Supplementary Information

1 Proof of Theorem 1

We collect the standing conditions and supporting estimates used in the proof of Theorem 1. Throughout, $\|\cdot\|_2$ denotes the Euclidean norm on $\mathbb{R}^K$, and $\|\cdot\|_{w,2}$ is the fixed physical norm induced by the baseplate.

Assumption 1 (Standing conditions) *Fix $T > 0$ and let $\mathcal{K} \subset \mathbb{R}^K$ be compact. Let $\mathbf{a}(t)$ solve the reference reduced dynamics (10) with $\mathbf{a}(0) = \mathbf{a}_0$. Set $t_n = n\Delta t$ and $N_{\text{steps}} = T/\Delta t$.*

1. ***Containment on $\mathcal{K}$.*** *The reference trajectory satisfies $\mathbf{a}(t) \in \mathcal{K}$ for all $t \in [0, T]$. Moreover, there exists $\Delta t_0 > 0$ such that for $0 < \Delta t \le \Delta t_0$, all intermediate states produced by the reference and learned Strang schedules up to time T remain in $\mathcal{K}$.*
2. ***Regularity of block fields.*** *For each block i, the vector fields F_i^{ref} and $F_i^{\boldsymbol{\theta}}$ are Lipschitz on $\mathcal{K}$ with constants $L_i > 0$. In particular, $F^{\text{ref}} = \sum_{i=1}^{N_{\text{blk}}} F_i^{\text{ref}}$ is Lipschitz on $\mathcal{K}$.*
3. ***Within-block accuracy (second order).*** *For each i and each $\tau \in \{\Delta t/2, \Delta t\}$ used by the Strang schedule, there exist one-step maps $S_{i,\tau}^{\text{ref}}$ and $S_{i,\tau}^{\boldsymbol{\theta}}$ for the isolated sub-dynamics $\mathbf{a}_t = F_i^{\text{ref}}(\mathbf{a})$ and $\mathbf{a}_t = F_i^{\boldsymbol{\theta}}(\mathbf{a})$ such that, for all $\mathbf{a} \in \mathcal{K}$,*

$$\left\|S_{i,\tau}^{\text{ref}}(\mathbf{a}) - \varphi_{i,\tau}^{\text{ref}}(\mathbf{a})\right\|_2 \le C_i^{\text{ref}} \tau^3, \qquad \left\|S_{i,\tau}^{\boldsymbol{\theta}}(\mathbf{a}) - \varphi_{i,\tau}^{\boldsymbol{\theta}}(\mathbf{a})\right\|_2 \le C_i^{\boldsymbol{\theta}} \tau^3,$$

   where $\varphi_{i,t}^{\text{ref}}$ and $\varphi_{i,t}^{\boldsymbol{\theta}}$ denote the exact subflows of $\mathbf{a}_t = F_i^{\text{ref}}(\mathbf{a})$ and $\mathbf{a}_t = F_i^{\boldsymbol{\theta}}(\mathbf{a})$, and the constants $C_i^{\text{ref}}, C_i^{\boldsymbol{\theta}} > 0$ are independent of τ and Δt.
4. ***Reference macro-step stability.*** *Let $S_{\Delta t}^{\text{ref}}$ be the symmetric Strang composition built from $\{S_{i,\tau}^{\text{ref}}\}_{i=1}^{N_{\text{blk}}}$. There exist $\Delta t_0 > 0$ and $L \ge 0$ such that for all $\mathbf{a}, \mathbf{b} \in \mathcal{K}$ and $0 < \Delta t \le \Delta t_0$,*

$$\left\|S_{\Delta t}^{\text{ref}}(\mathbf{a}) - S_{\Delta t}^{\text{ref}}(\mathbf{b})\right\|_2 \le (1 + L\Delta t)\, \|\mathbf{a} - \mathbf{b}\|_2.$$

5. ***Uniform block mismatch on $\mathcal{K}$.*** *For each i,*

$$\varepsilon_i := \sup_{\mathbf{a} \in \mathcal{K}} \left\|F_i^{\boldsymbol{\theta}}(\mathbf{a}) - F_i^{\text{ref}}(\mathbf{a})\right\|_2 < \infty.$$

6. ***Bounded reconstruction.*** *There exists $C_\Phi > 0$ such that $\|\Phi_b \mathbf{v}\|_{w,2} \le C_\Phi \|\mathbf{v}\|_2$ for all $\mathbf{v} \in \mathbb{R}^K$.*

Assumption 1 (3) is standard in practice. Each block evolves on the same finite-dimensional coefficient interface, so its isolated sub-dynamics can be advanced by a block-adapted second-order one-step scheme, such as an exact subflow when available or a standard second-order integrator; see [1–3]. Assumption 1 (5) quantifies the uniform block mismatch on $\mathcal{K}$. Its size depends on the approximation class and on training quality, including the expressiveness of the block model and the coverage and

accuracy of the operator-matching samples. In principle, ε_i can be made small with richer model classes, more representative training data, and sufficiently effective optimization, provided the target block map is well approximated within the chosen class; see [4, 5].

After Assumption 1, we summarize the block-level structural consequences of the parameterization

$$F_i^{\boldsymbol{\theta}}(\mathbf{a}) = -\,G_i \nabla_{\mathbf{a}} E_i^{a,\boldsymbol{\theta}}(\mathbf{a}) + J_i \nabla_{\mathbf{a}} H_i^{a,\boldsymbol{\theta}}(\mathbf{a}) + R_i^a(\mathbf{a}).$$

For clarity, we first state the dissipative and conservative cases in isolation, where the residual term is absent; see [1, 6].

Property 2 (Energy dissipation for learned dissipative blocks) *Let $E_i^{a,\boldsymbol{\theta}} : \mathbb{R}^K \to \mathbb{R}$ be continuously differentiable and consider the isolated dissipative coefficient dynamics*

$$\mathbf{a}_t = -\,G_i\,\nabla_{\mathbf{a}} E_i^{a,\boldsymbol{\theta}}(\mathbf{a}), \qquad G_i^\top = G_i, \quad G_i \succeq 0. \tag{S1}$$

Here, $G_i^\top = G_i$ and $G_i \succeq 0$ mean that G_i is symmetric positive semidefinite. Then along any trajectory $\mathbf{a}(t)$ of (S1), the learned scalar generator is non-increasing:

$$\frac{d}{dt} E_i^{a,\boldsymbol{\theta}}\big(\mathbf{a}(t)\big) \le 0.$$

Proof By the chain rule,

$$\frac{d}{dt} E_i^{a,\boldsymbol{\theta}}\big(\mathbf{a}(t)\big) = \nabla_{\mathbf{a}} E_i^{a,\boldsymbol{\theta}}(\mathbf{a})^\top \mathbf{a}_t = -\nabla_{\mathbf{a}} E_i^{a,\boldsymbol{\theta}}(\mathbf{a})^\top G_i \nabla_{\mathbf{a}} E_i^{a,\boldsymbol{\theta}}(\mathbf{a}) \le 0,$$

because $G_i \succeq 0$ implies $\xi^\top G_i \xi \ge 0$ for all $\xi \in \mathbb{R}^K$. □

Property 3 (Hamiltonian conservation for learned conservative blocks) *Let $H_i^{a,\boldsymbol{\theta}} : \mathbb{R}^K \to \mathbb{R}$ be continuously differentiable and consider the isolated conservative coefficient dynamics*

$$\mathbf{a}_t = J_i\,\nabla_{\mathbf{a}} H_i^{a,\boldsymbol{\theta}}(\mathbf{a}), \qquad J_i^\top = -J_i. \tag{S2}$$

Here, $J_i^\top = -J_i$ means that J_i is skew-symmetric. Then along any trajectory $\mathbf{a}(t)$ of (S2), the learned Hamiltonian is conserved:

$$\frac{d}{dt} H_i^{a,\boldsymbol{\theta}}\big(\mathbf{a}(t)\big) = 0.$$

Proof By the chain rule,

$$\frac{d}{dt} H_i^{a,\boldsymbol{\theta}}\big(\mathbf{a}(t)\big) = \nabla_{\mathbf{a}} H_i^{a,\boldsymbol{\theta}}(\mathbf{a})^\top \mathbf{a}_t = \nabla_{\mathbf{a}} H_i^{a,\boldsymbol{\theta}}(\mathbf{a})^\top J_i \nabla_{\mathbf{a}} H_i^{a,\boldsymbol{\theta}}(\mathbf{a}).$$

For any skew-symmetric matrix J_i, one has $\xi^\top J_i \xi = 0$ for all $\xi \in \mathbb{R}^K$. Hence the derivative vanishes. □

Under Assumption 1, the following is a standard consequence of symmetric Strang splitting; see [1, 3].

Lemma 4 (Strang composition [1, 3]) *Under Assumption 1, for sufficiently small Δt, there exist constant $C_T^{\mathrm{spl}} > 0$, independent of Δt, such that*

$$\left\|\mathbf{a}_{N_{\mathrm{steps}}}^{\mathrm{ref}} - \mathbf{a}(T)\right\|_2 \le C_T^{\mathrm{spl}}\,\Delta t^2,$$

where $\mathbf{a}_{n+1}^{\mathrm{ref}} = S_{\Delta t}^{\mathrm{ref}}(\mathbf{a}_n^{\mathrm{ref}})$ and $\mathbf{a}_0^{\mathrm{ref}} = \mathbf{a}_0$. Moreover, suppose that, for a dissipative block i, its isolated within-block update satisfies

$$E_i^{a,\bullet}\big(S_{i,\tau}^{\bullet}(\mathbf{a})\big) \le E_i^{a,\bullet}(\mathbf{a}), \qquad \forall\,\mathbf{a}\in\mathcal{K},\ \tau\in\{\Delta t/2, \Delta t\},$$

or, for a conservative block i, its isolated within-block update satisfies

$$H_i^{a,\bullet}\big(S_{i,\tau}^{\bullet}(\mathbf{a})\big) = H_i^{a,\bullet}(\mathbf{a}), \qquad \forall\,\mathbf{a}\in\mathcal{K},\ \tau\in\{\Delta t/2, \Delta t\},$$

where $\bullet \in \{\mathrm{ref}, \boldsymbol{\theta}\}$. Then the same inequality or equality holds at the corresponding substeps inside the symmetric Strang schedule.

Lemma 5 (Exact subflow perturbation on $\mathcal{K}$) *Assume Assumption 1. Fix a block i. For any $\mathbf{a}\in\mathcal{K}$ and any $t \ge 0$ such that both subflows stay in $\mathcal{K}$ on $[0,t]$,*

$$\left\|\varphi_{i,t}^{\boldsymbol{\theta}}(\mathbf{a}) - \varphi_{i,t}^{\mathrm{ref}}(\mathbf{a})\right\|_2 \le \frac{e^{L_i t}-1}{L_i}\,\varepsilon_i \le t\,e^{L_i t}\,\varepsilon_i.$$

Proof Let $\mathbf{a}_\theta(s) = \varphi_{i,s}^{\boldsymbol{\theta}}(\mathbf{a})$ and $\mathbf{a}_{\mathrm{ref}}(s) = \varphi_{i,s}^{\mathrm{ref}}(\mathbf{a})$. Then

$$\frac{d}{ds}\big(\mathbf{a}_\theta - \mathbf{a}_{\mathrm{ref}}\big) = F_i^{\boldsymbol{\theta}}(\mathbf{a}_\theta) - F_i^{\mathrm{ref}}(\mathbf{a}_{\mathrm{ref}}) = \underbrace{F_i^{\boldsymbol{\theta}}(\mathbf{a}_\theta) - F_i^{\boldsymbol{\theta}}(\mathbf{a}_{\mathrm{ref}})}_{\text{Lipschitz}} + \underbrace{F_i^{\boldsymbol{\theta}}(\mathbf{a}_{\mathrm{ref}}) - F_i^{\mathrm{ref}}(\mathbf{a}_{\mathrm{ref}})}_{\le\varepsilon_i}.$$

Taking norms and using Lipschitz continuity on $\mathcal{K}$ yields

$$\frac{d}{ds}\|\mathbf{a}_\theta - \mathbf{a}_{\mathrm{ref}}\|_2 \le L_i\|\mathbf{a}_\theta - \mathbf{a}_{\mathrm{ref}}\|_2 + \varepsilon_i.$$

By Grönwall [7], $\|\mathbf{a}_\theta(t) - \mathbf{a}_{\mathrm{ref}}(t)\|_2 \le \int_0^t e^{L_i(t-s)}\varepsilon_i\,ds = \frac{e^{L_i t}-1}{L_i}\varepsilon_i$, and the second inequality follows from $\frac{e^x-1}{x} \le e^x$ for $x \ge 0$. □

Lemma 6 (Substep defect) *Assume Assumption 1. Fix a block i and let $\tau \in \{\Delta t/2, \Delta t\}$. Then for all $\mathbf{a}\in\mathcal{K}$,*

$$\left\|S_{i,\tau}^{\boldsymbol{\theta}}(\mathbf{a}) - S_{i,\tau}^{\mathrm{ref}}(\mathbf{a})\right\|_2 \le \tau e^{L_i\tau}\,\varepsilon_i + \big(C_i^{\boldsymbol{\theta}} + C_i^{\mathrm{ref}}\big)\tau^3.$$

Proof Insert and subtract the exact subflows:
$\left\|S_{i,\tau}^{\boldsymbol{\theta}}(\mathbf{a}) - S_{i,\tau}^{\mathrm{ref}}(\mathbf{a})\right\|_2 \le \left\|S_{i,\tau}^{\boldsymbol{\theta}}(\mathbf{a}) - \varphi_{i,\tau}^{\boldsymbol{\theta}}(\mathbf{a})\right\|_2 + \left\|\varphi_{i,\tau}^{\boldsymbol{\theta}}(\mathbf{a}) - \varphi_{i,\tau}^{\mathrm{ref}}(\mathbf{a})\right\|_2 + \left\|\varphi_{i,\tau}^{\mathrm{ref}}(\mathbf{a}) - S_{i,\tau}^{\mathrm{ref}}(\mathbf{a})\right\|_2$.
Apply Assumption 1(3) to the first and third terms, and Lemma 5 to the middle term. □

Lemma 7 (One-step macro defect) *Assume Assumption 1. Let $S_{\Delta t}^{\boldsymbol{\theta}}$ and $S_{\Delta t}^{\mathrm{ref}}$ be the learned and reference symmetric Strang macro-steps built from the same schedule and substep durations $\tau \in \{\Delta t/2, \Delta t\}$. Then there exist constants $C_{\mathrm{blk}}, C_{\mathrm{num}} > 0$, independent of Δt, such that for all $\mathbf{a}\in\mathcal{K}$ and sufficiently small Δt,*

$$\left\|S_{\Delta t}^{\boldsymbol{\theta}}(\mathbf{a}) - S_{\Delta t}^{\mathrm{ref}}(\mathbf{a})\right\|_2 \le C_{\mathrm{blk}}\,\Delta t \sum_{i=1}^{N_{\mathrm{blk}}} \varepsilon_i \;+\; C_{\mathrm{num}}\,\Delta t^3.$$

Proof Write the Strang schedule as a fixed finite composition of substep maps,

$$S^{\mathrm{ref}}_{\Delta t} = T^{\mathrm{ref}}_m \circ \cdots \circ T^{\mathrm{ref}}_1, \qquad S^{\boldsymbol{\theta}}_{\Delta t} = T^{\boldsymbol{\theta}}_m \circ \cdots \circ T^{\boldsymbol{\theta}}_1,$$

where each T_j is some $S_{i,\tau}$ with $\tau \in \{\Delta t/2, \Delta t\}$. Define intermediate states $\mathbf{z}_0 = \mathbf{a}$, $\mathbf{z}_j = T^{\mathrm{ref}}_j(\mathbf{z}_{j-1})$ and $\mathbf{w}_0 = \mathbf{a}$, $\mathbf{w}_j = T^{\boldsymbol{\theta}}_j(\mathbf{w}_{j-1})$. By containment, $\mathbf{z}_j, \mathbf{w}_j \in \mathcal{K}$.

A telescoping bound gives

$$\|\mathbf{w}_m - \mathbf{z}_m\|_2 \le \sum_{j=1}^{m} \left\| T^{\boldsymbol{\theta}}_m \circ \cdots \circ T^{\boldsymbol{\theta}}_{j+1}(\mathbf{w}_j) - T^{\boldsymbol{\theta}}_m \circ \cdots \circ T^{\boldsymbol{\theta}}_{j+1}(\mathbf{z}_j) \right\|_2 + \sum_{j=1}^{m} \|T^{\boldsymbol{\theta}}_j(\mathbf{z}_{j-1}) - T^{\mathrm{ref}}_j(\mathbf{z}_{j-1})\|_2.$$

Using Lipschitz continuity of each substep map on $\mathcal{K}$ (implied by Assumption 1(2) for sufficiently small τ) bounds the first sum by a constant multiple of $\max_j \|\mathbf{w}_j - \mathbf{z}_j\|_2$ and is absorbed into the second sum for small Δt. For the second sum, apply Lemma 6 to each occurrence of block i: each contributes $\mathcal{O}(\tau\,\varepsilon_i) + \mathcal{O}(\tau^3)$. Since the schedule contains a fixed finite number of substeps and $\sum_j \tau = \mathcal{O}(\Delta t)$, we obtain

$$\|S^{\boldsymbol{\theta}}_{\Delta t}(\mathbf{a}) - S^{\mathrm{ref}}_{\Delta t}(\mathbf{a})\|_2 \le C_{\mathrm{blk}}\,\Delta t \sum_i \varepsilon_i + C_{\mathrm{num}}\,\Delta t^3,$$

with constants depending only on the Lipschitz bounds on $\mathcal{K}$ and the fixed schedule. □

Lemma 8 (Discrete Grönwall [8]) *Let $\Delta t > 0$, $N \in \mathbb{N}$, and set $T_N := N\Delta t$. Assume $L \ge 0$, $\alpha \ge 0$, and a nonnegative sequence $\{e_n\}_{n=0}^{N}$ satisfies*

$$e_{n+1} \le (1 + L\Delta t)e_n + \alpha\Delta t, \qquad n = 0, \ldots, N-1, \qquad e_0 = 0.$$

Then

$$e_N \le \alpha\, T_N\, \exp(LT_N) \le \alpha\, T\, \exp(LT), \qquad \textit{for any } T \ge T_N.$$

Proof of Theorem 1 The structure claim is immediate: as shown in Lemma 4, the Strang macro-step is a fixed composition of the within-block maps $S^{\boldsymbol{\theta}}_{i,\tau}$, so any per-substep monotonicity (resp. invariance) of $E^{a,\boldsymbol{\theta}}_i$ (resp. $H^{a,\boldsymbol{\theta}}_i$) is inherited at the corresponding locations in the schedule.

For the error bound, decompose

$$\|\mathbf{a}^{\boldsymbol{\theta}}_{N_{\mathrm{steps}}} - \mathbf{a}(T)\|_2 \le \|\mathbf{a}^{\boldsymbol{\theta}}_{N_{\mathrm{steps}}} - \mathbf{a}^{\mathrm{ref}}_{N_{\mathrm{steps}}}\|_2 + \|\mathbf{a}^{\mathrm{ref}}_{N_{\mathrm{steps}}} - \mathbf{a}(T)\|_2.$$

The second term is controlled by Lemma 4. For the first term, set $\mathbf{e}_n := \mathbf{a}^{\boldsymbol{\theta}}_n - \mathbf{a}^{\mathrm{ref}}_n$. Using triangle inequality, for each step

$$\begin{aligned} \|\mathbf{e}_{n+1}\|_2 &= \left\| S^{\boldsymbol{\theta}}_{\Delta t}(\mathbf{a}^{\boldsymbol{\theta}}_n) - S^{\mathrm{ref}}_{\Delta t}(\mathbf{a}^{\mathrm{ref}}_n) \right\|_2 \\ &\le \underbrace{\left\| S^{\boldsymbol{\theta}}_{\Delta t}(\mathbf{a}^{\boldsymbol{\theta}}_n) - S^{\mathrm{ref}}_{\Delta t}(\mathbf{a}^{\boldsymbol{\theta}}_n) \right\|_2}_{\text{macro defect at the same input}} + \underbrace{\left\| S^{\mathrm{ref}}_{\Delta t}(\mathbf{a}^{\boldsymbol{\theta}}_n) - S^{\mathrm{ref}}_{\Delta t}(\mathbf{a}^{\mathrm{ref}}_n) \right\|_2}_{\text{stability of } S^{\mathrm{ref}}_{\Delta t}}. \end{aligned}$$

By Lemma 7, the first term is bounded by $C_{\mathrm{blk}}\Delta t \sum_{i=1}^{N_{\mathrm{blk}}} \varepsilon_i + C_{\mathrm{num}}\Delta t^3$. By Assumption 1 (4), the second term is bounded by $(1 + L\Delta t)\|\mathbf{e}_n\|_2$. Hence,

$$\|\mathbf{e}_{n+1}\|_2 \le (1 + L\Delta t)\|\mathbf{e}_n\|_2 + C_{\mathrm{blk}}\Delta t \sum_{i=1}^{N_{\mathrm{blk}}} \varepsilon_i + C_{\mathrm{num}}\Delta t^3.$$

Applying Lemma 8 gives

$$\|\mathbf{e}_{N_{\text{steps}}}\|_2 \le C_T^{\text{blk}}\, T \sum_{i=1}^{N_{\text{blk}}} \varepsilon_i + C_T^{\text{num}}\, T\Delta t^2.$$

Here we set constants $C_T^{\text{blk}} := C_{\text{blk}} e^{LT}$ (and similarly for C_T^{num}). Combining with Lemma 4 gives

$$\|\mathbf{a}^{\boldsymbol{\theta}}_{N_{\text{steps}}} - \mathbf{a}(T)\|_2 \le C_T^{\text{blk}}\, T \sum_{i=1}^{N_{\text{blk}}} \varepsilon_i + C_T^{\text{spl}} \Delta t^2.$$

Finally, by Assumption 1 (6), $\|u^{\boldsymbol{\theta}}_{N_{\text{steps}}} - u(T)\|_{w,2} = \|\Phi_b(\mathbf{a}^{\boldsymbol{\theta}}_{N_{\text{steps}}} - \mathbf{a}(T))\|_{w,2} \le C_\Phi \|\mathbf{a}^{\boldsymbol{\theta}}_{N_{\text{steps}}} - \mathbf{a}(T)\|_2$, which yields (16) after absorbing constants into C_T. □

2 Block architectures and training details

This section records implementation details for block parameterizations and pretraining used in the numerical experiments. Each block is pretrained by the unified operator-matching objective (11) on coefficient samples $\mathbf{a} \sim \mu_b$, where μ_b is a baseplate-dependent spectral-decay Gaussian prior on the retained coefficient interface.

2.1 Coefficient prior μ_b.

In 1D, we sample independent coordinates

$$a_k \sim \mathcal{N}(0, \sigma_k^2), \qquad \sigma_k = \frac{\text{amp}}{(1+k)^\alpha}, \qquad k = 1, \dots, K. \tag{S3}$$

In 2D and 3D bases, we retain modes indexed by multi-indices (j, ℓ) (2D) or (j, ℓ, m) (3D). For notational simplicity, we fix a bijection between the retained multi-indices and a scalar index $k \in \{1, \dots, K\}$, and denote the corresponding coefficient by a_k. We then set

$$\sigma_k = \frac{\text{amp}}{\big(1 + \|k\|_2\big)^\alpha}. \tag{S4}$$

where k denotes the underlying multi-index. Here, we take amp $= 1$, $\alpha = 0.5$. For periodic Fourier baseplates, we draw coefficients in the real-valued Hermitian-packed representation: for each non self-conjugate mode, we sample $\Re(a_k), \Im(a_k) \sim \mathcal{N}\big(0, \sigma_k^2\big)$ independently, while for self-conjugate modes only the real part is sampled; unpacking enforces Hermitian symmetry. For cosine (Neumann) baseplates, coefficients are real, and we sample $a_k \sim \mathcal{N}\big(0, \sigma_k^2\big)$ using (S4). Unless otherwise stated, we generate 20,000 coefficient samples for training.

Targets in (11) are generated by the corresponding exact Galerkin/spectral operators restricted to the retained modes. Optimization uses AdamW with StepLR schedules, and all pretrained blocks are reused unchanged at inference time.

2.2 1D Dirichlet baseplate (Shen–Legendre)

We consider $\Omega = (-1, 1)$ with $u(\pm 1) = 0$ and represent fields in Shen's Legendre basis [9]

$$\phi_k(x) = L_{k-1}(x) - L_{k+1}(x), \qquad k = 1, \ldots, K,$$

so that $\phi_k(\pm 1) = 0$ and $u(x) = \sum_{k=1}^{K} a_k \, \phi_k(x)$. Grid evaluations use a Gauss–Legendre quadrature grid $\{x_q\}_{q=1}^{Q}$ with basis matrix $\Phi_b \in \mathbb{R}^{Q \times K}$, $(\Phi_b)_{qk} = \phi_k(x_q)$. We use $Q = 256$ and $K = 96$. The baseplate projection $\mathcal{P}_b$ is the discrete L^2 projection induced by the mass matrix $M_{ij} = \langle \phi_i, \phi_j \rangle_{L^2}$.

Diffusion block ($u \mapsto u_{xx}$).

We parameterize the energy generator $E^{a,\boldsymbol{\theta}}_{u_{xx}} : \mathbb{R}^K \to \mathbb{R}$ by an MLP (4 hidden layers, width 128, GELU activations) and define the learned vector field by the coefficient-space gradient-flow form,

$$F^{\boldsymbol{\theta}}_{u_{xx}}(\mathbf{a}) \;=\; -\, G \, \nabla_a E^{a,\boldsymbol{\theta}}_{u_{xx}}(\mathbf{a}), \tag{S5}$$

where $G = M^{-1}$ is the fixed mobility induced by the discrete L^2 metric on the Shen space. We train with AdamW (learning rate 10^{-3}) and StepLR (step size 50, decay factor 0.3) for 200 epochs with batch size 128.

Transport block ($u \mapsto u\, u_x$).

We learn a pointwise density $h_{\boldsymbol{\theta}} : \mathbb{R} \to \mathbb{R}$ (depth 4, width 128, GELU) and induce a Hamiltonian generator via the density construction described in Section 2. The learned transport vector field takes the form

$$F^{\boldsymbol{\theta}}_{uu_x}(\mathbf{a}) \;=\; J \, \nabla_a H^{a,\boldsymbol{\theta}}_{uu_x}(\mathbf{a}), \tag{S6}$$

where $J = M^{-1} S$ is fixed and represents ∂_x on the Shen space, with $S_{ij} = \langle \partial_x \phi_i, \phi_j \rangle_{L^2}$. We train $h_{\boldsymbol{\theta}}$ with AdamW (learning rate 10^{-4}, weight decay 10^{-4}) for 100 epochs with batch size 128.

Extended Data Figs. 1a-1b provide block-level sanity checks on a held-out coefficient state, comparing the learned operators against their Galerkin-projected reference counterparts in physical space. The reported discrepancies are on the order of 10^{-3} in the weighted L^2 norm evaluated at Gauss–Legendre nodes.

2.3 2D periodic baseplate (Fourier)

We consider $\Omega = [0, 2\pi)^2$ and represent real fields by a band-limited Fourier expansion

$$u(x, y) = \sum_{|j| \le K_{\mathrm{cut}}} \sum_{|\ell| \le K_{\mathrm{cut}}} a_{j,\ell} \, e^{i(jx + \ell y)}, \qquad a_{-j,-\ell} = \overline{a_{j,\ell}}.$$

Coefficients are stored in a real Hermitian-packed coordinate vector $\mathbf{a} \in \mathbb{R}^K$ that uniquely represents a real band-limited field. Grid evaluations use an $N \times N$ uniform

grid with FFT/iFFT transforms for Φ_b and $\mathcal{P}_b$. Unless stated otherwise, $N = 64$ and we retain modes up to $K_{\text{cut}} = 21$, so the retained complex mode set is $(2K_{\text{cut}} + 1)^2$ and the packed real dimension is denoted by K.

Laplacian diffusion block ($u \mapsto \Delta u$).

Since the Laplacian is mode-decoupled in the Fourier baseplate, we restrict the energy generator to a structured quadratic form,

$$E_{\Delta}^{a,\boldsymbol{\theta}}(\mathbf{a}) \;=\; \frac{1}{2}\,\mathbf{a}^{\top}\mathrm{diag}(\mathbf{c}^{\boldsymbol{\theta}})\,\mathbf{a}, \qquad \mathbf{c}^{\boldsymbol{\theta}} \in \mathbb{R}^{K}. \tag{S7}$$

We then define the diffusion vector field by the gradient-flow template

$$F_{\Delta}^{\boldsymbol{\theta}}(\mathbf{a}) \;=\; -\,G\,\nabla_a E_{\Delta}^{a,\boldsymbol{\theta}}(\mathbf{a}), \tag{S8}$$

with $G = I$ for the orthonormal Fourier coefficient metric. Training uses AdamW (learning rate 10^{-3}) with StepLR (step size 40, decay factor 0.3) for 80 epochs (batch size 128).

Hamiltonian transport blocks ($u \mapsto uu_x$ and $u \mapsto uu_y$).

We train two pointwise density networks $\rho_{\boldsymbol{\theta}_x}$ and $\rho_{\boldsymbol{\theta}_y}$ with identical architecture (depth 4, width 128, GELU), and assemble directional transport via fixed derivative operators J_x and J_y that represent ∂_x and ∂_y on the retained modes:

$$F_{uu_x}^{\boldsymbol{\theta}}(\mathbf{a}) = J_x\,\nabla_a H_{uu_x}^{a,\boldsymbol{\theta}_x}(\mathbf{a}), \qquad F_{uu_y}^{\boldsymbol{\theta}}(\mathbf{a}) = J_y\,\nabla_a H_{uu_y}^{a,\boldsymbol{\theta}_y}(\mathbf{a}). \tag{S9}$$

Training uses AdamW (learning rate 5×10^{-4}, weight decay 10^{-6}) for 500 epochs (batch size 16), with StepLR (step size 150, decay factor 0.5).

Poisson inversion block ($\Delta\psi = \omega$).

Given ω, we predict ψ on the same retained Fourier modes. We parameterize a diagonal quadratic generator with softplus-constrained weights and fit it using exact Fourier inversion targets. Training uses AdamW for 200 epochs (batch size 128) with StepLR (step size 80, decay factor 0.3).

2.4 2D Neumann baseplate (cosine/DCT)

We consider $\Omega = [0,1]^2$ with homogeneous Neumann boundary conditions and represent fields in a tensor-product cosine basis

$$u(x,y) = \sum_{j=0}^{K_{\text{cut}}} \sum_{\ell=0}^{K_{\text{cut}}} a_{j,\ell}\,\cos(\pi j x)\cos(\pi \ell y).$$

Grid evaluations use an endpoint $N \times N$ grid with IDCT-I/DCT-I transforms for Φ_b and $\mathcal{P}_b$. Unless stated otherwise, $N = 65$ and $K_{\text{cut}} = \min\big(K_{\max}, \lfloor (N-1)/3 \rfloor\big) = 21$,

so $K = (K_{\text{cut}} + 1)^2$. Let $\mathbf{a} \in \mathbb{R}^K$ be the vector obtained by stacking the coefficients $\{a_{k,\ell}\}$ in a fixed order.

Neumann Laplacian diffusion block ($u \mapsto \Delta u$).

This block follows the same Laplacian design and training recipe as the Fourier-baseplate diffusion block (S8), with the only change being the baseplate.

2.5 3D periodic baseplate (Fourier/FFT)

We consider $\Omega = [0, 2\pi)^3$ and represent real fields by a band-limited 3D Fourier expansion

$$u(x, y, z) = \sum_{|j| \le K_{\text{cut}}} \sum_{|\ell| \le K_{\text{cut}}} \sum_{|m| \le K_{\text{cut}}} a_{j,\ell,m}\, e^{i(jx+\ell y+mz)}, \qquad a_{-j,-\ell,-m} = \overline{a_{j,\ell,m}}.$$

Coefficients are stored in a real Hermitian-packed vector $\mathbf{a} \in \mathbb{R}^K$. Grid evaluations use an $N \times N \times N$ uniform grid with 3D FFT/iFFT transforms for Φ_b and $\mathcal{P}_b$. We retain modes up to K_{cut}, following the same truncation convention as in the 2D periodic baseplate.

3D Laplacian diffusion block ($u \mapsto \Delta u$).

The 3D Laplacian block is trained in exactly the same way as the 2D periodic Fourier Laplacian block (S8), except that the baseplate is the 3D Fourier expansion and the retained index set is three-dimensional.

3 Additional numerical experiments

This section reports supporting experiments that follow the same baseplate interface, pretrained-block reuse, and Strang block-composition protocol as in the main text, and collectively reinforce the advantages of LegONet in accuracy, stability, and plug-and-play reuse across PDE settings. Each experiment is evaluated by the weighted relative error rel and the normalized pointwise profile $e(x)$ defined in (4). Unless otherwise stated, linear or stiff coefficient-space substeps use exact or Crank–Nicolson-type updates, whereas nonlinear substeps, including reaction, forcing, and transport terms, use second-order explicit updates such as Heun schemes, or exact pointwise maps when available. Thus the within-block updates are consistent with the second-order assumption used in the analysis.

3.1 1D Dirichlet domains

Experiment A1: 1D Ginzburg–Landau.

We consider

$$u_t = u_{xx} - (u + u^3), \qquad x \in (-1, 1), \qquad u(\pm 1, t) = 0, \tag{S10}$$

advanced by a diffusion–reaction Strang composition $S^{\boldsymbol{\theta}}_{u_{xx},\Delta t/2}\circ S_{-(u+u^3),\Delta t}\circ S^{\boldsymbol{\theta}}_{u_{xx},\Delta t/2}$. The diffusion substep u_{xx} uses a Crank–Nicolson update in the Shen–Legendre coefficient space, while the reaction substep applies a second-order Heun update to the projected local term $-(u+u^3)$ on quadrature nodes. We take $\Delta t = 10^{-4}$ and $N_{\text{steps}} = 10^3$ ($T = 0.1$). Fig. 3a shows snapshots and the corresponding $e(x)$. Extended Data Fig. 1c reports the energy diagnostics.

Experiment A2: 1D heat equation with time-dependent Dirichlet data.

We consider

$$u_t = \nu u_{xx}, \qquad x \in (-1,1), \qquad u(-1,t) = A(t), \quad u(1,t) = B(t), \tag{S11}$$

with $\nu = 2\times 10^{-2}$ and impose the non-homogeneous boundary data by lifting, $u = u_0 + u_{\text{lift}}$, where $u_{\text{lift}}(x,t) = \frac{1-x}{2}A(t) + \frac{1+x}{2}B(t)$ so that $u_0(\pm 1,t) = 0$. The interior component satisfies

$$(u_0)_t = \nu (u_0)_{xx} + g(x,t), \qquad g(x,t) = -(u_{\text{lift}})_t,$$

and is advanced by a diffusion–forcing Strang composition $S_{g(x,t),\Delta t/2} \circ S^{\boldsymbol{\theta}}_{\nu(u_0)_{xx},\Delta t} \circ S_{g(x,t),\Delta t/2}$, with Crank–Nicolson diffusion and Heun forcing. We take $A(t) = A_0 + \alpha_A \sin(2\pi\omega t)$, $B(t) = B_0 + \alpha_B \cos(2\pi\omega t)$ with $(A_0, B_0) = (0.2, -0.2)$, $\alpha_A = \alpha_B = 5.6$, and $\omega = 1$. We use $\Delta t = 10^{-3}$ and $N_{\text{steps}} = 2\times 10^4$ ($T = 20$). Fig. 3b reports snapshots and $e(x)$. Energy diagnostics are collected in Extended Data Fig. 1d.

3.2 2D periodic domains

Experiment A3: 2D Allen–Cahn.

We consider

$$u_t = \varepsilon\,\Delta u + u - u^3, \qquad (x,y) \in \Omega = [0,2\pi)^2, \tag{S12}$$

with $\varepsilon = 10^{-2}$, advanced by a diffusion–reaction Strang composition. The diffusion substep applies a coefficient-space update for $\varepsilon\Delta$ on the retained Fourier modes, and the reaction substep applies a Heun update to $u - u^3$ pointwise on the grid followed by projection and de-aliasing. We take $\Delta t = 10^{-3}$ and $N_{\text{steps}} = 10^5$ ($T = 100$). Fig. 3c and Extended Data Figs. 2a, 2b report snapshots, rel, and energy decay.

Experiment A4: 2D vector Burgers.

We consider

$$\begin{aligned} u_t + u\,u_x + v\,u_y &= \nu\,\Delta u, \\ v_t + u\,v_x + v\,v_y &= \nu\,\Delta v, \end{aligned} \qquad (x,y) \in \Omega = [0,2\pi)^2, \tag{S13}$$

with $\nu = 10^{-3}$, advanced by a symmetric Strang composition on the retained Fourier modes as shown in Fig. 2c. Diffusion uses an implicit coefficient-space update, while transport is evaluated pseudo-spectrally: the self-advection terms are provided by the pretrained density blocks for uu_x and uu_y, and the cross terms are computed on

the grid with Fourier differentiation before projection. We take $\Delta t = 2 \times 10^{-3}$ and $N_{\text{steps}} = 1.5 \times 10^4$ ($T = 30$). Figs. 3e, 3f and Extended Data Figs. 2c, 2d report snapshots and rel for both components.

3.3 2D Neumann domains

Experiment A5: 2D Swift–Hohenberg.

We consider

$$\begin{cases} u_t = -(\Delta + k_0^2)^2 u + \mu u - u^3, & (x, y) \in \Omega = [0, 1]^2, \\ \partial_n u = 0, \quad \partial_n(\Delta u) = 0, & (x, y) \in \partial\Omega, \end{cases} \tag{S14}$$

with $\mu = 0.5$ and $k_0 = 6$, advanced by a linear–reaction Strang composition. The stiff linear substep advances $u_t = -(\Delta + k_0^2)^2 u$ in cosine coefficient space by mode-wise diagonal updates, and the reaction substep applies a pointwise update to $\mu u - u^3$ on the physical grid followed by projection to the retained cosine modes. We take $\Delta t = 10^{-2}$ and $N_{\text{steps}} = 2 \times 10^3$ ($T = 20$). Fig. 3d and Extended Data Fig. 2e report snapshots and rel.

Experiment A6: 2D Cahn–Hilliard with non-homogeneous Neumann flux.

We consider

$$u_t = \Delta\mu, \qquad \mu = -\varepsilon^2 \Delta u + (u^3 - u), \qquad (x, y) \in \Omega = [0, 1]^2, \tag{S15}$$

with boundary flux $\partial_y u(x, 0, t) = g(x)$ and $\partial_y u(x, 1, t) = -g(x)$, where $g(x) = g_{\text{amp}} \cos(\pi x)$, and homogeneous Neumann conditions in x. We set $\varepsilon = 5 \times 10^{-2}$ and $g_{\text{amp}} = 5 \times 10^{-2}$. We impose the flux by a harmonic lifting $u = u_0 + u_{\text{lift}}$ with $\Delta u_{\text{lift}} = 0$, so that u_0 satisfies homogeneous Neumann conditions and is represented in the cosine basis. Time stepping advances u_0 by a Strang splitting between the stiff linear operator $-\varepsilon^2 \Delta^2 u_0$ and the remaining nonlinear term evaluated on the physical grid. We take $\Delta t = 5 \times 10^{-4}$ and $N_{\text{steps}} = 2 \times 10^4$ ($T = 10$). Fig. 3g and Extended Data Fig. 2f report snapshots and rel.

3.4 3D periodic domains

Experiment A7: 3D Allen–Cahn with volume constraint.

We consider

$$u_t = \varepsilon \Delta u + u - u^3 - \lambda(t), \qquad \lambda(t) = \langle u - u^3 \rangle, \qquad (x, y, z) \in \Omega = [0, 2\pi)^3, \tag{S16}$$

with $\varepsilon = 10^{-2}$, advanced by the same diffusion–reaction Strang template as in 2D, with the spatial-average correction applied in the reaction step. Here $\langle f \rangle := |\Omega|^{-1} \int_\Omega f(x)\, dx$ denotes the spatial average. We take $\Delta t = 5 \times 10^{-3}$ and $N_{\text{steps}} = 800$ ($T = 4$).

This experiment deliberately departs from the spectral-decay Gaussian prior (S4) used in block pretraining and instead employs a two-phase voxel initialization with sharp interfaces as shown in Fig. 3h. Such non-smooth morphology induces a substantially different coefficient distribution, activating higher-frequency modes absent from the training prior. The test therefore probes robustness under coefficient-distribution shift. Fig. 3h reports phase renderings and rel.

OOD initial-condition priors for 3D Swift–Hohenberg

We evaluate robustness under two out-of-distribution initial-condition priors while keeping the same PDE parameters, baseplate, and rollout solver. The in-distribution (ID) prior follows the spectral Gaussian construction used throughout the paper, with modewise standard deviation $\sigma_k = \mathrm{amp}/(1 + \|k\|_2)^\alpha$. OOD1 uses the same Gaussian family but removes spectral decay by setting $\alpha = 0$, so that all retained Fourier modes have identical variance and the resulting fields carry increased high-frequency energy. OOD2 uses a blocky two-phase prior that introduces sharp interfaces in physical space: we first sample a piecewise-constant random field on a coarse grid of size N_c^3 with $N_c = 8$,

$$u_0^{(c)}(\xi) = -1 + 2B(\xi), \qquad B(\xi) \sim \mathrm{Bernoulli}(p), \qquad \xi \in \{1, \dots, N_c\}^3,$$

so that $\mathbb{P}[u_0^{(c)}(\xi) = +1] = p$ and $\mathbb{P}[u_0^{(c)}(\xi) = -1] = 1 - p$ independently over ξ. We set $p = 0.35$, upsample $u_0^{(c)}$ to the N^3 simulation grid, and apply the same amplitude normalization as in the ID setting. Together, OOD1 and OOD2 probe robustness to increased high-frequency content in coefficient space and to non-smooth initial interfaces in physical space, respectively.

4 Baseline configuration

This section summarizes the baseline setups for the main-text solver-level comparisons. We compare LegONet with two widely used supervised neural-operator baselines: Fourier Neural Operator (FNO) [10] and DeepONet [11], representing canonical spectral-convolution and branch–trunk architectures. All methods are evaluated in closed loop under an identical rollout protocol, starting from the same initial condition.

For the 1D Burgers experiment, we additionally include a standard physics-informed neural network (PINN) baseline, which learns a continuous surrogate $u^{\boldsymbol{\theta}}(x, t)$ from PDE residual and boundary/initial constraints without trajectory supervision. We do not include PINNs in 2D/3D because a like-for-like solver-level comparison would require long-horizon optimization over high-dimensional space–time fields under stiff/higher-order operators and coupled constraints. This makes training highly sensitive to collocation design and loss balancing, preventing a controlled comparison in our setting.

4.1 Baseline models

FNO.

We use an FNO time-stepper in residual-update form,

$$\mathbf{u}_{n+1} \;=\; \mathbf{u}_n \;+\; \alpha\, \delta^{\boldsymbol{\theta}}(\mathbf{u}_n; \Delta t), \tag{S17}$$

where $\delta^{\boldsymbol{\theta}}$ is an FNO backbone that maps the current discrete field to an increment on the same set of nodes, and $\alpha > 0$ is a residual scale. Residual parameterizations are commonly used to stabilize long-horizon rollouts of neural time-steppers.

DeepONet.

We use the standard DeepONet branch–trunk factorization to represent the one-step operator as a low-rank bilinear form with a residual update,

$$\mathbf{u}_{n+1}(\mathbf{x}) \;=\; \mathbf{u}_n(\mathbf{x}) \;+\; \alpha\, \big\langle b^{\boldsymbol{\theta}}(\mathbf{s}_n),\, t^{\boldsymbol{\theta}}(\mathbf{x}) \big\rangle,$$

where $\alpha > 0$ is a residual scale and $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product in $\mathbb{R}^r$. The branch network $b^{\boldsymbol{\theta}} : \mathbb{R}^{N_s} \to \mathbb{R}^r$ takes as input sensor measurements $\mathbf{s}_n = \big(u_n(\mathbf{x}_1), \ldots, u_n(\mathbf{x}_{N_s})\big)$ and outputs coefficients in a rank-r latent space. The trunk network $t^{\boldsymbol{\theta}} : \Omega \to \mathbb{R}^r$ maps a query location $\mathbf{x} \in \Omega$ to a location-dependent basis vector. The increment is evaluated as $\langle b^{\boldsymbol{\theta}}(\mathbf{s}_n), t^{\boldsymbol{\theta}}(\mathbf{x}) \rangle = \sum_{j=1}^{r} b_j^{\boldsymbol{\theta}}(\mathbf{s}_n)\, t_j^{\boldsymbol{\theta}}(\mathbf{x})$, and is computed at the experiment-specific evaluation nodes.

For each experiment, baselines are trained to advance the experiment-specific discretization and are evaluated on the same macro-time snapshots and evaluation nodes. We use standard instantiations of FNO and DeepONet and tune widths (and, for DeepONet, the rank) so that parameter counts are comparable to those of the corresponding LegONet blocks. Exact configurations are reported below. A key failure mode of teacher-forced one-step training is train–test mismatch in closed-loop rollouts. To mitigate this effect, we train FNO and DeepONet with a rollout-aware objective over a short unrolled window. In contrast to LegONet block pretraining, which matches instantaneous operator labels via (11), the supervised baselines minimize a rollout-aware K_{roll}-step loss over short trajectory windows:

$$\min_{\boldsymbol{\theta}} \; \mathbb{E}_{(\mathbf{u}_n, \mathbf{u}_{n+1}, \ldots, \mathbf{u}_{n+K_{\mathrm{roll}}})} \left[\frac{1}{K_{\mathrm{roll}}} \sum_{k=1}^{K_{\mathrm{roll}}} \big\| \mathbf{u}_{n+k} - \widehat{\mathbf{u}}_{n+k}^{\boldsymbol{\theta}} \big\|_{w,2}^2 \right], \tag{S18}$$

where $\widehat{\mathbf{u}}_n^{\boldsymbol{\theta}} = \mathbf{u}_n$ and $\widehat{\mathbf{u}}_{n+k}^{\boldsymbol{\theta}} = \mathcal{S}_{\boldsymbol{\theta}}\big(\widehat{\mathbf{u}}_{n+k-1}^{\boldsymbol{\theta}}\big)$ for $k \geq 1$. Here $\mathcal{S}_{\boldsymbol{\theta}}$ denotes the learned one-step snapshot map (FNO or DeepONet). For each experiment, we choose Δt_{eff}, K_{roll}, and N_{traj} defined below so that the resulting number of rollout windows is matched, up to a small tolerance, to the total block-training samples used by the corresponding LegONet assembly.

Experiment 1 (1D Burgers)

For baseline feasibility, we train supervised operator learners on a coarser effective time step rather than the fine micro step used by the reference solver. When the snapshot spacing is too small, the one-step map becomes near-identity and teacher-forced training is dominated by trivial identity fitting, which exacerbates train–test mismatch in closed-loop rollouts. We therefore use a shortened horizon $T = 0.2$ and an effective step $\Delta t_{\text{eff}} = 10^{-3}$, generate $N_{\text{traj}} = 220$ reference rollouts, and store trajectories on Gauss–Legendre quadrature nodes with $Q = 256$ points as a tensor of shape $(N_{\text{traj}}, T_m, Q)$, where $T_m = T/\Delta t_{\text{eff}} + 1 = 201$. All reported errors are computed on these Gauss–Legendre nodes using the weighted norm $\|\cdot\|_{w,2}$ in (4). Note that each LegONet block is pretrained by operator matching using 20,000 independent coefficient samples drawn from the Gaussian prior, yielding 40,000 samples total for the two blocks. The resulting rollout-window count matches the LegONet training-sample budget for this experiment.

While evaluation is always performed on the Gauss–Legendre nodes, the training interface depends on the baseline architecture. DeepONet is trained directly on the quadrature representation. FNO requires an equispaced grid to enable FFT-based spectral convolutions, so we resample each snapshot from the Gauss–Legendre nodes to an equispaced grid of the same resolution and use a fixed odd extension across the boundaries before the FFT layers, which is more consistent with the homogeneous Dirichlet boundary values. PINN is trained in continuous space–time using interior and boundary collocation points, and is evaluated by querying $u^\theta(x, t_n)$ at the Gauss–Legendre nodes.

To reduce capacity confounders, we tune the baseline widths and ranks so that their trainable parameter counts are comparable to those of the full LegONet assembly used in this experiment. Our FNO uses retained Fourier modes $m = 16$, channel width 40, depth 2, and residual scaling $\alpha = 0.1$. DeepONet follows the branch–trunk factorization: the branch ingests $N_s = 64$ sensor values of the current snapshot (selected as a fixed subset of the $Q = 256$ Gauss–Legendre nodes) and outputs a latent vector in $\mathbb{R}^r$ with $r = 145$, while the trunk maps query locations x to $\mathbb{R}^r$. We use 3-layer MLPs (width 128, GELU) for both branch and trunk, and apply the same residual scaling $\alpha = 0.1$ in the one-step map. We train FNO and DeepONet with the rollout-aware objective (S18) using $K_{\text{roll}} = 25$ and AdamW (learning rate 5×10^{-4}, weight decay 10^{-6}), batch size 16, for 2000 epochs.

The PINN represents the solution as a continuous function on $x \in [-1, 1]$ and $t \in [0, T]$ and is trained without trajectory supervision by minimizing a weighted sum of a PDE-residual loss and initial/boundary penalties,

$$\min_{\boldsymbol{\theta}} \; w_{\text{phys}}\,\mathcal{L}_{\text{phys}} \;+\; w_{\text{ic}}\,\mathcal{L}_{\text{ic}} \;+\; w_{\text{bc}}\,\mathcal{L}_{\text{bc}}, \qquad (w_{\text{phys}}, w_{\text{ic}}, w_{\text{bc}}) = (1, 10, 10), \tag{S19}$$

where $\mathcal{L}_{\text{phys}}$ is the mean-squared PDE residual evaluated at interior collocation points and $\mathcal{L}_{\text{ic}}, \mathcal{L}_{\text{bc}}$ are mean-squared penalties enforcing the initial and boundary conditions. We use an MLP with width 149, depth 6, and tanh activations, optimized with Adam (learning rate 2×10^{-4}) for 8000 epochs. At each epoch, we resample collocation

points uniformly in space–time and use $N_{\text{int}} = 4096$ interior points and $N_{\text{bd}} = 1024$ initial/boundary points.

Experiment 2 (2D Navier–Stokes)

We generate $N_{\text{traj}} = 800$ reference rollouts on a 64×64 grid up to $T = 50$ and store effective snapshots with step size $\Delta t_{\text{eff}} = 1.0$. This choice matches the LegONet sample budget up to a small tolerance. We train an FNO time-stepper in residual-update form with scaling $\alpha = 1$, using a 2D spectral-convolution backbone with retained modes $m = 12$, channel width 64, and depth 4. Training uses the rollout-aware objective (S18) with unroll length $K_{\text{roll}} = 10$ and AdamW (learning rate 10^{-3}, weight decay 10^{-4}), batch size 4, for 2000 epochs. DeepONet follows the standard branch–trunk construction on the same effective-time data, also in residual-update form with $\alpha = 1$. The branch network ingests $N_s = 1024$ fixed spatial sensors (a uniform subsampling of the 64×64 grid) and outputs a rank-r latent vector with $r = 128$, while the trunk maps 2D query locations $\mathbf{x} = (x, y)$ to $\mathbb{R}^r$. Both branch and trunk are implemented as MLPs (width 256, depth 3, GELU), and training follows the same rollout-aware protocol and model selection criteria as for FNO. In this experiment, LegONet reuses a pretrained Laplacian diffusion block and a Poisson inversion block on the periodic Fourier baseplate with structured diagonal parametrization. Supervised baselines, by contrast, must approximate the full nonlinear time-advance operator at step size Δt_{eff} and therefore require substantially higher-capacity networks for stable long-horizon rollouts.

Experiment 3 (3D Swift–Hohenberg)

We construct a supervised rollout dataset using the Strang-splitting reference solver on a 64^3 periodic grid up to $T = 30$. We record effective snapshots every $\Delta t_{\text{eff}} = 0.15$. The dataset contains $N_{\text{traj}} = 100$ trajectories, yielding $T_m = T/\Delta t_{\text{eff}} + 1 = 201$ stored frames per trajectory.

We train a 3D FNO time-stepper using the residual update with fixed scale $\alpha = 0.6$. The model uses a spectral-convolution backbone with retained modes $m = 16$, channel width 64, depth 6, and LayerNorm. Training uses the rollout-aware objective (S18) with unroll length $K_{\text{roll}} = 8$ and AdamW (learning rate 6×10^{-4}, weight decay 10^{-6}) for 2000 epochs. We do not include a DeepONet baseline in 3D because the standard branch–trunk evaluation requires producing an r-dimensional trunk feature at every spatial query location to form a full-field update. On a 64^3 grid, this introduces a prohibitive compute and memory footprint for rollout-aware training under the same closed-loop field supervision used for FNO and LegONet. Subsampling query points would change the evaluation interface and would no longer constitute a like-for-like solver-level comparison.

References

[1] Hairer, E., Lubich, C., Wanner, G.: Structure-preserving algorithms for ordinary differential equations. Geometric numerical integration **31** (2006)

[2] Sanz-Serna, J.-M., Calvo, M.-P.: Numerical Hamiltonian Problems vol. 7. Courier Dover Publications, Mineola, NY (2018)

[3] Strang, G.: On the construction and comparison of difference schemes. SIAM Journal on Numerical Analysis **5**(3), 506–517 (1968)

[4] Cybenko, G.: Approximation by superpositions of a sigmoidal function. Mathematics of control, signals and systems **2**(4), 303–314 (1989)

[5] Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Networks **2**(5), 359–366 (1989)

[6] Ambrosio, L., Gigli, N., Savaré, G.: Gradient Flows: in Metric Spaces and in the Space of Probability Measures. Springer, Berlin, Heidelberg (2005)

[7] Hartman, P.: Ordinary Differential Equations. SIAM, Philadelphia, PA (2002)

[8] Hairer, E., Wanner, G., Nørsett, S.P.: Solving Ordinary Differential Equations I: Nonstiff Problems. Springer, Berlin, Heidelberg (1993)

[9] Shen, J.: Efficient spectral-Galerkin method I. Direct solvers of second-and fourth-order equations using Legendre polynomials. SIAM Journal on Scientific Computing **15**(6), 1489–1505 (1994)

[10] Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., Anandkumar, A.: Fourier neural operator for parametric partial differential equations. arXiv preprint arXiv:2010.08895 (2020)

[11] Lu, L., Jin, P., Karniadakis, G.E.: DeepONet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. arXiv preprint arXiv:1910.03193 (2019)