q = u_a(t_q)$. Thus $((\mathcal{S}_N^h a^q)_i, (\nabla_h \mathcal{S}_N^h a^q)_i) = (u_a(t_q,x_i), \nabla u_a(t_q,x_i))$, which lies in K by definition. □

Proof of Lemma 4.17. The point-wise values $v(x_i)$ are well defined by Assumption 2.2. Since $\mathcal{S}_N$ is an isometry from $(\mathbb{R}^N, \|\cdot\|_{\ell^2_\alpha})$ onto $(E_N, \|\cdot\|_{X^\alpha})$, duality on the finite-dimensional space E_N gives

$$\left\|\Pi_{N,h}v\right\|_{X^\alpha} = \sup_{\substack{\phi\in E_N \\ \|\phi\|_{X^{-\alpha}}\le 1}} \left|\left\langle \mathcal{S}_N\mathcal{C}_N^h\mathcal{I}_h v, \phi\right\rangle\right| = \sup_{\substack{\phi\in E_N \\ \|\phi\|_{X^{-\alpha}}\le 1}} \left|\sum_{i=1}^{M(h)} \omega_i^{N,h} v(x_i)\phi(x_i)\right| \le C_{\mathrm{stab}}^\alpha \|v\|_{X^\alpha},$$

using the $s = \alpha$ case from Assumption 4.16. □

Proof of Theorem 4.20. We first derive a discrete semigroup estimate. Let $v \in X^\beta \cap C(\overline{\Omega})$ and $\tau > 0$. Since $\mathcal{S}_N$ is an isometry from $(\mathbb{R}^N, \|\cdot\|_{\ell^2_\alpha})$ onto $(E_N, \|\cdot\|_{X^\alpha})$, the following norms are equal:

$$\|S_N(\tau)\mathcal{C}_N^h v\|_{\ell^2_\alpha} = \|\mathcal{S}_N S_N(\tau)\mathcal{C}_N^h v\|_{X^\alpha}.$$

Using duality of the X^α norm on E_N, we can characterize the norm by:

$$\|\mathcal{S}_N S_N(\tau)\mathcal{C}_N^h v\|_{X^\alpha} = \sup_{\substack{\phi\in E_N \\ \|\phi\|_{X^{-\alpha}}\le 1}} \left|\left\langle \mathcal{S}_N S_N(\tau)\mathcal{C}_N^h v, \phi\right\rangle\right|.$$

The semigroup satisfies $S(\tau)\mathcal{S}_N = \mathcal{S}_N S_N(\tau)$ and is self-adjoint, so

$$\left\langle \mathcal{S}_N S_N(\tau)\mathcal{C}_N^h v, \phi\right\rangle = \left\langle S(\tau)\mathcal{S}_N\mathcal{C}_N^h v, \phi\right\rangle = \left\langle \mathcal{S}_N\mathcal{C}_N^h v, S(\tau)\phi\right\rangle = \sum_{i=1}^{M(h)} \omega_i^{N,h} v(x_i)(S(\tau)\phi)(x_i),$$

using the quadrature rule and the definitions of $\mathcal{C}_N^h$ and $\mathcal{S}_N$. Since $\phi \in E_N$ and $S(\tau)E_N \subseteq E_N$, using Assumption 4.16 leads to

$$\|S_N(\tau)\mathcal{C}_N^h v\|_{\ell^2_\alpha} \le C_{\mathrm{stab}} \|v\|_{X^\beta} \sup_{\substack{\phi\in E_N \\ \|\phi\|_{X^{-\alpha}}\le 1}} \|S(\tau)\phi\|_{X^{-\beta}} \le C C_{\mathrm{stab}} \tau^{-(\alpha-\beta)/2} \|v\|_{X^\beta},$$

where we used the semigroup property for negative spaces and the supremum.

Next, we prove a bound on the fully discrete map. Recall that $a = (a^0, \dots, a^Q) \in B_R^{N,Q}$, and the continuous variable is defined by the lifting $u_a := \mathcal{L}_{N,Q}a$. Proposition 4.13 implies that $\|u_a\|_{X_T^\alpha} = \|a\|_{X_{N,Q}^\alpha} \le R$, so $u_a \in B_R^\alpha$. For each $q = 0, \dots, Q$, define the continuous reconstruction $\widetilde{u}_a^q := \mathcal{S}_N a^q \in E_N$, whose values

and derivatives at the sample points are $\mathcal{S}_N^h a^q = \mathcal{I}_h \widetilde{u}_a^q$ and $\nabla_h \mathcal{S}_N^h a^q = \mathcal{I}_h^{\mathrm{grad}} \widetilde{u}_a^q$. We define the continuous forcing function by

$$F_a^q(x) := f\left(\widetilde{u}_a^q(x), \nabla \widetilde{u}_a^q(x)\right),$$

whose value at the sample point x_i is exactly

$$F_a^q(x_i) = f\left((\mathcal{S}_N^h a^q)_i, (\nabla_h \mathcal{S}_N^h a^q)_i\right).$$

Therefore, applying the discrete coefficient map gives

$$\mathcal{C}_N^h \mathcal{I}_h F_a^q = \mathcal{C}_N^h \left[\mathcal{F}^h\left(\mathcal{S}_N^h a^q, \nabla_h \mathcal{S}_N^h a^q\right)\right],$$

which are the values inside the discrete fixed-point map. Since $\widetilde{u}_a^q = u_a(t_q)$, the force is also equal to $F_a^q = f\left(u_a(t_q), \nabla u_a(t_q)\right)$ and so by assumption $\|F_a^q\|_{X^\beta} \le F_R$.

Taking norms in the definition of $\Psi_{N,Q}^h$ leads to

$$\begin{aligned}\|(\Psi_{N,Q}^h(a))^q\|_{\ell^2_\alpha} &\le \|a_0^h\|_{\ell^2_\alpha} + \Delta t \sum_{r=0}^{q-1} \|S_N(t_q - t_r)\mathcal{C}_N^h \mathcal{I}_h F_a^r\|_{\ell^2_\alpha} \\ &\le \|a_0^h\|_{\ell^2_\alpha} + C C_{\mathrm{stab}} F_R \Delta t \sum_{r=0}^{q-1} (t_q - t_r)^{-(\alpha-\beta)/2},\end{aligned}$$

where we used that $S_N(t)$ is contractive on ℓ^2_α and the discrete semigroup estimate above. For exponents $\sigma = \frac{\alpha-\beta}{2} < 1$ and $\theta = 1 - \sigma$, and recalling that $t_q - t_r = (q-r)\Delta t$, we have

$$\Delta t \sum_{r=0}^{q-1} (t_q - t_r)^{-\sigma} = \Delta t^{1-\sigma} \sum_{r=0}^{q-1} (q-r)^{-\sigma} = \Delta t^{1-\sigma} \sum_{j=1}^{q} j^{-\sigma} \le (1-\sigma)^{-1} (q\Delta t)^{1-\sigma} \le (1-\sigma)^{-1} T^\theta.$$

We redefine the constant C to absorb the σ terms, thus yielding

$$\|(\Psi_{N,Q}^h(a))^q\|_{\ell^2_\alpha} \le \|a_0^h\|_{\ell^2_\alpha} + C C_{\mathrm{stab}} F_R T^\theta \quad \Longrightarrow \quad \|\Psi_{N,Q}^h(a)\|_{X_{N,Q}^\alpha} \le \|a_0^h\|_{\ell^2_\alpha} + C C_{\mathrm{stab}} F_R T^\theta.$$

To show that the discrete map is contractive, take $a, b \in B_R^{N,Q}$, and let $u_a := \mathcal{L}_{N,Q} a$, $u_b := \mathcal{L}_{N,Q} b$, $\widetilde{u}_a^q := \mathcal{S}_N a^q$, $\widetilde{u}_b^q := \mathcal{S}_N b^q$. In the same way as before, $\mathcal{S}_N^h a^q = \mathcal{I}_h \widetilde{u}_a^q$ and $\mathcal{S}_N^h b^q = \mathcal{I}_h \widetilde{u}_b^q$, and we define F_a^q and F_b^q accordingly. Since $\widetilde{u}_a^q = u_a(t_q)$ and $\widetilde{u}_b^q = u_b(t_q)$, we can apply the local Lipschitz assumption and the definition of the discrete norm to bound

$$\|F_a^q - F_b^q\|_{X^\beta} \le L_R \|u_a(t_q) - u_b(t_q)\|_{X^\alpha} = L_R \|a^q - b^q\|_{\ell^2_\alpha} \le L_R \|a - b\|_{X_{N,Q}^\alpha}.$$

The difference of the full maps is bounded by

$$\begin{aligned}\|(\Psi_{N,Q}^h(a))^q - (\Psi_{N,Q}^h(b))^q\|_{\ell^2_\alpha} &= \Delta t \left\| \sum_{r=0}^{q-1} S_N(t_q - t_r) \mathcal{C}_N^h \mathcal{I}_h \left(F_a^r - F_b^r\right) \right\|_{\ell^2_\alpha} \\ &\le C C_{\mathrm{stab}} L_R \Delta t \sum_{r=0}^{q-1} (t_q - t_r)^{-(\alpha-\beta)/2} \|a - b\|_{X_{N,Q}^\alpha} \\ &\le C C_{\mathrm{stab}} L_R T^\theta \|a - b\|_{X_{N,Q}^\alpha}.\end{aligned}$$

Taking the maximum over q gives

$$\|\Psi_{N,Q}^h(a) - \Psi_{N,Q}^h(b)\|_{X_{N,Q}^\alpha} \le \kappa \|a - b\|_{X_{N,Q}^\alpha},$$

where $\kappa := CC_{\text{stab}} L_R T^\theta$ is independent of N, h, and Q.

It follows directly that if $\|a_0^h\|_{\ell^2_{\vec{\alpha}}} + CC_{\text{stab}} F_R T^\theta \le R$, then $\Psi^h_{N,Q}(B_R^{N,Q}) \subseteq B_R^{N,Q}$. In addition, if $\kappa < 1$, then $\Psi^h_{N,Q}$ is a contraction on the complete set $B_R^{N,Q}$, and thus by the Banach fixed-point theorem there exists a unique fixed point $a^h_{N,Q} \in B_R^{N,Q}$. □

Proof of Theorem 4.22. Recall that $\sigma = \frac{\alpha-\beta}{2}$ and $\theta := 1 - \sigma$, so that the conditions are $0 < \sigma < 1$ and $\theta > 0$. Define the following functions:

$$F(t) := f(u(t), \nabla u(t)), \qquad F_N(t) := f(P_N u(t), \nabla P_N u(t)), \qquad \widetilde{u}^h_N(t) := \mathcal{S}_N \mathcal{C}^h_N \mathcal{I}_h u(t).$$

The proof uses two (discrete) semigroup estimates. The first estimate was derived at the beginning of the proof of Theorem 4.20, i.e., for any $v \in X^\beta \cap C(\overline{\Omega})$ and $\tau > 0$, we have

$$\left\| \mathcal{S}_N S_N(\tau) \mathcal{C}^h_N v \right\|_{X^\alpha} \le C C^\beta_{\text{stab}} \tau^{-\sigma} \|v\|_{X^\beta}.$$

The second estimate uses Lemma 4.17 and the contractivity of $S_N(\tau)$ to get:

$$\|\mathcal{S}_N S_N(\tau) \mathcal{C}^h_N v\|_{X^\alpha} \le C^\alpha_{\text{stab}} \|v\|_{X^\alpha}$$

for any $\tau \ge 0$. Specifically, this shows that $\|\widetilde{u}^h_N(t)\|_{X^\alpha} \le C^\alpha_{\text{stab}} R$. Therefore $\widetilde{u}^h_N(t)$ and $P_N u(t)$ lie in a fixed bounded subset of X^α, uniformly in N, h and t, and the Lipschitz constant of f on this set can be absorbed by C_R.

We estimate directly the difference between one step of the fully discrete map and the solution u. Pick an index $q \in \{0, \dots, Q\}$, then

$$\begin{aligned}\left(\Phi^h_{N,Q}[u]\right)(t_q) - u(t_q) =& S(t_q)\mathcal{S}_N a^h_0 - S(t_q)u_0 + \Delta t \sum_{l=0}^{q-1} S(t_q - t_l)\mathcal{S}_N \mathcal{C}^h_N \mathcal{I}_h \left[f(\widetilde{u}^h_N(t_l), \nabla \widetilde{u}^h_N(t_l)) \right] \\ &- \int_0^{t_q} S(t_q - s) F(s)\, ds.\end{aligned}$$

Since $\mathcal{S}_N \mathcal{C}_N = P_N$, we have

$$S(t_q)\mathcal{S}_N a^h_0 - S(t_q)u_0 = S(t_q)\mathcal{S}_N \left(a^h_0 - \mathcal{C}_N u_0 \right) + S(t_q)(P_N - I)u_0.$$

We decompose the difference into several parts:

$$\left(\Phi^h_{N,Q}[u]\right)(t_q) - u(t_q) = E_{\text{init},q} + E_{\text{Gal},q} + E_{\text{nl},q} + E_{\text{quad},q} + E_{\text{proj},q} + E_{\text{time},q},$$

where each term represents a specific approximation step:

$$E_{\text{init},q} := S(t_q)\mathcal{S}_N \left(a^h_0 - \mathcal{C}_N u_0 \right),$$

$$E_{\text{Gal},q} := S(t_q)(P_N - I)u_0,$$

$$E_{\text{nl},q} := \Delta t \sum_{l=0}^{q-1} S(t_q - t_l)\mathcal{S}_N \mathcal{C}^h_N \mathcal{I}_h \left[f(\widetilde{u}^h_N(t_l), \nabla \widetilde{u}^h_N(t_l)) - F_N(t_l) \right],$$

$$E_{\text{quad},q} := \Delta t \sum_{l=0}^{q-1} S(t_q - t_l)\mathcal{S}_N \left(\mathcal{C}^h_N \mathcal{I}_h F_N(t_l) - \mathcal{C}_N F_N(t_l) \right),$$

$$E_{\text{proj},q} := \Delta t \sum_{l=0}^{q-1} S(t_q - t_l) \left(P_N F_N(t_l) - F(t_l) \right),$$

and

$$E_{\text{time},q} := \Delta t \sum_{l=0}^{q-1} S(t_q - t_l)F(t_l) - \int_0^{t_q} S(t_q - s)F(s)\,ds.$$

Each term can be bounded separately. When $q = 0$, all sums are set to zero, so in the bounds that deal with singular kernel, we will assume $q \geq 1$.

First, by the contraction property of $S(t)$ on X^α and since $\mathcal{S}_N$ is an isometry, we have:

$$\|E_{\text{init},q}\|_{X^\alpha} \leq \|a_0^h - \mathcal{C}_N u_0\|_{\ell^2_\alpha} = E^h_{\text{init}}, \qquad \|E_{\text{Gal},q}\|_{X^\alpha} \leq \|(I - P_N)u_0\|_{X^\alpha}.$$

Next, consider $E_{\text{nl},q}$, which measures the effect of evaluating the law at encoded samples $\widetilde{u}_N^h$ instead of $P_N u$. Since $P_N = \mathcal{S}_N\mathcal{C}_N$, we have

$$\widetilde{u}_N^h(t) - P_N u(t) = \mathcal{S}_N\left(\mathcal{C}_N^h\mathcal{I}_h u(t) - \mathcal{C}_N u(t)\right),$$

so by the isometry of $\mathcal{S}_N$ and by Assumption 4.18 with $s = \alpha$, we get

$$\|\widetilde{u}_N^h(t) - P_N u(t)\|_{X^\alpha} = \left\|\mathcal{C}_N^h\mathcal{I}_h u(t) - \mathcal{C}_N u(t)\right\|_{\ell^2_\alpha} \leq C_{\text{quad}} h^r \|u(t)\|_{X^{\alpha+r}}.$$

For $t > 0$, since $r \leq \gamma - \alpha$ when $\gamma_{\text{s}} = \gamma$, the continuous embedding gives $\|u(t)\|_{X^{\alpha+r}} \leq C\|u(t)\|_{X^\gamma}$. The bootstrapped case where r can be larger than $\gamma - \alpha$ is left for the end of the proof. By the local Lipschitz continuity of f,

$$\left\|f(\widetilde{u}_N^h(t), \nabla\widetilde{u}_N^h(t)) - F_N(t)\right\|_{X^\beta} \leq C_R h^r \|u(t)\|_{X^\gamma},$$

for all $t > 0$, where the Lipschitz constant is absorbed in the constant above. At $t = 0$, we use boundedness

$$\|f(\widetilde{u}_N^h(0), \nabla\widetilde{u}_N^h(0)) - F_N(0)\|_{X^\beta} \leq C_R.$$

The $l = 0$ contribution thus gives

$$\|E^{(0)}_{\text{nl},q}\|_{X^\alpha} \leq C_R \Delta t\, t_q^{-\sigma} \leq C_R(\Delta t)^{1-\sigma} = C_R(\Delta t)^\theta,$$

since $t_q \geq \Delta t$ and the others terms give

$$\|E^{(>0)}_{\text{nl},q}\|_{X^\alpha} \leq C_R h^r \Delta t \sum_{l=1}^{q-1} (t_q - t_l)^{-\sigma}\|u(t_l)\|_{X^\gamma}.$$

We set the exponents to $\zeta := \frac{\gamma-\alpha}{2}$ and $\bar{\sigma} := \frac{\gamma-\beta}{2}$, so $0 < \zeta < 1$, $\bar{\sigma} < 1$, and $\sigma + \zeta = \bar{\sigma}$. By Proposition 2.6, $\|u(t)\|_{X^\gamma} \leq C_R t^{-\zeta} + C_R t^{1-\bar{\sigma}}$, and thus we will have contributions from both terms within the sum. For the first contribution, when $q \geq 2$,

$$\Delta t \sum_{l=1}^{q-1} (t_q - t_l)^{-\sigma} t_l^{-\zeta} = (\Delta t)^{1-\sigma-\zeta} \sum_{l=1}^{q-1} (q-l)^{-\sigma} l^{-\zeta}.$$

We split the sum at $l = \lfloor q/2 \rfloor$ to get

$$\sum_{l=1}^{q-1} (q-l)^{-\sigma} l^{-\zeta} \leq Cq^{-\sigma} \sum_{l=1}^{\lfloor q/2 \rfloor} l^{-\zeta} + Cq^{-\zeta} \sum_{j=1}^{\lceil q/2 \rceil} j^{-\sigma} \leq Cq^{1-\sigma-\zeta},$$

since $0 < \zeta < 1$ and $0 < \sigma < 1$. So the first sum is bounded by

$$\Delta t \sum_{l=1}^{q-1} (t_q - t_l)^{-\sigma} t_l^{-\zeta} \leq C(\Delta t\, q)^{1-\sigma-\zeta} = Ct_q^{1-\bar{\sigma}} \leq CT^{1-\bar{\sigma}}.$$

The second sum is bounded by

$$\Delta t \sum_{l=1}^{q-1} (t_q - t_l)^{-\sigma} t_l^{1-\bar{\sigma}} \le T^{1-\bar{\sigma}} \Delta t \sum_{l=1}^{q-1} (t_q - t_l)^{-\sigma} = T^{1-\bar{\sigma}} (\Delta t)^\theta \sum_{j=1}^{q-1} j^{-\sigma} \le C T^{1-\bar{\sigma}} t_q^\theta \le C T^{1-\bar{\sigma}+\theta}.$$

And together the max error for the nonlinear reconstruction is $\max_{0\le q\le Q} \|E_{\mathrm{nl},q}\|_{X^\alpha} \le C_R(h^r + (\Delta t)^\theta)$.

For the spatial quadrature error, set $d_l := \mathcal{C}_N^h \mathcal{I}_h F_N(t_l) - \mathcal{C}_N F_N(t_l)$, and thus the error is

$$E_{\mathrm{quad},q} = \Delta t \sum_{l=0}^{q-1} S(t_q - t_l)\, \mathcal{S}_N d_l.$$

Since $\mathcal{S}_N$ is an isometry from $(\mathbb{R}^N, \|\cdot\|_{\ell^2_\beta})$ to $(E_N, \|\cdot\|_{X^\beta})$, the smoothing from Assumption 2.3 gives

$$\|S(\tau)\mathcal{S}_N d_l\|_{X^\alpha} \le C\tau^{-\sigma} \|d_l\|_{\ell^2_\beta}, \tag{33}$$

for all $\tau > 0$. For every node with $t_l > 0$, Assumption 4.18 with $s = \beta$ and the weighted hypothesis on F_N give

$$\|d_l\|_{\ell^2_\beta} \le C_{\mathrm{quad}} h^r \|F_N(t_l)\|_{X^{\beta+r}} \le C_R h^r\, t_l^{-\varrho}. \tag{34}$$

The rest needs to deal with $l = 0$ and the sum, which takes several cases.

First, if $\varrho = 0$ we use the bootstrap. Note that (34) also holds at $l = 0$. This is due to lower semicontinuity, in particular, $F_N(t) \to F_N(0)$ in X^β as $t \to 0^+$, by (3) and the continuity of u in X^α, so the coefficients of $F_N(t)$ converge to those of $F_N(0)$. Lower semicontinuity of the $X^{\beta+r}$ norm for the coefficient convergence follows from Fatou's lemma applied to the series $\|v\|^2_{X^{\beta+r}} = \sum_k \lambda_k^{\beta+r} |\langle v, e_k\rangle|^2$ (see also Lemma 6.2) and yields $\|F_N(0)\|_{X^{\beta+r}} \le \liminf_{t\to 0^+} \|F_N(t)\|_{X^{\beta+r}} \le C_R$. Putting (34) into (33) for every l gives

$$\|E_{\mathrm{quad},q}\|_{X^\alpha} \le C_R h^r\, \Delta t \sum_{l=0}^{q-1} (t_q - t_l)^{-\sigma} = C_R h^r (\Delta t)^\theta \sum_{j=1}^{q} j^{-\sigma} \le \frac{C_R}{\theta} h^r\, t_q^\theta \le C_R h^r.$$

Next, for the case $\varrho = r/2$, which is the case without the bootstrap, $\gamma_{\mathrm{s}} = \gamma$. Since $r \le \gamma_{\mathrm{s}} - \alpha = \gamma - \alpha = 2\zeta$ by (7), we have $\varrho \le \zeta$ and $\sigma + \varrho \le \bar{\sigma} < 1$. For the nodes $l \ge 1$, (33) and (34) give

$$\Delta t \sum_{l=1}^{q-1} \left\|S(t_q - t_l)\mathcal{S}_N d_l\right\|_{X^\alpha} \le C_R h^r\, \Delta t \sum_{l=1}^{q-1} (t_q - t_l)^{-\sigma} t_l^{-\varrho} \le C_R h^r\, T^{1-\sigma-\varrho},$$

using the weighted convolution bound for $E_{\mathrm{nl},q}$, applied with ζ replaced by ϱ. The node $l = 0$ is estimated without using the quadrature consistency assumption, since the weight $t_l^{-\varrho}$ in (34) is not finite. Using $S(t_q)\mathcal{S}_N = \mathcal{S}_N S_N(t_q)$ and $\mathcal{S}_N \mathcal{C}_N = P_N$, we split d_0 into two parts:

$$\Delta t \left\|S(t_q)\mathcal{S}_N d_0\right\|_{X^\alpha} \le \Delta t \left\|\mathcal{S}_N S_N(t_q) \mathcal{C}_N^h \mathcal{I}_h F_N(0)\right\|_{X^\alpha} + \Delta t \left\|S(t_q) P_N F_N(0)\right\|_{X^\alpha} \le C \left(C_{\mathrm{stab}}^\beta + 1\right) \Delta t\, t_q^{-\sigma} \|F_N(0)\|_{X^\beta}.$$

The first term is bounded by the discrete smoothing estimate stated at the start of this proof. The second is bounded by Assumption 2.3 and since P_N is a contraction on X^β. Since $\|P_N u_0\|_{X^\alpha} \le \|u_0\|_{X^\alpha} \le R$, we have $\|F_N(0)\|_{X^\beta} \le F_R$, and $t_q \ge \Delta t$ gives $\Delta t\, t_q^{-\sigma} \le (\Delta t)^\theta$. Thus it adds at most $C_R(\Delta t)^\theta$.

In both cases,

$$\max_{0\le q\le Q} \|E_{\mathrm{quad},q}\|_{X^\alpha} \le C_R \left(h^r + (\Delta t)^\theta\right),$$

where the term $(\Delta t)^\theta$ appears only when $\varrho = r/2$. In that case there is no bootstrap, so $\zeta_{\mathrm{t}} = \zeta$, and $(\Delta t)^\theta \le T^{\theta-\zeta}(\Delta t)^\zeta$ is absorbed into the temporal error term of the theorem.

Next for the force projection error, we decompose it using $P_N F_N - F = P_N(F_N - F) + (P_N - I)F$, and consider the first part. This leads to three bounds that are needed.

First, using the Lipschitz condition from (3) and the fact that P_N is an orthogonal projection and contraction on X^α (so both inputs are B_R^α), we have

$$\|F_N(t_l) - F(t_l)\|_{X^\beta} \le L_R \, \|P_N u(t_l) - u(t_l)\|_{X^\alpha} . \tag{35}$$

Second, that projection measured in X^α is the tail bound from Proposition 2.12 with regularity exponent γ and thus we have

$$\|P_N u(t_l) - u(t_l)\|_{X^\alpha} \le \lambda_{N+1}^{-\zeta} \, \|u(t_l)\|_{X^\gamma} \le C_R \, \lambda_{N+1}^{-\zeta} \, t_l^{-\zeta}, \tag{36}$$

where in the last inequality we apply Lemma 2.7 Part 2 for $l > 0$. At $l = 0$, we use instead the boundedness $\|F_N(0) - F(0)\|_{X^\beta} \le C_R$. For the third bound, the resulting X^β bounds from the previous steps are in X^β but we did not include the semigroup part of the projection error, which lifts back to X^α. By Assumption 2.3 and since P_N is a contraction on X^β, the semigroup smooths by:

$$\|S(\tau) P_N v\|_{X^\alpha} \le C\tau^{-\sigma} \|v\|_{X^\beta}. \tag{37}$$

Putting all three steps together, the $l > 0$ terms gives us

$$\Delta t \sum_{l=1}^{q-1} \left\| S(t_q - t_l) P_N \left(F_N(t_l) - F(t_l) \right) \right\|_{X^\alpha} \le C_R \lambda_{N+1}^{-\zeta} \, \Delta t \sum_{l=1}^{q-1} (t_q - t_l)^{-\sigma} t_l^{-\zeta} \le C_R T^{1-\bar{\sigma}} \lambda_{N+1}^{-\zeta},$$

the last step is the weighted convolution estimate shown in $E_{\mathrm{nl},q}$, which converges since $\sigma + \zeta = \bar{\sigma} < 1$. The term $l = 0$ cannot use (36), and is estimated through boundedness:

$$\Delta t \left\| S(t_q) P_N \left(F_N(0) - F(0) \right) \right\|_{X^\alpha} \le C_R \, \Delta t \, t_q^{-\sigma} \le C_R (\Delta t)^{1-\sigma} = C_R (\Delta t)^\theta,$$

where $t_q \ge \Delta t$, with the worst case when $q = 1$. Thus,

$$\max_{0 \le q \le Q} \left\| \Delta t \sum_{l=0}^{q-1} S(t_q - t_l) P_N \left(F_N(t_l) - F(t_l) \right) \right\|_{X^\alpha} \le C_R \left(\lambda_{N+1}^{-\zeta} + (\Delta t)^\theta \right) \le C_R \left(\lambda_{N+1}^{-\zeta} + (\Delta t)^\zeta \right),$$

where T is absorbed in the constants and the last inequality used $(\Delta t)^\theta \le T^{\theta - \zeta} (\Delta t)^\zeta$.

The second part of the projection error measures the tail $(I - P_N)F$. For $\tau > 0$ and $v \in X^\beta$,

$$\|S(\tau)(I - P_N) v\|_{X^\alpha}^2 = \sum_{k > N} \lambda_k^{\alpha - \beta} e^{-2\lambda_k \tau} \lambda_k^\beta |v_k|^2.$$

For any fixed $c \in (0,1)$, $\lambda_k^\sigma e^{-\lambda_k \tau} = \tau^{-\sigma} (\lambda_k \tau)^\sigma e^{-(1-c)\lambda_k \tau} e^{-c\lambda_k \tau}$. Since the sup is bounded, $\sup_{x \ge 0} x^\sigma e^{-(1-c)x} < \infty$, and $\lambda_k \ge \lambda_{N+1}$, we have $\lambda_k^\sigma e^{-\lambda_k \tau} \le C\tau^{-\sigma} e^{-c\lambda_{N+1}\tau}$, and thus

$$\|S(\tau)(I - P_N) v\|_{X^\alpha} \le C\tau^{-\sigma} e^{-c\lambda_{N+1}\tau} \|v\|_{X^\beta}.$$

Using the bound on F in X^β (uniform) we have

$$\left\| \Delta t \sum_{l=0}^{q-1} S(t_q - t_l)(I - P_N) F(t_l) \right\|_{X^\alpha} \le C_R \Delta t \sum_{j=1}^{q} (j\Delta t)^{-\sigma} e^{-c\lambda_{N+1} j \Delta t}.$$

The function $g(s) = s^{-\sigma} e^{-c\lambda s}$ is positive and decreasing on $\mathbb{R}_+$ so we use the integral bound

$$\Delta t \sum_{j=1}^{\infty} (j\Delta t)^{-\sigma} e^{-c\lambda j \Delta t} \le \int_0^\infty s^{-\sigma} e^{-c\lambda s} \, ds = C\lambda^{\sigma - 1},$$

which shows

$$\max_{0 \le q \le Q} \left\| \Delta t \sum_{l=0}^{q-1} S(t_q - t_l)(I - P_N) F(t_l) \right\|_{X^\alpha} \le C_R \lambda_{N+1}^{-\theta}.$$

Since $\gamma - \beta < 2$, we have $\zeta = \frac{\gamma-\alpha}{2} < 1 - \frac{\alpha-\beta}{2} = \theta$. Since $\zeta < \theta$ and $\lambda_{N+1} \geq \lambda_1$, we have $\lambda_{N+1}^{-\theta} = \lambda_{N+1}^{-\zeta}\lambda_{N+1}^{-(\theta-\zeta)} \leq \lambda_1^{-(\theta-\zeta)}\lambda_{N+1}^{-\zeta}$ and therefore,

$$\max_{0\leq q\leq Q} \|E_{\text{proj},q}\|_{X^\alpha} \leq C_R \left[\lambda_{N+1}^{-(\gamma-\alpha)/2} + (\Delta t)^\zeta\right],$$

where the constants absorb λ_1 factors only.

The last error term is the temporal quadrature error. Using Proposition 4.1 with $g = F$ gives

$$\max_{0\leq q\leq Q} \|E_{\text{time},q}\|_{X^\alpha} \leq C\left(\|F\|_{\infty,\beta}(\Delta t)^\theta + [F]_\zeta T^{\theta-\zeta}(\Delta t)^\zeta\right) \leq C_R(\Delta t)^\zeta,$$

with $\zeta < \theta$. Combining all the previous bounds proves the first total bound.

For the second result in the theorem, denote the piece-wise linear interpolant by $\mathcal{J}_Q u$. On each subinterval $[t_q, t_{q+1}]$, both $\Phi_{N,Q}^h[u]$ and $\mathcal{J}_Q u$ are affine functions in time. So taking the line segment $t = (1-s_0)t_q + s_0 t_{q+1}$ with $s_0 \in [0,1]$, the norm is the max over endpoints:

$$\left\|\Phi_{N,Q}^h[u](t) - \mathcal{J}_Q u(t)\right\|_{X^\alpha} \leq \max_{j=q,q+1}\left\|\Phi_{N,Q}^h[u](t_j) - u(t_j)\right\|_{X^\alpha}.$$

Taking the sup over each interval and then the maximum over q gives the second results and completes the proof.

If $\gamma_s = \gamma' > \gamma$, Lemma 2.9 gives, for every $t \in [0,T]$

$$\|u(t)\|_{X^{\gamma_s}} \leq R_{\gamma'}, \qquad \|F(t)\|_{X^{\gamma_s-(\alpha-\beta)}} \leq C_R.$$

Since these bounds hold at $t = 0$, we avoid the previous issues at $l = 0$. For brevity, the following is the list of estimates that change:

1. The Galerkin term bound is now $\|(I - P_N)u_0\|_{X^\alpha} \leq \lambda_{N+1}^{-\zeta_s}\|u_0\|_{X^{\gamma_s}}$.
2. The nonlinear reconstruction bound changes as follows. Since $\alpha + r \leq \gamma_s$, the factor $\|u(t_l)\|_{X^{\alpha+r}}$ is at most $CR_{\gamma'}$ at every l, so

$$\|E_{\text{nl},q}\|_{X^\alpha} \leq C_R h^r \,\Delta t \sum_{l=0}^{q-1}(t_q - t_l)^{-\sigma} \leq C_R T^\theta h^r.$$

3. For the quadrature error, the weighted condition of the theorem holds with $\varrho = 0$. Since $\alpha + r \leq \gamma_s$ and P_N is a contraction on $X^{\alpha+r}$, so $\|P_N u(t)\|_{X^{\alpha+r}} \leq \|u(t)\|_{X^{\alpha+r}} \leq CR_{\gamma'}$ for all $t \in [0,T]$. Assumption 2.8 with $s = \alpha + r$ then gives us

$$\sup_{0\leq t\leq T}\|f(P_N u(t), \nabla P_N u(t))\|_{X^{\beta+r}} \leq K_R\left(1 + CR_{\gamma'}\right).$$

   The estimate for $\varrho = 0$ above applies, and $\max_q \|E_{\text{quad},q}\|_{X^\alpha} \leq C_R h^r$.

4. For the projection term: In (36) the weighted bound is replaced by $\|P_N u(t_l) - u(t_l)\|_{X^\alpha} \leq \lambda_{N+1}^{-\zeta_s} R_{\gamma'}$ for every l, so the first sum is at most $C_R\lambda_{N+1}^{-\zeta_s}\,\Delta t\sum_{l=0}^{q-1}(t_q - t_l)^{-\sigma} \leq C_R T^\theta \lambda_{N+1}^{-\zeta_s}$. For the tail bound, we apply Proposition 2.12 with $\beta + 2\zeta_s = \gamma_s - (\alpha - \beta)$ which gives us

$$\|S(\tau)(I - P_N)F\|_{X^\alpha} \leq C\tau^{-\sigma}\|(I - P_N)F\|_{X^\beta} \leq C\tau^{-\sigma}\lambda_{N+1}^{-\zeta_s}\|F\|_{X^{\gamma_s-(\alpha-\beta)}},$$

   and summing this part over l gives $C_R T^\theta \lambda_{N+1}^{-\zeta_s}$.

5. Lastly, we update the temporal error term. We apply the second part of Proposition 4.1 with $g = F$, $\beta' := \gamma_s - (\alpha - \beta)$, and $\kappa := \zeta_t$, so that $\frac{\beta'-\beta}{2} = \zeta_s$. Its properties hold:

- Lemma 2.9 gives $\sup_t \|F(t)\|_{X^{\beta'}} \le G := K_R(1 + R_{\gamma'})$.
- Lemma 2.10, whose exponent $\min\{1, \zeta_s\}$ is ζ_t, gives $\|F(t) - F(s)\|_{X^\beta} \le H\,(t-s)^{\zeta_t}$ with $H := L_R C_t$.
- The condition $\kappa < \theta + \frac{\beta' - \beta}{2}$ is $\zeta_t < \theta + \zeta_s$, which holds because $\zeta_t \le \zeta_s$ and $\theta > 0$.

Therefore, the error is $\max_q \|E_{\text{time},q}\|_{X^\alpha} \le C\left(T^\theta H + T^{1-a} G\right)(\Delta t)^{\zeta_t} \le C_R(\Delta t)^{\zeta_t}$.

In this case, $(\Delta t)^\theta$ does not appear, which gives the bound of the theorem with $\lambda_{N+1}^{-\zeta_s}$ and $(\Delta t)^{\zeta_t}$, depending on the extra regularity induced via bootstrapping. □

Proof of Lemma 5.1. For $t \in [t_q, t_{q+1})$, set the difference to be $\tau := t - t_q \in [0, \Delta t)$. Using the property that $S(\tau)\mathcal{S}_N = \mathcal{S}_N S_N(\tau)$, we split the error into

$$\left(\mathcal{L}^S_{N,Q} a\right)(t) - u(t) = S(\tau)\left(\mathcal{S}_N a^q - u(t_q)\right) + \left(S(\tau)u(t_q) - u(t_q + \tau)\right).$$

The first difference is mapped by $S(\tau)$ (contractive in X^α), and thus is a bound on the node-wise difference directly. For the second term, we can start the mild form at t_q to get

$$u(t_q + \tau) = S(\tau)u(t_q) + \int_0^\tau S(\tau - s) f\left(u(t_q + s), \nabla u(t_q + s)\right)\, ds,$$

and thus applying Assumption 2.3 yields

$$\left\|S(\tau)u(t_q) - u(t_q + \tau)\right\|_{X^\alpha} \le C\tau^{1-(\alpha-\beta)/2} \|f(u, \nabla u)\|_{X_T^\beta} \le C F_R (\Delta t)^\theta.$$

Taking the supremum over t and q completes the first part.

Using the bootstrap, the force is in $X^{\gamma_s - (\alpha - \beta)}$ and has norm at most $K_R(1 + R_{\gamma'})$. The smoothing from that space into X^α includes an additional $(t-\tau)^{-(\sigma - \zeta_s)_+}$ factor so the Duhamel term is at most $CK_R(1 + R_{\gamma'})(\Delta t)^{1-(\sigma-\zeta_s)_+}$, where $1 - (\sigma - \zeta_s)_+ = \min\{1, \theta + \zeta_s\} \ge \zeta_t$. □

Proof of Lemma 5.3. We look at the error induced by sampling the function and its coefficients, splitting it into two pieces: $\mathcal{S}_N b^q - u(t_q) = \mathcal{S}_N(\mathcal{C}_N^h \mathcal{I}_h - \mathcal{C}_N)u(t_q) + (P_N - I)u(t_q)$. Using the mild form, we have

$$u(t) = S(t)u_0 + \mathcal{D}[g](t), \qquad g(t) = f\left(u(t), \nabla u(t)\right). \tag{38}$$

For the spectral error (first term), $(I - P_N)$ commutes with $S(t)$, so we have

$$\|(I - P_N)S(t)u_0\|_{X^\alpha} = \|S(t)(I - P_N)u_0\|_{X^\alpha} \le \|(I - P_N)u_0\|_{X^\alpha} = \varepsilon_N^{\text{tail}}(u_0),$$

where for the inequality we used the contraction $S(t)$ on X^α. For the Duhamel term, by Assumption 2.3 and Lemma 2.7(i), we get

$$\|\mathcal{D}[g](t)\|_{X^\gamma} \le C\int_0^t (t-s)^{-\bar\sigma}\, ds\; F_R = \frac{C}{1-\bar\sigma}\, t^{1-\bar\sigma} F_R \le C\, T^{\theta - \zeta} F_R,$$

which is finite since $\bar\sigma < 1$, so Proposition 2.12 gives $\|(I - P_N)\mathcal{D}[g](t)\|_{X^\alpha} \le \lambda_{N+1}^{-\zeta} \|\mathcal{D}[g](t)\|_{X^\gamma} \le C_R \lambda_{N+1}^{-\zeta}$.

The next part is splitting the quadrature error (coefficient error) in the same way. One part gives $\|(\mathcal{C}_N^h \mathcal{I}_h - \mathcal{C}_N)S(t)u_0\|_{\ell^2_{\vec\alpha}} \le \varepsilon^{\text{enc}}_{N,h}(u_0)$ directly by Definition 5.2. On the Duhamel term we use Assumption 4.18 to get the bound $C_{\text{quad}} h^r \|\mathcal{D}[g](t)\|_{X^{\alpha+r}}$, and

$$\|\mathcal{D}[g](t)\|_{X^{\alpha+r}} \le C\|\mathcal{D}[g](t)\|_{X^\gamma} \le C_R$$

as before, where $r \le \gamma - \alpha$. If $\gamma_s = \gamma' > \gamma$, Lemma 2.9 improves the estimate to

$$\|\mathcal{D}[g](t)\|_{X^{\gamma_s}} \le \|u(t)\|_{X^{\gamma_s}} + \|S(t)u_0\|_{X^{\gamma_s}} \le R_{\gamma'} + \|u_0\|_{X^{\gamma'}},$$

instead. The same estimates yield $\lambda_{N+1}^{-\zeta_s}$ and allows for $r \le \gamma_s - \alpha$. Adding together the four parts and taking the maximum over q gives the result. The final part is directly from the triangle inequality and inclusion of the discrete ball Proposition 4.13. □

Proof of Corollary 5.7. Given $p > p_f$, pick a valid representative DAG G with $p_{\max}(G) < p$. Such graphs exist by the definition of the infimum. If we apply Theorem 5.6 to G, we get $W \le C_G \eta^{-p_{\max}(G)\chi_s} \log \eta^{-1} \le C_G \eta^{-p\chi_s} \log \eta^{-1}$ for $\eta \le 1$ and $\alpha \ne 2$. For $\alpha = 2$, where $\chi = 1$, the additional factor $(\log \eta^{-1})^{p_{\max}(G)}$ is at most $C\eta^{-(p-p_{\max}(G))}$ since $p > p_{\max}(G)$. In both cases $C_p := C_G$'s dependence on p is through the choice of G. So when the infimum is attained, we can pick the G that attains it. The last part is the case $p_{\max} = 0$ (Theorem 5.6), since then $\varepsilon^{-p_{\max}} = 1$. □

Proof of Proposition 5.9. First we look at one window. Let $a_\varepsilon^n(c)$ be the n-th neural iterate starting from c, so $\tilde{u}_{j+1} = \mathcal{S}_N[a_\varepsilon^n(b_j)]^Q$. The difference can be split into

$$\tilde{u}_{j+1} - u_{j+1} = \mathcal{S}_N \left(a_\varepsilon^n(b_j) - x(b_j)\right)^Q + \mathcal{S}_N \left(\Phi_{\text{win}}(b_j) - \Phi_{\text{win}}(\beta_j)\right) + \left(\mathcal{S}_N \Phi_{\text{win}}(\beta_j) - u_{j+1}\right).$$

For the first term, recall that $x(b_j)$ is the fixed point of $\Psi_{N,Q}^h$ started from b_j (admissible by (ii)), so (9) and (13) give

$$\left\| \mathcal{S}_N \left[a_\varepsilon^n(b_j) - x(b_j)\right]^Q \right\|_{X^\alpha} \le \left\| a_\varepsilon^n(b_j) - x(b_j) \right\|_{X^\alpha_{N,Q}} \le \frac{E_{\text{net}}}{1-\kappa} + 2R\,\kappa^n.$$

For the third term, $\mathcal{S}_N \Phi_{\text{win}}(\beta_j) - u_{j+1}$ is the nodal error at $q = Q$ of the fixed point $x(\beta_j)$ using the data u_j. Thus using (14), (16), and Lemma 5.3 yields

$$\left\| \mathcal{S}_N \Phi_{\text{win}}(\beta_j) - u_{j+1} \right\|_{X^\alpha} \le \frac{E_{\text{cons}} + E_{\text{samp}}}{1-\kappa} + E_{\text{samp}},$$

where E_{cons} and E_{samp} use the data u_j. Adding them together, they can be bounded by Theorem 5.4 (without E_{rec}), which is at most η_{win} by (i). Since $\|v\|_{X^s} \le \lambda_1^{-(\alpha-s)/2} \|v\|_{X^\alpha}$ for $s \le \alpha$, the terms are at most $\lambda_1^{-(\alpha-s)/2}\eta_{\text{win}}$ in X^s. We use (iii) to bound the middle term (in both norms). Writing the difference as $e_j := \|b_j - \beta_j\|_{\ell^2_{\tilde{s}}}$, we have

$$\|\tilde{u}_{j+1} - u_{j+1}\|_{X^\alpha} \le \eta_{\text{win}} + C_{\text{lift}} e_j, \qquad \|\tilde{u}_{j+1} - u_{j+1}\|_{X^s} \le \lambda_1^{-(\alpha-s)/2}\eta_{\text{win}} + \Theta_{\text{win}} e_j. \tag{39}$$

Next we look over the windows. Since $\tilde{u}_{j+1} \in E_N$,

$$e_{j+1} \le \|P_N(\tilde{u}_{j+1} - u_{j+1})\|_{X^s} + \|(\mathcal{C}_N - \mathcal{C}_N^h \mathcal{I}_h) u_{j+1}\|_{\ell^2_{\tilde{s}}} \le \|\tilde{u}_{j+1} - u_{j+1}\|_{X^s} + \lambda_1^{-(\alpha-s)/2} \varepsilon_{N,h}^{\text{enc}}(u_{j+1}).$$

Then using (39), we get $e_{j+1} \le \Theta_{\text{win}} e_j + \lambda_1^{-(\alpha-s)/2}\bar{\eta}$. Since $e_0 = 0$, we get $e_j \le \lambda_1^{-(\alpha-s)/2}\bar{\eta} \sum_{i<j} \Theta_{\text{win}}^i$, and the first inequality of (39) produces the result.

Note that the steps above used (ii) and (iii) with b_j, which requires $\|b_j - \beta_j\|_{\ell^2_{\tilde{\alpha}}} \le \delta$. By the same splitting, we have

$$\|b_j - \beta_j\|_{\ell^2_{\tilde{\alpha}}} \le \|\tilde{u}_j - u_j\|_{X^\alpha} + \varepsilon_{N,h}^{\text{enc}}(u_j) \le \bar{\eta} + C_{\text{lift}} e_{j-1}.$$

The condition that this is less than δ in the theorem statement allows for the induction to be valid. □

The condition (iii) holds when we set $s = \alpha$ and $\Theta_{\text{win}} = C_{\text{lift}} = 1/(1-\kappa)$. Under an energy estimate, it holds with $s = 0$, see Lemmas A.1 and A.2.

A.4 Realization as a ReLU Network

Proof of Theorem 5.13. The proof combines the prior results and is a bit of bookkeeping. By (23) and Proposition 4.13, $A^0(w) \in B_{R_0}^{N,Q}$ is an admissible starting point for Theorem 5.4. Let $A^k(w) = a_\varepsilon^k$ denote the k-th neural Picard sweep. The accuracy bound (i) from this theorem is the result of Theorem 5.6.

We next count the weights of a single sweep of $\Psi_{N,Q,\varepsilon}^h$. This has three parts. For the synthesis, at each of the Q time points the pair $\left(\mathcal{S}_N^h a^r, \nabla_h \mathcal{S}_N^h a^r\right)$ is computed by applying $d_x + 1$ transformation to a coefficient vector a^r and are from Assumption 5.11. In the Fourier case they are repeated transforms, adjusted by multipliers $\mathrm{i}k_j$, and thus a single factorization is enough. For a general eigenbasis $\partial_j e_k$ may not be directly

proportional to e_k and Assumption 5.11 applied to each of the $d_x + 1$ operations. By Assumption 5.11, this process costs at most $(1 + d_x)d\,(\mathcal{T}(M) + N)$ weights and $C \log M$ layers per time stamp, and so it is at most $(1 + d_x)dQ\,(\mathcal{T}(M) + N)$ weights for the synthesis.

For the nonlinear part, the neural network $\widehat{f}_\varepsilon$ is applied in parallel at the MQ space-time points, which contributes $MQ\,W(\widehat{f}_\varepsilon)$ weights and depth $\mathrm{dep}(\widehat{f}_\varepsilon)$. Since the copies are identical, the number of distinct weights is just $W(\widehat{f}_\varepsilon)$.

For the analysis and the temporal parts, there are Q applications of $\mathcal{C}_N^h$ which costs at most $dQ\mathcal{T}(M)$ weights and $C \log M$ layers. We need to also compute

$$v^q := S_N(t_q)\, a_0^h + \Delta t \sum_{r=0}^{q-1} S_N(t_q - t_r)\, \mathcal{C}_N^h g^r,$$

for $q = 0, \dots, Q$. These are exactly the iterates from (22), with diagonal $S_N(\Delta t)$ and so it decouples into dN independent scalar iterations $v_k^{q+1} = \rho_k v_k^q + c_k^q$ with $\rho_k := e^{-\lambda_k \Delta t}$. The process is associative, i.e., the maps $x \mapsto \rho x + c$ can be composed by $(\rho_1, c_1) \circ (\rho_2, c_2) = (\rho_1\rho_2, \rho_1 c_2 + c_1)$, and since the multipliers ρ are constants, each composition is an affine map of the data components (c_1, c_2) with two nonzero weights. A parallel scan can produce $v^0, \dots, v^Q$ from $c^0, \dots, c^{Q-1}$ and a_0^h using CQ nodes in $C \log(2Q)$ layers. In a deep network, the quantities that are not updated at a given layer are carried forward using the identity process from Lemma 3.3(iv), which is at most $Q + 1$ values per layer with four weights each. Therefore, one scalar equation costs at most $CQ \log(2Q)$ weights. Summing over all the dN recurrence equations, the total cost is (at most) $CdNQ \log(2Q)$ weights and $C \log(2Q)$ layers.

Putting this together, one neural Picard sweep has depth at most $\mathrm{dep}(\widehat{f}_\varepsilon) + C \log(MQ)$ and size at most

$$C\left[(2 + d_x)dQ\mathcal{T}(M) + dNQ \log(2Q) + MQ\, W(\widehat{f}_\varepsilon)\right],$$

where $(1 + d_x)dQN$ from the derivative synthesis have been absorbed into $dNQ \log(2Q)$. Each sweep also maps a_0^h (dimension dN) through the layers using the identity (Lemma 3.3(iv)), since it starts from $v^0 = a_0^h$. This costs $CdN\left(\mathrm{dep}(\widehat{f}_\varepsilon) + \log(MQ)\right) \le CdN\left(W(\widehat{f}_\varepsilon) + \log(MQ)\right)$ weights, using $\mathrm{dep}(\widehat{f}_\varepsilon) \le W(\widehat{f}_\varepsilon)$, and the first part is at most $MQ\, W(\widehat{f}_\varepsilon)$ if $N \le MQ$. Composing n sweeps and including the encoder step $w \mapsto \mathcal{C}_N^h w$, gives (ii) and the first bound in (iii). If $N \le CM$, then $dN\left(W(\widehat{f}_\varepsilon) + \log(MQ)\right) \le C\left(MQ\, W(\widehat{f}_\varepsilon) + dMQ \log(MQ)\right)$, so the identity cost can be absorbed into (24).

For the second bound in (iii), we assumed that $N \le CM$ and $\mathcal{T}(M) \le CM \log M$. Thus (up to constants) $dNQ \log(2Q) \le CdMQ \log(MQ)$, $dN \log(MQ) \le dMQ \log(MQ)$, $(2 + d_x)dQ\,\mathcal{T}(M) \le C(2 + d_x)dMQ \log M$, and the encoder costs $\mathcal{T}(M) \le CM \log M$. Combining that together gives (24). Part (iv) follows from Lemma 3.4 and Definition 4.6, where the block is one neural network applied at nMQ points, and the spectral layers are the same across time-stamps and sweeps. □

Proof of Lemma 5.17. For any $g \in \widetilde{X}^\beta$ and $k \le N$, using (25) gives

$$(\mathcal{C}_N \mathcal{M} g)_k = \langle \mathcal{M} g, e_k \rangle_H = \langle g, \mathcal{M}^* e_k \rangle = \mu_k \langle g, e_k \rangle = (D_\mu \mathcal{C}_N g)_k,$$

which is result (i). The matrix is diagonal, and diagonal matrices commute with $S_N(t) = e^{-t\Lambda_N}$. Also, the norm is

$$\|D_\mu c\|_{\ell^2_s}^2 = \textstyle\sum_k \lambda_k^s |\mu_k|^2 |c_k|^2 \le C_{\mathcal{M}}^2 \|c\|_{\ell^2_s}^2.$$

For (ii), we look at where $\mathcal{C}_N^h$ is used. In Theorem 4.20, it appears only through the discrete smoothing $\|S_N(\tau)\mathcal{C}_N^h v\|_{\ell^2_\alpha} \le CC_{\mathrm{stab}} \tau^{-(\alpha-\beta)/2} \|v\|_{\widetilde{X}^\beta}$, and so putting D_μ into it and using (i) would just multiply the RHS by at most $C_{\mathcal{M}}$. The term $v = f_{\mathrm{loc}}(u, \nabla u)$ would now be estimated in $\widetilde{X}^\beta$, giving a L_R^{loc} and F_R^{loc} factor instead of L_R and F_R. The same applies in Theorems 4.21 and 4.22, where D_μ commutes with $S_N(t_q - t_r)$ and can be absorbed into the constants. Theorems 5.4 and 5.6 use only the conclusions of these results.

Part (iii) is direct: $D_\mu = \mathrm{diag}(\mu_1, \dots, \mu_N)$ is determined only by $\mathcal{M}$ and the eigenfunctions $e_1, \dots, e_N$ of L through (25). If $\mathcal{M}$ is the orthogonal projection of $\widetilde{H}$ onto H, then $\mathcal{M}^* e_k = e_k$ since $e_k \in H$, so we have $\mu_k = 1$ and $D_\mu = I$. □

A.5 Solution Operator Proofs

Proof of Lemma 6.1. In the proof of Proposition 2.5, we can replace the radius by the value $R = 1 + 2R_0$. The invariance part holds since $\|S(\cdot)u_0\|_{X_T^\alpha} \le R_0 < R/2$ for every u_0 with $\|u_0\|_{X^\alpha} \le R_0$. The value of T_0 is set by $CT_0^\theta L_R \le \frac{1}{2}$, which only sees u_0 through R, which is uniform. □

Proof of Lemma 6.2. By Proposition 2.12, $\|I - P_N\|_{\mathcal{L}(X^\gamma, X^\alpha)} \le \lambda_{N+1}^{-(\gamma-\alpha)/2} \to 0$ as $N \to \infty$. This map is the limit of finite-rank operators P_N (measured in the operator norm) and thus is compact, so the set $\mathcal{U}_{R_0}$ is totally bounded in X^α. It is also closed in X^α, since if we take $u_m \to u$ in X^α with $\|u_m\|_{X^\gamma} \le R_0$, then $\langle u_m, e_k\rangle \to \langle u, e_k\rangle$ for every k. By Fatou's lemma, the series is bounded by

$$\|u\|_{X^\gamma}^2 = \sup_K \sum_{k\le K} \lambda_k^\gamma |\langle u, e_k\rangle|^2 = \sup_K \liminf_{m\to\infty} \sum_{k\le K} \lambda_k^\gamma |\langle u_m, e_k\rangle|^2 \le \liminf_{m\to\infty} \|u_m\|_{X^\gamma}^2 \le R_0^2.$$

Thus by completeness, we have shown compactness. □

Proof of Lemma 6.4. Set $E_h := \mathcal{C}_N^h \mathcal{I}_h - \mathcal{C}_N$, so by Lemma 4.17 and the bound $\|\mathcal{C}_N v\|_{\ell^2_\alpha} \le \|v\|_{X^\alpha}$, we have the uniform in h estimate $\|E_h v\|_{\ell^2_\alpha} \le C_E \|v\|_{X^\alpha}$ (with $C_E := C_{\text{stab}}^\alpha + 1$). The set $\bigcup_{K\ge1} E_K$ (finite eigen-expansion) is dense in X^α. For $w \in \bigcup_{K\ge1} E_K$, Assumption 4.18 gives

$$\|E_h w\|_{\ell^2_\alpha} \le C_{\text{quad}} h^r \|w\|_{X^{\alpha+r}} \to 0,$$

as $h \to 0$. Given $\epsilon > 0$, choose a finite $\frac{\epsilon}{3C_E}$-net $\{v_1, \dots, v_J\}$ of $\mathcal{V}$ (exists by compactness) and a finite set of elements $w_j \in \bigcup_{K\ge1} E_K$ where $\|v_j - w_j\|_{X^\alpha} \le \frac{\epsilon}{3C_E}$. For $v \in \mathcal{V}$ and the nearest v_j, we have

$$\|E_h v\|_{\ell^2_\alpha} \le C_E \|v - w_j\|_{X^\alpha} + \|E_h w_j\|_{\ell^2_\alpha} \le \frac{2\epsilon}{3} + \max_{1\le i\le J} \|E_h w_i\|_{\ell^2_\alpha}.$$

The max is over finitely many terms and is at most $\frac{\epsilon}{3}$ for h small enough. Therefore, we have $\sup_{v\in\mathcal{V}} \|E_h v\|_{\ell^2_\alpha} \le \epsilon$ for h sufficiently small. □

Proof of Theorem 6.5. We just need to show that every constant used in Theorems 5.4, 5.6, and 5.13 depends only on the data via R_0 and ρ_γ. Thus, the parameter can be controlled uniformly and the neural network can be applied to every $u_0 \in \mathcal{U}$.

Since $\|u_0\|_{X^\alpha} \le R_0$, Lemma 6.1 gives $R = 1 + 2R_0$, T_0, and a contraction constant κ_0 which is the same to all $u_0 \in \mathcal{U}$, and that $G_T[u_0] \in B_R^\alpha$. Because of this, we have several implications. The compact set K from Assumption 3.5 depends only on R, so one neural network $\widehat{f_\varepsilon}$ is valid for every input data, and using Corollary 4.14 it is evaluated only on set K. The constants L_R and F_R, and thus $\kappa = CC_{\text{stab}} L_R T^\theta$ in Theorem 4.20, are the same for the data, and by Assumption 6.3 with ρ_γ, are the values R_γ and C_R from Lemmas 5.1 and 5.3 and from Theorem 4.22. The initialization terms are uniformly controlled. First, Assumption 4.18 gives

$$E_{\text{init}}^h \le C_{\text{quad}} h^r \|u_0\|_{X^{\alpha+r}} \le C\rho_\gamma h^r,$$

since $\alpha + r \le \gamma$ by (7). Second by Proposition 2.12, we have $\|(I - P_N)u_0\|_{X^\alpha} \le \lambda_{N+1}^{-(\gamma-\alpha)/2}\rho_\gamma$.

Lastly, the admissibility condition (23) holds uniformly over the input set by Lemma 4.17, namely, $\|\mathcal{C}_N^h \mathcal{I}_h u_0\|_{\ell^2_\alpha} \le C_{\text{stab}}^\alpha R_0$ for every $u_0 \in \mathcal{U}$ and admissible (N, h). We need to increase the radius by replacing R_0 by $C_{\text{stab}}^\alpha R_0$ and R by $1 + 2C_{\text{stab}}^\alpha R_0$ in Lemma 6.1. But fortunately, this can be done once and can be fixed before any parameter is set. This will change constants but will not affect the exponent. Therefore the bound of Theorem 5.4 holds with one constant C_R for all $u_0 \in \mathcal{U}$, and the balancing of terms in Theorem 5.6 picks $(N, h, Q, n, \varepsilon)$ once and uniformly. Theorem 5.13 then applies directly and completes the proof. □

Proof of Proposition 7.2. Result (i) is directly implied by $\|\mathcal{M}g\|_{X^\beta}^2 = \sum_k \lambda_k^{\alpha-1} |\mu_k|^2 |\langle g, \tilde{e}_k\rangle|^2 \le C_{\mathcal{M}}^2 \|g\|_{\widetilde{X}^\alpha}^2$. For part (ii), note that D_μ and $S_N(\tau)$ are both diagonal and so they commute, which yields

$$\left\| S_N(\tau) D_\mu \widetilde{\mathcal{C}}_N^h v \right\|_{\ell^2_\alpha} = \left\| D_\mu S_N(\tau) \widetilde{\mathcal{C}}_N^h v \right\|_{\ell^2_\alpha} \le C_{\mathcal{M}} \left\| S_N(\tau) \widetilde{\mathcal{C}}_N^h v \right\|_{\ell^2_{\alpha+1}},$$

where we used that $\|D_\mu c\|_{\ell^2_\alpha}^2 = \sum_k \lambda_k^\alpha |\mu_k|^2 |c_k|^2 \le C_{\mathcal{M}}^2 \sum_k \lambda_k^{\alpha+1} |c_k|^2$ and the condition that $|\mu_k| \le C_{\mathcal{M}} \lambda_k^{1/2}$. Next, by duality in the proof of Theorem 4.20 applied to the ambient space and over $\widetilde{E}_N := \operatorname{span}\{\tilde{e}_1, \dots, \tilde{e}_N\}$, we have

$$\left\| S_N(\tau) \widetilde{\mathcal{C}}_N^h v \right\|_{\ell^2_{\alpha+1}} = \sup_{\substack{\phi \in \widetilde{E}_N \\ \|\phi\|_{\widetilde{X}^{-\alpha-1}} \le 1}} \left| \sum_i \omega_i^{N,h} v(x_i) (S(\tau)\phi)(x_i) \right| \le C_{\mathrm{stab}}^\alpha \|v\|_{\widetilde{X}^\alpha} \sup_\phi \|S(\tau)\phi\|_{\widetilde{X}^{-\alpha}},$$

where $S(\tau)$ maps into itself since $\tilde{e}_k$ are eigenfunctions and $\|S(\tau)\phi\|_{X^{-\alpha}} \le C\tau^{-1/2} \|\phi\|_{X^{-\alpha-1}}$ by the negative-order spaces from Definition 4.15. Since the exponent $\frac{1}{2}$ agrees with $\sigma = \frac{\alpha-\beta}{2}$ for $\beta = \alpha - 1$, the other estimate hold.

For Part (iv), the network error g_ε is only controlled in ℓ^∞, so $\|\widetilde{\mathcal{C}}_N^h g_\varepsilon\|_{\ell^2_0} \le C\varepsilon$ from the proof of Theorem 4.21, and

$$\left\| S_N(\tau) D_\mu \widetilde{\mathcal{C}}_N^h g_\varepsilon \right\|_{\ell^2_\alpha} \le C_{\mathcal{M}} \left\| S_N(\tau) \widetilde{\mathcal{C}}_N^h g_\varepsilon \right\|_{\ell^2_{\alpha+1}} \le C\varepsilon \min\left\{ \lambda_N^{(\alpha+1)/2}, \tau^{-(\alpha+1)/2} \right\}.$$

The rest of that proof applies, where we put $\alpha + 1$ in place of α, and thus have the factor $\xi_{\alpha+1}$.

Lastly, part (iii) holds since the network now only sees the values $\mathcal{S}_N^h a^r$, which are well-defined pointwise using continuity alone. □

A.6 Window Stability for General Governing Laws

This subsection gives the condition (iii) of Proposition 5.9 for general laws, where Lemma 5.10 is the Allen-Cahn case.

Lemma A.1 (Discrete Energy Stability)**.** *Let $B \subset \mathbb{R}^N$, and suppose there are $\xi \in \mathbb{R}$ and $\Gamma_0 \ge 0$ such that, for $a', b' \in B$,*

$$\langle G_N(a') - G_N(b'), a' - b' \rangle_0 \le \xi \, \|a' - b'\|_{\ell^2_0}^2, \qquad \|G_N(a') - G_N(b')\|_{\ell^2_0} \le \Gamma_0 \|a' - b'\|_{\ell^2_0}.$$

If $\Delta t \, \Gamma_0^2 \le 1$ and $x^q(a), x^q(b) \in B$ for $0 \le q < Q$, then

$$\|z^q\|_{\ell^2_0} \le e^{\psi t_q} \|z^0\|_{\ell^2_0}$$

for $0 \le q \le Q$ where $\psi := \xi + \frac{1}{2} - \lambda_1$.

Proof of Lemma A.1. By (19) and (20), $\|z^{q+1}\|_{\ell^2_0} \le e^{-\lambda_1 \Delta t} \|z^q + \Delta t \, d^q\|_{\ell^2_0}$. Using the assumption and expanding the expression gives

$$\|z^q + \Delta t \, d^q\|_{\ell^2_0}^2 = \|z^q\|_{\ell^2_0}^2 + 2\Delta t \langle d^q, z^q \rangle_0 + \Delta t^2 \|d^q\|_{\ell^2_0}^2 \le (1 + (2\xi + 1)\Delta t) \, \|z^q\|_{\ell^2_0}^2 \le e^{(2\xi+1)\Delta t} \|z^q\|_{\ell^2_0}^2,$$

where we used $\Delta t^2 \Gamma_0^2 \le \Delta t$ and $1 + x \le e^x$. Thus $\|z^{q+1}\|_{\ell^2_0} \le e^{\psi \Delta t} \|z^q\|_{\ell^2_0}$, and the claim follows using induction. □

The one-sided constant ξ can be much smaller than a Lipschitz constant. If $f = f(u)$ satisfies $\langle f(y) - f(y'), y - y' \rangle \le \xi_f |y - y'|^2$ and has Lipschitz constant $\operatorname{Lip}(f)$ on a convex neighborhood of K, then (21), (6), and Assumption 4.16(i) gives the conditions $\xi \le C_{\mathrm{stab}}^0 \max\{\xi_f, 0\}$ and $\Gamma_0 \le C_{\mathrm{stab}}^0 \operatorname{Lip}(f)$, independently of N and h. For laws involving ∇u, Γ_0 grows like $\lambda_N^{1/2}$, and an N-independent ξ would require some control of the quadrature aliasing. While this is possible, we do not pursue it here.

Lemma A.2 (X^α Over Windows)**.** *Let $T \le 1$ and B be from Lemma A.1. If $x^q(a), x^q(b) \in B$ for $0 \le q < Q$ and the following hold*

(i) $\|d^0\|_{\ell^2_0} \le \Gamma_0 \|z^0\|_{\ell^2_0}$ with $\Delta t \, \Gamma_0 \le 1$;

(ii) $\|G_N(a') - G_N(b')\|_{\ell^2_\beta} \le \Gamma_\alpha \|a' - b'\|_{\ell^2_\alpha}$ *for* $a', b' \in B$, *and* $2^{\alpha/2} c_\alpha \Gamma_\alpha T^\theta/\theta \le \frac{1}{2}$;

(iii) $\|d^p\|_{\ell^2_0} \le \Gamma_* \, t_p^{-1/2} \|z^0\|_{\ell^2_0}$ *for* $1 \le p < Q$,

then

$$\|\Phi_{\text{win}}(a) - \Phi_{\text{win}}(b)\|_{\ell^2_\alpha} \le C_{\text{lift}} \, \|a - b\|_{\ell^2_0},$$

where $C_{\text{lift}} := 4c_\alpha T^{-\alpha/2} \left(1 + 2^{(\alpha-1)/2} \Gamma_* T^{1/2}\right)$.

Condition (i) is weaker than the condition of Lemma A.1 and it holds under the choices of Theorem 5.6 when $u_0 \in X^{\gamma_s}$. This holds even for laws involving the gradient, there $\Gamma_0 \le C(1 + \lambda_N^{1/2})$, while $\Delta t \, \lambda_N \le C(c_{\text{err}}\eta)^{1/\zeta_t - 1/\zeta_s} \le C$ since $\zeta_t \le \zeta_s$, thus $\Delta t \, \Gamma_0 \le C\Delta t^{1/2} \le 1$ for sufficiently small η. Condition (iii) of Lemma A.2 follows from either of two bounds on G_N. Using the energy estimate (Lemma A.1) it holds with $\Gamma_* = \Gamma_0 e^{\psi_+ T} T^{1/2}$ and $\psi_+ := \max\{\psi, 0\}$, since $\|d^p\|_{\ell^2_0} \le \Gamma_0 \|z^p\|_{\ell^2_0} \le \Gamma_0 e^{\psi_+ T} \|z^0\|_{\ell^2_0}$ and $t_p \le T$. Using a Lipschitz bound $\|G_N(a') - G_N(b')\|_{\ell^2_0} \le \Gamma_1 \|a' - b'\|_{\ell^2_1}$ on B, it holds with $\Gamma_* = 4c_\alpha \Gamma_1 e^{64 c_\alpha^2 \Gamma_1^2 T}$, as shown after the proof.

Proof of Lemma A.2. We can write out the time-stepper (19) which gives

$$z^q = S_N(t_q) z^0 + \Delta t \sum_{p<q} S_N(t_q - t_p) d^p.$$

Let $W := \max_{1 \le q \le Q} t_q^{\alpha/2} \|z^q\|_{\ell^2_\alpha}$ and fix $1 \le q \le Q$. We split the sum into the term $p = 0$, the early terms with $1 \le p$ and $t_p \le t_q/2$, and the remaining terms $t_q/2 < t_p < t_q$.

First $p = 0$ and the z^0 part. By (20) and (i), they give

$$c_\alpha t_q^{-\alpha/2} (1 + \Delta t \Gamma_0) \|z^0\|_{\ell^2_0} \le 2 c_\alpha t_q^{-\alpha/2} \|z^0\|_{\ell^2_0},$$

since $\Delta t \, \Gamma_0 \le 1$ by (i). For the early terms, we have $t_q - t_p \ge t_q/2$, so using (20), (iii), and $\Delta t \sum_{t_p \le t_q/2} t_p^{-1/2} \le 2(t_q/2)^{1/2}$, they are bounded by $2c_\alpha \Gamma_* (t_q/2)^{(1-\alpha)/2} \|z^0\|_{\ell^2_0}$. The remaining terms satisfy $\|z^p\|_{\ell^2_\alpha} \le t_p^{-\alpha/2} W \le (t_q/2)^{-\alpha/2} W$. Thus using (ii) and (20) with $r = \alpha - \beta$, they give at most

$$c_\alpha \Gamma_\alpha \left(\frac{t_q}{2}\right)^{-\alpha/2} W \, \Delta t \sum_{t_q/2 < t_p < t_q} (t_q - t_p)^{-\sigma} \le c_\alpha \Gamma_\alpha \left(\frac{t_q}{2}\right)^{-\alpha/2} \frac{T^\theta}{\theta} W.$$

By definition of W, we take the maximum after multiplying by $t_q^{\alpha/2}$, using $t_q^{1/2} \le T^{1/2}$ and condition (ii) to get

$$W \le 2c_\alpha \|z^0\|_{\ell^2_0} + 2^{(\alpha+1)/2} c_\alpha \Gamma_* T^{1/2} \|z^0\|_{\ell^2_0} + \tfrac{1}{2} W,$$

so $W \le C_{\text{lift}} T^{\alpha/2} \|z^0\|_{\ell^2_0}$, and $\|\Phi_{\text{win}}(a) - \Phi_{\text{win}}(b)\|_{\ell^2_\alpha} = \|z^Q\|_{\ell^2_\alpha} \le T^{-\alpha/2} W$. □

Proof of the second case of (iii). We can also show this just with Lipschitz. We need $\|z^p\|_{\ell^2_1} \lesssim t_p^{-1/2} \|z^0\|_{\ell^2_0}$, since then $\|d^p\|_{\ell^2_0} \le \Gamma_1 \|z^p\|_{\ell^2_1}$ would yield (iii).

By (20) with $r = 1$, valid since $\alpha \ge 1$, the z^0 and the $p = 0$ term are at most $2c_\alpha t_q^{-1/2} \|z^0\|_{\ell^2_0}$, exactly as above. The remaining terms give

$$\|z^q\|_{\ell^2_1} \le 2c_\alpha t_q^{-1/2} \|z^0\|_{\ell^2_0} + c_\alpha \Gamma_1 \, \Delta t \sum_{p=1}^{q-1} (t_q - t_p)^{-1/2} \|z^p\|_{\ell^2_1}, \tag{40}$$

for $1 \le q \le Q$. We bound $V := \max_{1\le q\le Q} t_q^{1/2} e^{-c_\Gamma^2 t_q} \|z^q\|_{\ell_1^2}$ with $c_\Gamma := 8c_\alpha\Gamma_1$ to close the system. The factor $t_q^{1/2}$ removes the singularity at $t = 0$, and $e^{-c_\Gamma^2 t_q}$ makes the sum in (40) small without shrinking the window. Multiplying (40) by $t_q^{1/2} e^{-c_\Gamma^2 t_q}$ and using $\|z^p\|_{\ell_1^2} \le t_p^{-1/2} e^{c_\Gamma^2 t_p} V$, we have the inequality

$$t_q^{1/2} e^{-c_\Gamma^2 t_q} \|z^q\|_{\ell_1^2} \le 2c_\alpha \|z^0\|_{\ell_0^2} + c_\alpha\Gamma_1 V\, \Delta t \sum_{p=1}^{q-1} \frac{t_q^{1/2}}{t_p^{1/2}(t_q - t_p)^{1/2}}\, e^{-c_\Gamma^2 (t_q - t_p)}.$$

We consider two cases. For $t_p \le t_q/2$, we have $t_q - t_p \ge t_q/2$ and $\Delta t \sum_{t_p \le t_q/2} t_p^{-1/2} \le 2(t_q/2)^{1/2}$, so in this case we have $2t_q^{1/2} e^{-c_\Gamma^2 t_q/2} \le 2e^{-1/2} c_\Gamma^{-1}$. For the other case where $t_p > t_q/2$, we have $t_q^{1/2} t_p^{-1/2} \le \sqrt{2}$, so this part gives $\sqrt{2}\int_0^\infty s^{-1/2} e^{-c_\Gamma^2 s}\, ds = \sqrt{2\pi}\, c_\Gamma^{-1}$. Since $2e^{-1/2} + \sqrt{2\pi} < 4$, the total sum is at most $4c_\Gamma^{-1}$, and taking the max over q yields

$$V \le 2c_\alpha \|z^0\|_{\ell_0^2} + \frac{4c_\alpha\Gamma_1}{c_\Gamma} V = 2c_\alpha \|z^0\|_{\ell_0^2} + \frac{1}{2} V.$$

Therefore $V \le 4c_\alpha \|z^0\|_{\ell_0^2}$, i.e., $\|z^p\|_{\ell_1^2} \le 4c_\alpha e^{c_\Gamma^2 T} t_p^{-1/2} \|z^0\|_{\ell_0^2}$, which shows (iii) with $\Gamma_* = 4c_\alpha\Gamma_1 e^{64 c_\alpha^2 \Gamma_1^2 T}$.

The second case applies to every $f(u, \nabla u)$ considered here except those in divergence form, since it only needs the Lipschitz bound. In divergence form, D_μ is unbounded and so Γ_1 scales like $\lambda_N^{1/2}$. Thus condition (iii) of Proposition 5.9 is taken from the contraction with $s = \alpha$. With $v = \mathcal{S}_N a'$ and $w = \mathcal{S}_N b'$, (21), (6) and Assumption 4.16(i), applied to $v - w$ and $\nabla(v - w)$, give $\Gamma_1 \le C^0_{\mathrm{stab}} \operatorname{Lip}(f) C_1$, where $\|y\|_H + \|\nabla y\|_H \le C_1 \|y\|_{X^1}$. Condition (ii) also always holds, with $\Gamma_\alpha = C^\beta_{\mathrm{stab}} L_R$, by the quadrature stability Assumption 4.16(ii) with $s = \beta$, but possibly with a shorter window.

B Quantitative ReLU Approximation Results

We recall two approximation results used for the DAG approximation in Section 3, that is, the classical rate for smooth univariate functions, and the approximation of multilinear maps.

Theorem B.1 (Quantitative Approximation of Smooth Scalar Functions)**.** *If $s \ge 1$ is an integer, $[a, b] \subset \mathbb{R}$ is a compact interval, and $g \in C^s([a, b])$, then there exist constants $C > 0$ and $\delta_0 \in (0, 1)$, depending only on g, s, and $[a, b]$, such that for every $\delta \in (0, \delta_0]$ there is a ReLU network $N_\delta : \mathbb{R} \to \mathbb{R}$ with*

$$\|g - N_\delta\|_{L^\infty([a,b])} \le \delta, \qquad \operatorname{dep}(N_\delta) \le C \log(\delta^{-1}), \qquad W(N_\delta) \le C\, \delta^{-1/s} \log(\delta^{-1}).$$

This is a one-dimensional version of [86, Theorem 1], where one uses $g / \max\{1, \|g\|_{W^{s,\infty}}\}$ and notes that $C^s([a, b]) \subset W^{s,\infty}((a, b))$. The restriction to integer s is from the referenced theorem and is used in Assumption 3.5.

Remark B.2 (Continuous Weights via Local Approximation)**.** *Let $M_\delta := \lceil \delta^{-1/s} \rceil$, and let y_i be nodes with spacing $h_\delta := (b - a)/M_\delta$ covering $[a, b]$. The hat functions $\phi_i(y) := h_\delta^{-1} (\sigma(y - y_{i-1}) - 2\sigma(y - y_i) + \sigma(y - y_{i+1}))$ are realized exactly by ReLU networks and form a partition of unity on $[a, b]$. Since ϕ_i is supported where $|y - y_i| \le h_\delta$, the combination of local Taylor polynomials*

$$\sum_i \phi_i(y) \sum_{l<s} \frac{g^{(l)}(y_i)}{l!} (y - y_i)^l$$

is within $\frac{\|g^{(s)}\|_\infty}{s!} h_\delta^s \le C\delta$ of g.

Each of the $O(\delta^{-1/s})$ local terms $\phi_i(y)(y - y_i)^l$ is a product of at most s factors bounded on $[a, b]$, so the multiplication networks of [86], as in Theorem B.3, approximate it with size and depth $O(\log \delta^{-1})$. At each y only two hat functions are active, so accuracy $\delta / (2s \max\{1, \max_{l<s} \|g^{(l)}\|_\infty\})$ per term is sufficient. The resulting network has the size bound of Theorem B.1. The hidden layers do not depend on g, and g appears only through the output

weights $g^{(l)}(y_i)/l!$, which depend continuously on g in C^s. The continuity is needed for comparisons in the lower bound results of Remark 5.8, Remark 7.4, and Theorem 7.5. The weight continuity of this construction was noted in [86].

Theorem B.3 (Quantitative Approximation of Multilinear Maps)**.** *Let $r \geq 2$, $n, p \geq 1$, and let $T : (\mathbb{R}^n)^r \to \mathbb{R}^p$ be multilinear. Let $b > 0$ and set $\mathcal{Q} := ([-b, b]^n)^r$. Then there exist constants $C > 0$ and $\delta_0 \in (0, 1)$, depending only on T, r, n, p, and b, such that for every $\delta \in (0, \delta_0]$ there is a ReLU network $\widetilde{T}_\delta : (\mathbb{R}^n)^r \to \mathbb{R}^p$ with*

$$\sup_{y \in \mathcal{Q}} \left\| T(y) - \widetilde{T}_\delta(y) \right\|_{\mathbb{R}^p} \leq \delta, \qquad \operatorname{dep}(\widetilde{T}_\delta) \leq C \log(\delta^{-1}), \qquad W(\widetilde{T}_\delta) \leq C \log(\delta^{-1}).$$

Proof of Theorem B.3. Denote $y = (y^{(1)}, \ldots, y^{(r)})$ with $y^{(l)} \in \mathbb{R}^n$ and thus the multilinear function can be written as

$$T(y) = \sum_{k_1, \ldots, k_r} c_{k_1 \ldots k_r} \, y^{(1)}_{k_1} \cdots y^{(r)}_{k_r}, \qquad c_{k_1 \ldots k_r} := T(e_{k_1}, \ldots, e_{k_r}) \in \mathbb{R}^p,$$

i.e., a fixed linear combination of n^r products of r coordinates, each bounded by b. The coefficients depend only on T, r and n. It is therefore enough to approximate $P(x_1, \ldots, x_r) := x_1 \cdots x_r$ on $[-B, B]^r$, $B := \max\{b, 1\}$, to collect all n^r copies in parallel and apply an exact affine output layer to multiply the terms by the proper coefficients.

From [86], for every $\mu \in (0, 1)$ and $B' \geq 1$ there is a ReLU network $\widetilde{\times}_\mu$ with

$$\sup_{|x|, |y| \leq B'} |xy - \widetilde{\times}_\mu(x, y)| \leq \mu$$

and $\operatorname{dep}(\widetilde{\times}_\mu), W(\widetilde{\times}_\mu) \leq C(B') \log(\mu^{-1})$. We next construct each of the r-fold products using a standard binary tree with $\lceil \log_2 r \rceil$ levels and by replacing each multiplication with the approximation $\widetilde{\times}_\mu$. Note that if r is not a power of two, then we will have one extra node that can be transferred to the next level in the tree using the identity $x = \sigma(x) - \sigma(-x)$ (which adds no error). Let P_l and $\widetilde{P}_l$ be the exact and computed values of a node at level l (which are products of at most 2^l of the inputs). Set the error at that level to be $e_l := \max |P_l - \widetilde{P}_l|$ and let $B_l := B^{2^l} + 1$.

The polynomial is bounded by $|P_l| \leq B^{2^l}$, since every term is at most $B \geq 1$ in magnitude. If two nodes at level l have errors at most e_l and satisfy $|\widetilde{P}_l|, |\widetilde{P}'_l| \leq B_l$, then

$$|P_l P'_l - \widetilde{P}_l \widetilde{P}'_l| \leq |P_l| \, |P'_l - \widetilde{P}'_l| + |\widetilde{P}'_l| \, |P_l - \widetilde{P}_l| \leq 2 B_l \, e_l.$$

The multiplication network is accurate up to error μ on the box of radius $B_{\max} := B_{\lceil \log_2 r \rceil}$, so it adds at most μ to the error, thus $e_{l+1} \leq 2 B_l e_l + \mu$. Using induction starting with $e_0 = 0$ gives,

$$e_l \leq C_1(B, r) \mu, \qquad C_1(B, r) := \sum_{l=0}^{\lceil \log_2 r \rceil - 1} \; \prod_{m=l+1}^{\lceil \log_2 r \rceil - 1} 2 B_m,$$

for every level $l \leq \lceil \log_2 r \rceil$ and since every term in C_1 is at least 1. The bound $|\widetilde{P}_l| \leq B_l$ is valid by induction on l since it holds for $l = 0$ and if $e_l \leq 1$ then $|\widetilde{P}_l| \leq |P_l| + e_l \leq B^{2^l} + 1 = B_l$. Setting $C_1 \mu \leq 1$, implies that every $e_l \leq 1$ and every step is evaluated inside each of the valid boxes. The network $\widetilde{P}_\mu$ has depth at most $\lceil \log_2 r \rceil \, C(B_{\max}) \log(\mu^{-1})$. Its size is at most $r - 1$ multiplication networks with identity steps of four weights per layer, which are at most $C(B, r) \log(\mu^{-1})$.

Lastly, we denote the coefficient magnitude by

$$c_{\text{sum}} := \sum_{k_1, \ldots, k_r} \| c_{k_1 \ldots k_r} \|_{\mathbb{R}^p},$$

and set $\mu := \delta / (C_1(B, r) \max\{c_{\text{sum}}, 1\})$, which is valid for all $\delta \leq \delta_0$ since then $C_1 \mu \leq 1$. By putting the layers together in parallel (one tree network per monomial) and including the exact output layer, we get $\widetilde{T}_\delta$ with $\sup_{y \in \mathcal{Q}} \| T(y) - \widetilde{T}_\delta(y) \|_{\mathbb{R}^p} \leq c_{\text{sum}} C_1 \mu \leq \delta$. Up to constants, the depth is that of a single tree, and the parallel combination multiplies the size by n^r. Since $\log(\mu^{-1}) \leq C \log(\delta^{-1})$ for $\delta \leq \delta_0$, it follows that $\operatorname{dep}(\widetilde{T}_\delta) \leq C \log(\delta^{-1})$ and $W(\widetilde{T}_\delta) \leq C \log(\delta^{-1})$, where C depends only on T, r, n, p and b. □