PI-NCA: NEURAL CELLULAR AUTOMATA THAT CONSERVE BY CONSTRUCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Learned PDE emulators can be computationally expensive and difficult to scale. Neural cellular automata (NCAs) are therefore an appealing template: a small local rule shared by every cell mirrors the locality of a differential operator. Used directly, however, an NCA predicts unconstrained state increments; in our evaluation, its mass drifts by tens of units, and on Navier–Stokes it performs worse than predicting no change. We introduce Physics-Informed Neural Cellular Automata (PI-NCA) and its multi-channel variant, MC-PI-NCA, to address this failure structurally. PI-NCA makes the shared rule output a flux rather than an increment and advances the state by the flux's periodic discrete divergence, thereby enforcing exact conservation of the field total for any learned weights. For coupled systems, the construction predicts one flux per field and conserves each field's total independently; MC-PI-NCA is a wider instantiation designed for joint multi-field perception. Either architecture can also apply a parameter-free projection that keeps predictions within a physically valid range while preserving mass. We compare both architectures with the Fourier neural operator (FNO), full- and matched-size ResNet and U-Net surrogates, the standard NCA, and an identity floor on ten 2-D phenomena, using five to ten seeds and Holm-corrected paired tests. Ablations isolate the effects of the flux head and projection and identify regimes in which the conservation prior should not be used. On the shallow-water equations, PI-NCA and MC-PI-NCA are the two most accurate models while using parameter counts $112\times$ and $54\times$ smaller than that of the FNO. On Cahn–Hilliard, PI-NCA has relative L^2 error 20.63% lower than the best baseline and 43.50% lower than the FNO; on heat and advection–diffusion, the proposed models rank second only to the FNO with parameter counts 60–$130\times$ smaller. PI-NCA outperforms the NCA on seven of the ten evaluated phenomena with 33% fewer parameters and trains $8.3\times$ faster than the FNO. Overall, PI-NCA demonstrates that conservation can be incorporated into compact NCAs as an architectural property rather than a training objective, yielding competitive accuracy with substantially lower parameter and computational costs.

1 INTRODUCTION

A learned emulator is useful only if it continues to produce plausible states when its own errors are fed back as input. For physical systems, plausibility has a sharper meaning: quantities conserved by the governing equation should remain conserved. Most neural surrogates do not guarantee this. Global neural operators such as the FNO (Li et al., 2021a) learn accurate short-horizon maps, but nothing in their update prevents total mass or momentum from drifting. On every conservative phenomenon we measured, their end-of-rollout mass error is of order one or larger.

Neural cellular automata (Mordvintsev et al., 2020) offer a promising starting point. Their update is a single local rule applied identically at every cell, which is exactly the structure of an explicit finite-difference stencil, and a useful rule fits in a few thousand parameters. Recent work has shown that NCAs can be trained as autoregressive PDE surrogates (Richardson et al., 2024; Saha and Wang, 2026). The standard NCA update, however, adds an unconstrained increment $f_\theta(s_t)$ to the state. That increment is free to create or destroy any quantity, with visible consequences in our benchmark: the standard NCA's heat error grows $22.9\times$ over the evaluation horizon, its total heat drifts by 65 units, and on Navier–Stokes it scores below a do-nothing identity emulator at ten seeds.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

PI-NCA. We change *what the rule predicts*. A PI-NCA cell emits a flux along each spatial axis, and the state is updated by the periodic discrete divergence of that flux, as in a finite-volume scheme (LeVeque, 2002). Every unit of flux leaving one cell enters its neighbour, so the divergence sums to zero over the grid and the total cannot change. This property holds at initialization, throughout training, and after training for any weights, while the rule remains as small and local as the NCA it replaces.

Multi-field systems and MC-PI-NCA. Physical systems usually carry several conserved fields that interact, such as fluid height and the two momentum components of the shallow-water equations. A single conserved sum is the wrong constraint because it allows one field to gain what another loses. The multi-field construction perceives all fields jointly but predicts a separate flux pair for each, so every field's total is conserved independently while the fields remain coupled through the shared perception. MC-PI-NCA uses a wider perception network for this setting.

Both architectures can also respect a hard state range, such as the order parameter of a phase-field model. Clipping to the range breaks conservation, and the obvious uniform correction breaks the range again. We restore mass in proportion to each cell's remaining headroom. This correction is exact, cannot push a cell outside the range, and adds no parameters.

Results in brief. We compare the two architectures with seven learned baselines and an identity floor under a common training protocol. We use paired tests on shared held-out initial conditions with Holm–Bonferroni correction and calibrate every result against the reference solver's own discretization error.

- **Coupled conservation laws.** On shallow water, PI-NCA ($5{,}280$ parameters) and MC-PI-NCA ($10{,}992$) are the two most accurate of eleven models. Both are significantly more accurate than the FNO ($592{,}995$ parameters), and MC-PI-NCA is also significantly more accurate than the U-Net.
- **Stiff bounded dynamics.** On Cahn–Hilliard, PI-NCA with the bound projection is the most accurate model across ten seeds, outperforming the FNO on all 80 initial conditions. The projection alone lowers PI-NCA's error by 30%.
- **Accuracy per parameter.** On heat, MC-PI-NCA is second only to an FNO 60 times larger; on advection–diffusion, PI-NCA is second only to an FNO 130 times larger. MC-PI-NCA significantly outperforms the ResNet on heat, while PI-NCA significantly outperforms both the ResNet and the U-Net on advection–diffusion.
- **Over the standard NCA.** PI-NCA is more accurate on seven of ten phenomena with one-third fewer parameters and loses only on Nagumo and FitzHugh–Nagumo, whose reaction terms do not conserve their fields.
- **Cost and consistency.** On heat, PI-NCA trains $8.3\times$ faster than the FNO, and its mass error is four orders of magnitude lower.

Contributions.

1. **PI-NCA**, a neural cellular automaton whose shared local rule predicts fluxes and guarantees exact conservation for all parameters (Proposition 1).
2. **A multi-field extension** with independent per-field conservation (Proposition 2), instantiated as compact PI-NCA or the wider MC-PI-NCA, together with a parameter-free bound projection that keeps mass exact under a hard state range (Proposition 3).
3. **A controlled evaluation** on ten 2-D phenomena with matched-capacity ablations of the flux head and the projection and an explicit account of the regimes where the conservation prior is the wrong one.

2 Related Work

Neural cellular automata for dynamics. NCAs learn one homogeneous local update applied at every cell (Mordvintsev et al., 2020). Richardson et al. (2024) train NCAs on PDE trajectories,

show generalization beyond the training data, and constrain them to respect symmetries. Saha and Wang (2026) use an NCA surrogate for PDE time-stepping and compare it with PDE-Net, a PINN, and an FNO on five canonical equations. Together, these studies establish that NCAs are viable autoregressive surrogates. Our focus is different: we make a conservation law a structural property of the rule itself and ask what that buys.

Conservation by construction. Finite-volume bookkeeping makes conservation a property of the update rather than of the loss (LeVeque, 2002). FINN brings this structure into learned models by composing modules that represent inter-cell fluxes, so each quantity obeys its own balance law (Praditia et al., 2021; Karlbauer et al., 2022). Neural Conservation Laws construct networks that satisfy the continuity equation exactly (Richter-Powell et al., 2022). Other approaches enforce constraints by projection or correction after prediction (Hansen et al., 2023; Liu et al., 2024). We build directly on the flux-form idea and do not claim it as new. PI-NCA and MC-PI-NCA place the structure in one shared 3×3 circular perception followed by a purely pointwise MLP, with identical weights at every cell and the same rule used unchanged across all ten equations; FINN instead composes per-quantity flux modules. Conservation is enforced per field for coupled systems and remains exact under hard state bounds through a projection with a feasibility guarantee. Our contribution is its compact NCA realization, per-field extension, mass-preserving bound projection, and evaluation. Ablation A4 changes only the output head of a matched local backbone (0.7% parameter difference).

Learned discretizations and graph solvers. PDE-Net constrains convolution filters to approximate differential operators (Long et al., 2018), and data-driven discretizations learn stencil coefficients (Bar-Sinai et al., 2019). Message-passing and mesh-based simulators generalize across geometries and resolutions (Brandstetter et al., 2022; Pfaff et al., 2021; Sanchez-Gonzalez et al., 2020), but require graph construction that a periodic regular grid does not need and can exhibit known under-reaching failure modes (Tesán et al., 2026). Our domains are periodic and regular, so these methods are discussed rather than benchmarked.

Neural operators and physics-informed learning. The FNO learns a global operator through truncated Fourier mixing (Li et al., 2021a; Kovachki et al., 2023); DeepONet learns maps between function spaces (Lu et al., 2021); PINO and physics-informed DeepONet add residual terms (Li et al., 2021b; Wang et al., 2021). PINNs fit one initial-value problem per run (Raissi et al., 2019) and have documented optimization and spectral-bias pathologies (Krishnapriyan et al., 2021; Wang et al., 2024; Rahaman et al., 2019). We use the FNO as the principal global baseline. Autoregressive rollout error is a distinct failure mode from one-step accuracy (Lippe et al., 2023; Um et al., 2020), which is why every model here is trained through the rollout. Weak baselines and reporting bias are recognized problems in scientific machine learning benchmarks (McGreivy and Hakim, 2024; Takamoto et al., 2022; Koehler et al., 2024), which is why we report a trivial floor, physics-free controls, and the solver's own error next to every result.

3 METHOD

Let $s_t \in \mathbb{R}^{H \times W \times C}$ be the discrete state on a 2-D periodic grid, with one channel per physical field. An emulator g_θ approximates the one-step map of a numerical reference solver $\mathcal{S}_{\Delta t}$ and is applied recursively.

3.1 PI-NCA: SCALAR STATE INCREMENTS TO FLUXES

We first define PI-NCA for a single field ($C = 1$); Section 3.2 extends the construction to coupled multi-field states. A standard NCA perceives each cell's neighbourhood with a learned circular convolution P_θ, processes the percept with a per-cell MLP, and adds the result to the state:

$$s_{t+1} = s_t + f_\theta(s_t), \qquad f_\theta(s_t) = \mathrm{MLP}_\theta\big(\mathrm{ReLU}(P_\theta(s_t))\big). \tag{1}$$

PI-NCA keeps the perception and the MLP but reinterprets the output. The rule emits a flux $F_\theta(s_t) = (F_1, F_2)$, with $F_a \in \mathbb{R}^{H \times W}$ for each spatial axis, and advances the state by its periodic backward-difference divergence:

$$s_{t+1} = s_t + D_h F_\theta(s_t), \qquad D_h F = \sum_{a=1}^{2} \big(\mathrm{roll}_a(F_a) - F_a\big), \tag{2}$$

where roll_a shifts a field by one cell along axis a with wrap-around. F_a at a cell is the amount transferred across that cell's face on axis a; the same amount is subtracted on one side and added on the other. The network is a 3×3 circular perception convolution with 32 channels, a per-cell MLP with 64 and 32 channels, and a zero-initialized flux head, totaling $4{,}576$ parameters for a single field. Because the head starts at zero, PI-NCA starts as the identity map.

The flux output and fixed divergence guarantee conservation; the NCA supplies the compact, homogeneous, parameter-shared local backbone in which the flux is learned.

Proposition 1 (Exact conservation)**.** *For every parameter value θ and every state s_t, the update of Equation 2 satisfies $\sum_{\text{cells}} s_{t+1} = \sum_{\text{cells}} s_t$.*

The proof is one line: roll_a permutes the cells, so $\sum \mathrm{roll}_a(F_a) = \sum F_a$ and each term of $D_h F$ sums to zero (Appendix A). Two consequences matter in practice. Conservation is not something the optimizer has to discover or maintain, so it cannot be traded against accuracy during training or degrade over a long rollout. Moreover, the hypothesis class is smaller: among all local updates, PI-NCA can represent only those that redistribute. That restriction is the source of its advantage on conservative dynamics and of its failure on reaction systems, and Section 5.4 measures both. In floating point the invariant holds to round-off, about 10^{-7} relative over a rollout on our grids. The construction extends to three dimensions by adding the corresponding third-axis term to the sum.

3.2 Multi-field extension and MC-PI-NCA

For a system with $C > 1$ fields, we extend the scalar construction by predicting a flux pair for each field, giving $2C$ output channels in 2-D, and applying the divergence channel by channel:

$$s_{t+1}^{(c)} = s_t^{(c)} + D_h F_\theta^{(c)}(s_t), \qquad c = 1, \ldots, C. \tag{3}$$

Every $F^{(c)}$ is computed from the full multi-field neighbourhood, so the fields remain coupled through perception while their totals are held separately. MC-PI-NCA is sized for this joint perception, with 48 perception and 96 hidden channels.

Proposition 2 (Per-field conservation)**.** *Under Equation 3, $\sum_{\text{cells}} s_{t+1}^{(c)} = \sum_{\text{cells}} s_t^{(c)}$ for each field c separately, even when every flux $F_\theta^{(c)}$ depends on all C input fields.*

For $C = 1$, Proposition 2 reduces exactly to Proposition 1; its new content for $C > 1$ is the simultaneous preservation of C independent totals despite cross-field coupling in the learned fluxes. This differs from conserving one lumped total. A lumped constraint is satisfied by an update that adds mass to one field and removes the same amount from another, which violates the physics of any system whose fields are distinct conserved quantities; in the shallow-water equations it would let fluid height be converted into momentum. The multi-field construction is the natural prior for source-free conservation laws written in conserved variables. In our experiments, PI-NCA denotes its compact-width instantiation and MC-PI-NCA its wider instantiation on multi-field problems; both use Equation 3. On single-field problems, both reduce to Equation 2 and differ only in width, so comparisons between them isolate width under the same flux-form update rather than a different conservation law.

3.3 Bound projection: hard ranges without losing mass

Phase-field models such as Cahn–Hilliard evolve an order parameter confined to a range $[\ell, h]$. Emulators that leave the range can diverge within a few steps, and clipping is an effective stabilizer, but clipping destroys conservation. With the projection enabled, PI-NCA clips the candidate state and restores the recorded mass $M_t = \sum s_t$:

$$\widetilde{s} = \mathrm{clip}\big(s_t + D_h F_\theta(s_t), \ell, h\big), \qquad s_{t+1} = \mathcal{R}_{M_t}(\widetilde{s}). \tag{4}$$

The natural restoration, a uniform offset $(M_t - \sum \widetilde{s})/N$ added to all N cells, moves cells that the clip has just placed on a bound back outside it. We instead distribute the deficit $d = M_t - \sum_j \widetilde{s}_j$ in proportion to each cell's headroom, $r_i = h - \widetilde{s}_i$ when mass must be added and $r_i = \widetilde{s}_i - \ell$ when it must be removed:

$$\mathcal{R}_{M_t}(\widetilde{s})_i = \widetilde{s}_i + \mathrm{sign}(d)\ \min\Big(\tfrac{|d|}{C_r}, 1\Big)\, r_i, \qquad C_r = \textstyle\sum_j r_j. \tag{5}$$

Proposition 3 (Bounded and mass-exact). *If $\widetilde{s} \in [\ell, h]^N$ and $M_t \in [N\ell, Nh]$, then $\mathcal{R}_{M_t}(\widetilde{s}) \in [\ell, h]^N$ and $\sum \mathcal{R}_{M_t}(\widetilde{s}) = M_t$. If M_t lies outside that interval, the output stays in $[\ell, h]^N$ and saturates at the nearest bound.*

The projection is a single vectorized pass with no parameters and is differentiable almost everywhere, so the model can be trained end to end through it. It is an architectural configuration rather than a different model, intended only for fields that have a physical range; the range is measured from a short solver rollout rather than fixed by hand. We refer to PI-NCA with the projection enabled as *bounded PI-NCA*. Unlike the unbounded flux update, the projection is not strictly local: per field and per step it requires two scalar global reductions, for the deficit d and headroom total C_r. These reductions require $O(N)$ work and logarithmic reduction depth, asymptotically less communication than the FNO's all-to-all FFT, but a distributed implementation still needs one synchronization stage per step. Unbounded PI-NCA and MC-PI-NCA remain strictly local.

Training. Every model, proposed or baseline, minimizes the same rollout loss

$$\mathcal{L}_K(\theta) = \frac{1}{K} \sum_{\tau=1}^{K} \mathrm{MSE}(g_\theta^{\circ\tau}(s_0), \mathcal{S}_{\Delta t}^{\circ\tau}(s_0)) \,, \tag{6}$$

with gradients propagated through the model's own trajectory. No term of the loss refers to conservation; PI-NCA and MC-PI-NCA already guarantee it.

4 Experimental Setup

Phenomena. We consider ten 2-D periodic problems spanning the structural cases that the prior should distinguish: conservative scalar transport (heat, advection–diffusion), a conservative multi-field law (shallow water), a stiff fourth-order bounded equation (Cahn–Hilliard), a bounded reaction–diffusion equation (Allen–Cahn), a dispersive system (wave), globally coupled flow (Navier–Stokes in vorticity form), and three reaction systems whose fields are created and destroyed (Nagumo, FitzHugh–Nagumo, Gray–Scott). Equations and solver settings are in Appendix B.

Models. The proposed architectures are PI-NCA (plain or with its bounded configuration) and MC-PI-NCA. The baselines are the FNO (Li et al., 2021a) at its standard size and at the NCA parameter budget (FNO-S); a residual CNN (ResNet) and a U-Net, each at full size and at a matched budget of $5{,}364$–$7{,}464$ parameters (ResNet-S, U-Net-S); the standard NCA of Equation 1; and an identity emulator $g(x) = x$ as a floor, which makes "learned worse than nothing" a reportable outcome. All learned models have zero-initialized output heads and start as the identity. The reported comparison contains eleven models in total.

Protocol. All 2-D results use one protocol: a 48^2 grid, $K = 12$ training steps, 48 evaluation steps, and 1,200 AdamW gradient steps with a learning rate of 10^{-3} and weight decay 10^{-5}. A 30-epoch linear warmup starts at 2×10^{-4}; weights use He/Kaiming-normal initialization and output heads are zero-initialized. The safety factor for the divergence guard is 1.25, and training initial conditions receive ten solver pre-seeding steps, except on Cahn–Hilliard, where pre-seeding is disabled because the states coarsen too quickly for pre-evolved states to match the evaluation distribution. Each epoch splits a fresh PRNG key and samples a new batch of initial conditions, so 1,200 epochs means 1,200 gradient steps. With a batch size of 32, this covers 38,400 freshly sampled initial conditions, with no dataset reuse or train–test overlap: this is an infinite-data regime. Full-size ResNet uses batch 16 because the batch-32 configuration exceeds available device memory, so it sees 19,200 freshly sampled conditions; all other models use batch 32. The effect of this batch-size deviation on ResNet accuracy is unknown. No per-architecture hyperparameter tuning was performed, which may underoptimize the FNO and U-Net. Every model is trained with five seeds on every phenomenon and with ten seeds on heat, Cahn–Hilliard and Navier–Stokes. A single-seed 3-D suite is reported in Appendix H.

Statistics. Every seed is evaluated on eight held-out initial conditions from the distinct PRNG stream `seed + 10000`. All models see these conditions in the same order, so comparisons are

Table 1: PI-NCA against the standard NCA it is built from, on every phenomenon (five seeds, $n = 40$ paired evaluations each, Holm–Bonferroni across the ten tests). PI-NCA uses $4{,}576$ parameters on single-field problems against the standard NCA's $6{,}784$. "Win" is the fraction of held-out initial conditions on which PI-NCA has lower error than NCA. The two losses occur on reaction systems that create and destroy their fields, where an exact conservation law is the wrong assumption. The flux-divergence rule is better on every conservative target.

Phenomenon	PI-NCA	NCA	Ratio	Win	Verdict
Heat	**0.00684**	0.0542	7.9×	1.00	PI-NCA better
Advection–diff.	**0.00150**	0.00600	4.0×	1.00	PI-NCA better
Shallow water	**0.00709**	0.0164	2.3×	1.00	PI-NCA better
Gray–Scott	**0.388**	0.595	1.5×	0.62	PI-NCA better
Wave	**0.0135**	0.0204	1.5×	1.00	PI-NCA better
Navier–Stokes	**0.390**	0.492	1.3×	0.90	PI-NCA better
Cahn–Hilliard	**0.541**	0.636	1.2×	0.80	PI-NCA better
Allen–Cahn	0.0473	0.0465	1.0×	0.57	tie
Nagumo	0.379	**0.0149**	0.04×	0.00	NCA better
FitzHugh–Nagumo	0.992	**0.0529**	0.05×	0.00	NCA better

paired. The primary metric is per-initial-condition relative L^2 error at the end of the rollout, summarized with a 10,000-sample percentile bootstrap. A difference is reported as significant only when the bootstrap interval of the paired differences excludes zero and a Wilcoxon signed-rank test rejects after Holm–Bonferroni correction over the comparison family. We also record per-field mass error, error growth over the horizon, spectral error distribution, parameter count, training time, and inference latency (Appendix D).

Calibration. Emulators are trained and scored against the solver, so we also measure the solver's own error against closed-form solutions for the linear problems and by timestep refinement otherwise (Table 6). Differences below this error scale measure agreement with a particular discretization rather than physical accuracy.

Compute. All results come from a single unattended 22.2-hour run on one consumer GPU, preceded by a correctness gate in which 204 tests passed and one was skipped (205 collected). Each results file records the software version, device, and full configuration.

5 Results

5.1 PI-NCA against the NCA it replaces

The most direct test of the idea holds everything fixed except what the rule predicts. Table 1 compares PI-NCA with the standard NCA on all ten phenomena.

PI-NCA is significantly more accurate on seven phenomena, by factors of 7.9 on heat, 4.0 on advection–diffusion, and 2.3 on shallow water, while using $4{,}576$ parameters compared with the NCA's $6{,}784$. The improvement is largest where the dynamics are conservative and smooth, and it persists on problems with more complex structure: Cahn–Hilliard, wave, Gray–Scott, and Navier–Stokes. The flux form also changes how errors evolve. On heat, the NCA's error grows $22.9\times$ from the first quarter of the horizon to the end, compared with $5.5\times$ for PI-NCA (Table 7). On Navier–Stokes at ten seeds, the NCA is less accurate than the identity floor (0.528 against 0.513), while PI-NCA is more accurate (0.441, $p = 3.1 \times 10^{-14}$ for PI-NCA versus the identity). In these experiments, restricting the rule to redistribution reduces error growth relative to the unconstrained NCA; it does not establish general long-horizon stability.

The two losses are informative rather than incidental. Nagumo and FitzHugh–Nagumo have reaction terms that create and destroy their fields, and a rule that can only move mass cannot represent them: on Nagumo, PI-NCA stays at the identity floor, and on FitzHugh–Nagumo its error is 0.99. Section 5.4 shows the same effect under a fully matched ablation.

Table 2: All ten 2-D phenomena at the main operating point (five seeds); entries are mean per-initial-condition relative L^2 error. The proposed configuration is selected based on PDE structure. *Identity* is $g(x) = x$, and *teacher* is the reference solver's discretization error (Table 6).

Phenomenon	Structure	Best overall	rel-L^2	Selected proposed	rel-L^2	Identity	Teacher
Shallow water	3-field, conservative	PI-NCA	**0.00709**	MC-PI-NCA	0.00742	0.0518	9.7×10^{-7}
Cahn–Hilliard	4th-order, stiff, bounded	PI-NCA (bounded)	**0.369**	PI-NCA (bounded)	0.369	0.918	0.0441
Heat	conservative, smooth	FNO	0.00157	PI-NCA	0.00684	0.191	0.00114
Advection–diff.	conservative, transported	FNO	0.00123	PI-NCA	0.00150	0.228	0.00472
Allen–Cahn	bounded, sharp interfaces	FNO	0.00889	PI-NCA	0.0473	0.0549	0.00017
Wave	2-field, hyperbolic	U-Net	0.0129	PI-NCA	0.0135	0.0507	0.00061
Navier–Stokes	globally coupled, turbulent	FNO	0.0730	PI-NCA	0.390	0.450	0.0251
Gray–Scott	2-field reaction–diffusion	U-Net	0.138	PI-NCA	0.388	0.345	0.00497
Nagumo	bistable, non-conservative	U-Net	0.00751	PI-NCA	0.379	0.380	0.00282
FitzHugh–Nagumo	2-field excitable, non-conservative	NCA	0.0529	PI-NCA	0.992	1.135	0.00601

Table 3: Paired comparisons supporting the main accuracy claims. Wilcoxon tests are Holm–Bonferroni corrected across the nine rows; all remain significant. "Win" is the fraction of initial conditions with lower proposed-model error. Heat and Cahn–Hilliard use $n = 80$ paired evaluations and the other phenomena use $n = 40$.

Phenomenon	Proposed model	Params	rel-L^2		Baseline	Params	rel-L^2	Win
Shallow water	PI-NCA	5,280	**0.00709**	vs	FNO	592,995	0.00869	0.85
	MC-PI-NCA	10,992	**0.00742**	vs	FNO	592,995	0.00869	0.95
	MC-PI-NCA	10,992	**0.00742**	vs	U-Net	265,584	0.00898	0.90
Cahn–Hilliard	PI-NCA (bounded)	4,576	**0.368**	vs	ResNet	74,336	0.465	0.91
	PI-NCA (bounded)	4,576	**0.368**	vs	FNO	592,897	0.650	1.00
Heat	MC-PI-NCA	9,936	**0.00377**	vs	ResNet	74,336	0.00607	0.82
	MC-PI-NCA	9,936	**0.00377**	vs	ResNet-S	5,364	0.00556	0.72
Advection–diff.	PI-NCA	4,576	**0.00150**	vs	U-Net	265,104	0.00349	1.00
	PI-NCA	4,576	**0.00150**	vs	ResNet	74,336	0.00461	1.00

5.2 Where PI-NCA and MC-PI-NCA lead

Table 2 places the better-performing proposed model among all eleven models on each phenomenon, and Table 3 gives the paired tests behind the main claims.

Coupled conservation laws. Shallow water is the problem the construction is built for: three conserved fields, no sources, and coupling between them. PI-NCA is the most accurate model at 0.00709 with $5{,}280$ parameters, and MC-PI-NCA is second at 0.00742, statistically indistinguishable from it ($p = 0.63$). The FNO, with $592{,}995$ parameters, is third at 0.00869. PI-NCA has lower error than the FNO on 85% of initial conditions, while MC-PI-NCA does so on 95%; MC-PI-NCA also beats the U-Net on 90%. Both proposed models hold each field's mass to about 10^{-4} in absolute terms over the rollout, against 2.1 for the FNO and 3.0 for the U-Net.

Stiff, bounded dynamics. Cahn–Hilliard is fourth order and stiff, and its order parameter is confined to a range. With the bound projection, PI-NCA is the most accurate model across ten seeds (0.368). The strongest baseline is the full-size ResNet at 0.465, which bounded PI-NCA beats on 91% of initial conditions with a parameter count $16\times$ smaller; the FNO is ninth of eleven at 0.650, and bounded PI-NCA beats it on all 80 initial conditions. The solver's own error on this problem is 0.044, so these differences are well above the calibration level, and the independent five-seed matrix has the same leader and runner-up. Without the projection, PI-NCA (0.528) has a lower mean error than the U-Net and the FNO but trails the ResNet; the projection is what turns a competitive model into the best one (Section 5.4). Because teacher pre-seeding is disabled for this equation, the evaluation covers the spinodal and early-coarsening transient rather than the slow Ostwald-ripening regime.

Accuracy per parameter. On heat and advection–diffusion the FNO is the most accurate model, and a proposed model ranks second on both. On heat, MC-PI-NCA reaches 0.00377 with $9{,}936$ parameters, a sixtieth of the FNO's; it is significantly better than the full-size ResNet and the matched

Table 4: Controlled ablations, grid 48^2. Each block changes exactly one thing. *A4* swaps the conservative flux head for an unconstrained residual head at a matched backbone (4,576 against 4,544 parameters, five seeds), on one conservative and one non-conservative equation. *A1* enables PI-NCA's bound projection on an otherwise identical 4,576-parameter model, which adds no parameters (ten seeds on Cahn–Hilliard, five elsewhere). Mass drift is the absolute change in each field's total at the end of the rollout.

Ablation	Setting	rel-L^2	Mass drift	Reading
A4 — flux head versus residual head, matched backbone				
	heat, flux head	**0.00743**	7.0×10^{-5}	conservation helps by $4.6\times$
	heat, residual head	0.0339	14.3	where the quantity is conserved
	Nagumo, flux head	0.379	1.8×10^{-5}	and hurts by $24\times$
	Nagumo, residual head	**0.0161**	707	where it is not
A1 — PI-NCA with and without the bound projection				
	CH, projection off	0.528	38.3	30% lower error and
	CH, projection on	**0.368**	5.5×10^{-5}	nearly six orders less drift
	Allen–Cahn, off	**0.04733**	7.0×10^{-6}	a small but significant
	Allen–Cahn, on	0.04747	8.4×10^{-6}	cost (0.3%) on Allen–Cahn
	Shallow water, off	**0.00709**	8.5×10^{-5}	harmful where the dynamics
	Shallow water, on	0.0159	5.4×10^{-4}	have no hard state bound

ResNet-S, and statistically indistinguishable from a U-Net 27 times larger. On advection–diffusion, PI-NCA reaches 0.00150 with $4{,}576$ parameters, a parameter count $130\times$ smaller than the FNO's, and is $2.3\times$ more accurate than the U-Net and $3.1\times$ more accurate than the ResNet. Both problems are close to the limit of what distillation from this solver can resolve: the FNO's heat error is within 30% of the solver's own error, and on advection–diffusion PI-NCA's gap to the FNO, 2.7×10^{-4}, is eighteen times smaller than the solver's error of 0.0047. Within that limit, the proposed models deliver comparable accuracy with roughly two orders of magnitude fewer parameters. Table 9 in the appendix groups the heat results by parameter budget; MC-PI-NCA is the most accurate model in the small budget.

5.3 Exact conservation in practice

Proposition 1 is a statement about exact arithmetic; the practical question is what it buys in a trained model over a long rollout. On heat, the absolute mass error at the end of the 48-step evaluation is 8.2×10^{-5} for PI-NCA and 6.1×10^{-5} for MC-PI-NCA, compared with 2.04 for the FNO, 2.74 for the U-Net, and 65.3 for the standard NCA (Table 7). The same separation holds on advection–diffusion, wave, and shallow water: both proposed models stay below 10^{-3}, and every learned baseline, including the parameter-matched ones, is above 1. For a conservative system this is not a secondary metric. A surrogate whose total drifts produces physically inconsistent states even when its pointwise error is small, and the drift compounds with horizon. An unbounded PI-NCA loses mass only when an external clamp intervenes, as in the Cahn–Hilliard case discussed under A1.

5.4 Why it works: controlled ablations

The comparisons above differ in more than one component. Table 4 isolates the two design choices, changing exactly one element at a time.

A4: which component does the work? Replacing a free residual head with the PI-NCA flux head, at a matched backbone and within 1% of the parameter count, improves heat by $4.6\times$ (0.00743 against 0.0339). On Nagumo, the same substitution makes the error $24\times$ worse (0.379 against 0.0161). The mass column explains both. The flux head holds the total below 10^{-4} on both equations. On heat, that is the correct physics; on Nagumo, whose reaction term changes the total by design, it is precisely the wrong constraint. The prior is powerful because it is strong, and it belongs where the target has the invariant. Thus, within this matched backbone, the flux output and divergence cause the conservation and accuracy change; the NCA provides the compact shared local rule. A4 does not establish the same accuracy gain for global or multi-resolution backbones.

A1: the bound projection. On Cahn–Hilliard, enabling the projection on an otherwise identical $4{,}576$-parameter PI-NCA lowers error from 0.528 to 0.368, yields lower error on all 80 paired initial conditions, and reduces mass drift from 38.3 to 5.5×10^{-5}. The drift of the unprojected model does not contradict Proposition 1: every model in the evaluation passes through a divergence guard that clamps states to 1.25 times the solver's physical range, and on Cahn–Hilliard the unprojected rollouts reach that clamp, which removes mass the flux update had conserved. The projection makes bounding and conservation compatible. Its degradation elsewhere is expected. On Allen–Cahn, the error rises by 0.3%, a small but significant increase; on shallow water, whose dynamics have no hard bound, enforcing a measured range more than doubles the error. The projection should therefore be enabled only for fields with a genuine physical range. Appendix Figure 4 visualizes A4 and the Cahn–Hilliard half of A1.

5.5 Where the conservation prior does not fit

Three regimes fall outside the design. **Non-conservative reactions**: on Nagumo, FitzHugh–Nagumo, and Gray–Scott, the proposed models are far behind. They are worse than the identity floor on Gray–Scott and level with it on Nagumo; A4 attributes this directly to the conservation constraint. **Global coupling**: on Navier–Stokes, the FNO and the U-Net lead, and the best proposed model ranks sixth, although it remains above the floor, whereas the standard NCA does not. **Sharp interfaces with non-conserved dynamics**: on Allen–Cahn, the FNO leads and MC-PI-NCA ranks fifth. On wave, PI-NCA ranks sixth, though within 5% of the best error.

The ranking is stable where the proposed models lead. We varied grid size, training horizon, and training budget around the main operating point for five representative models (Table 10). On Cahn–Hilliard, PI-NCA is the best of the five in all nine settings, even without the projection; on heat and Navier–Stokes, the FNO is best at every setting. Orderings below the winner do change, most strongly on Navier–Stokes along the training horizon ($\tau = 0$), so every ranking is quoted with its operating point.

6 Discussion and Limitations

PI-NCA and MC-PI-NCA turn a physical invariant into an architectural fact. The results suggest concise practical guidance. If the target conserves its fields, a flux-predicting rule is a better default than a free increment and is competitive with, or better than, models two orders of magnitude larger. If the fields are coupled, conserve them separately. If the state has a hard range, enable the headroom projection; if it does not, leave it off. If the dynamics create or destroy the quantity, do not use the constraint.

The evaluation is limited to periodic regular grids and the reported rollout horizon and resolution; the planned out-of-distribution, guard-free stability, and resolution-transfer studies produced no usable results at this scale. The fixed 3×3 stencil also limits global reach; wider or dilated perception (Yu and Koltun, 2016) could expand the receptive field but is not evaluated here, and the architecture does not control spectral error. The 3-D study is single-seed and lacks convolutional controls. Flux-head variants of the FNO and U-Net and external FINN and Neural Conservation Laws baselines are not reported. Appendix F gives the complete limitations and missing-axis account.

7 Conclusion

PI-NCA predicts fluxes and therefore conserves exactly for any weights; its multi-field extension applies the guarantee to each field of a coupled system, with MC-PI-NCA providing a wider instantiation. The architectures lead on shallow water and bounded Cahn–Hilliard, are competitive on heat and advection–diffusion, and improve on the standard NCA on seven of ten phenomena. Ablations show that the conservation structure helps when the invariant is correct and hurts when it is not. Cellular automata therefore provide a compact setting in which conservation can be an update property rather than a training objective.

AI USE STATEMENT

Generative AI tools were used to draft and revise this manuscript, develop and check parts of the benchmarking and analysis code, identify and verify literature, and assist with the interpretation of results. The authors reviewed all AI-assisted work and tested the AI-assisted code through the correctness and regression checks described in Appendix I. Experimental measurements were produced by executing the reviewed code; no numerical result in this paper was written by hand or estimated. The authors take responsibility for the final text, claims, references, code, and reported results.

ETHICS STATEMENT

This work concerns numerical PDE benchmarks and involves no human subjects. The main foreseeable risk is treating a neural surrogate as a validated simulator. We therefore distinguish solver emulation from continuum accuracy, report the solver's own error and cost alongside every model, and state which evaluation axes were not measured. Use in safety-critical simulation would require application-specific validation beyond this study.

REPRODUCIBILITY STATEMENT

The update rules, projection, and training objective are specified in Section 3, with proofs in Appendix A and layer-level details in Appendix C. Appendix I records software versions, the correctness gate, and reproduction procedure. Every result comes from a single unattended run on one GPU; each results file records its backend, devices, and exact configuration. Rankings of the reported models and the paired tests in Tables 3, 1, and 4, which compare pairs not tabulated by the run itself, are recomputed from the saved per-initial-condition errors by the anonymized supplementary analysis.

REFERENCES

Yohai Bar-Sinai, Stephan Hoyer, Jason Hickey, and Michael P. Brenner. Learning data-driven discretizations for partial differential equations. *Proceedings of the National Academy of Sciences*, 116(31):15344–15349, 2019. doi: 10.1073/pnas.1814058116.

Johannes Brandstetter, Daniel Worrall, and Max Welling. Message passing neural PDE solvers. *International Conference on Learning Representations*, 2022. arXiv:2202.03376.

Derek Hansen, Danielle C. Maddix, Shima Alizadeh, Gaurav Gupta, and Michael W. Mahoney. Learning physical models that can respect conservation laws. *Proceedings of the International Conference on Machine Learning*, PMLR 202:12469–12510, 2023. arXiv:2302.11002.

Matthias Karlbauer, Timothy Praditia, Sebastian Otte, Sergey Oladyshkin, Wolfgang Nowak, and Martin V. Butz. Composing partial differential equations with physics-aware neural networks. *International Conference on Machine Learning*, 2022. arXiv:2111.11798.

Felix Koehler, Simon Niedermayr, Rüdiger Westermann, and Nils Thuerey. APEBench: A benchmark for autoregressive neural emulators of PDEs. *Advances in Neural Information Processing Systems*, 37, 2024. arXiv:2411.00180.

Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces. *Journal of Machine Learning Research*, 24(89):1–97, 2023. arXiv:2108.08481.

Aditi S. Krishnapriyan, Amir Gholami, Shandian Zhe, Robert M. Kirby, and Michael W. Mahoney. Characterizing possible failure modes in physics-informed neural networks. *Advances in Neural Information Processing Systems*, 34, 2021. arXiv:2109.01050.

Randall J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2002. doi: 10.1017/CBO9780511791253.

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

Zongyi Li, Hongkai Zheng, Nikola Kovachki, David Jin, Haoxuan Chen, Burigede Liu, Kamyar Azizzadenesheli, and Anima Anandkumar. Physics-informed neural operator for learning partial differential equations. *arXiv preprint*, 2021. arXiv:2111.03794.

Phillip Lippe, Bastiaan S. Veeling, Paris Perdikaris, Richard E. Turner, and Johannes Brandstetter. PDE-Refiner: Achieving accurate long rollouts with neural PDE solvers. *Advances in Neural Information Processing Systems*, 36, 2023. arXiv:2308.05732.

Ning Liu, Yiming Fan, Xianyi Zeng, Milan Klöwer, Lu Zhang, and Yue Yu. Harnessing the power of neural operators with automatically encoded conservation laws. *Proceedings of the International Conference on Machine Learning*, PMLR 235:30965–30997, 2024. arXiv:2312.11176.

Zichao Long, Yiping Lu, Xianzhong Ma, and Bin Dong. PDE-Net: Learning PDEs from data. *Proceedings of the International Conference on Machine Learning*, PMLR 80:3208–3216, 2018. arXiv:1710.09668.

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*, 3:218–229, 2021. doi: 10.1038/s42256-021-00302-5.

Nick McGreivy and Ammar Hakim. Weak baselines and reporting biases lead to overoptimism in machine learning for fluid-related partial differential equations. *Nature Machine Intelligence*, 6:1256–1269, 2024. doi: 10.1038/s42256-024-00897-5.

Alexander Mordvintsev, Ettore Randazzo, Eyvind Niklasson, and Michael Levin. Growing neural cellular automata. *Distill*, 2020. doi: 10.23915/distill.00023.

Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter W. Battaglia. Learning mesh-based simulation with graph networks. *International Conference on Learning Representations*, 2021. arXiv:2010.03409.

Timothy Praditia, Matthias Karlbauer, Sebastian Otte, Sergey Oladyshkin, Martin V. Butz, and Wolfgang Nowak. Finite volume neural network: Modeling subsurface contaminant transport. *arXiv preprint*, 2021. arXiv:2104.06010.

Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. *Proceedings of the International Conference on Machine Learning*, PMLR 97:5301–5310, 2019. `https://proceedings.mlr.press/v97/rahaman19a.html`.

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. doi: 10.1016/j.jcp.2018.10.045.

Alex D. Richardson, Tibor Antal, Richard A. Blythe, and Linus J. Schumacher. Learning spatio-temporal patterns with neural cellular automata. *PLOS Computational Biology*, 20(4):e1011589, 2024. doi: 10.1371/journal.pcbi.1011589.

Jack Richter-Powell, Yaron Lipman, and Ricky T. Q. Chen. Neural conservation laws: A divergence-free perspective. *Advances in Neural Information Processing Systems*, 35, 2022. arXiv:2210.01741.

Esha Saha and Hao Wang. Learning PDE time-stepping with neural cellular automata. *arXiv preprint*, 2026. arXiv:2608.30328.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter W. Battaglia. Learning to simulate complex physics with graph networks. *Proceedings of the International Conference on Machine Learning*, 2020. arXiv:2002.09405.

Makoto Takamoto, Timothy Praditia, Raphael Leiteritz, Dan MacKinlay, Francesco Alesiani, Dirk Pflüger, and Mathias Niepert. PDEBench: An extensive benchmark for scientific machine learning. *Advances in Neural Information Processing Systems*, 35, 2022. arXiv:2210.07182.

Lucas Tesán, Mikel M. Iparraguirre, David González, Pedro Martins, and Elías Cueto. On the under-reaching phenomenon in message-passing neural PDE solvers: Revisiting the CFL condition. *Computer Methods in Applied Mechanics and Engineering*, 449:118476, 2026. doi: 10.1016/j.cma.2025.118476.

Kiwon Um, Robert Brand, Yun (Raymond) Fei, Philipp Holl, and Nils Thuerey. Solver-in-the-Loop: Learning from differentiable physics to interact with iterative PDE-solvers. *Advances in Neural Information Processing Systems*, 33, 2020. arXiv:2007.00016.

Sifan Wang, Hanwen Wang, and Paris Perdikaris. Learning the solution operator of parametric partial differential equations with physics-informed DeepONets. *Science Advances*, 7(40):eabi8605, 2021. doi: 10.1126/sciadv.abi8605.

Sifan Wang, Shyam Sankaran, and Paris Perdikaris. Respecting causality for training physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 421:116813, 2024. doi: 10.1016/j.cma.2024.116813.

Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *International Conference on Learning Representations*, 2016. arXiv:1511.07122.

A PROOFS

Throughout, the grid has N cells with periodic indexing, and roll_a shifts a field by one cell along axis a with wrap-around, so it is a permutation of the cells.

Proposition 1. Summing Equation 2 over all cells,

$$\sum_{\text{cells}} s_{t+1} - \sum_{\text{cells}} s_t = \sum_{a=1}^{d} \Big(\sum_{\text{cells}} \mathrm{roll}_a(F_a) - \sum_{\text{cells}} F_a \Big) = 0, \tag{7}$$

because a sum over all cells is invariant under a permutation of the cells. Nothing in the argument depends on how F_a was computed, so the identity holds for every parameter value, every state, and every perception stencil. In floating-point arithmetic, the identity holds up to round-off, as reflected by the measured relative mass errors of order 10^{-7}.

Proposition 2. Equation 3 applies the scalar update to each channel c with its own flux $F^{(c)}$. Applying the argument above to channel c alone gives the result for that channel; the dependence of $F^{(c)}$ on the other channels plays no role. By contrast, a single lumped constraint $\sum_c \sum_{\text{cells}} s^{(c)}$ admits updates with $\sum_{\text{cells}} \Delta s^{(1)} = -\sum_{\text{cells}} \Delta s^{(2)} \neq 0$, which violate the conservation of each field.

Proposition 3. Take $d = M_t - \sum_j \widetilde{s}_j > 0$; the case $d < 0$ is symmetric with $r_i = \widetilde{s}_i - \ell$, and $d = 0$ leaves the state unchanged. Here $r_i = h - \widetilde{s}_i \geq 0$ because $\widetilde{s}_i \leq h$, and $C_r = \sum_j r_j = Nh - \sum_j \widetilde{s}_j$. Let $\lambda = \min(d/C_r, 1) \in (0, 1]$. Each output is $\widetilde{s}_i + \lambda(h - \widetilde{s}_i)$, a convex combination of $\widetilde{s}_i$ and h, and so lies in $[\widetilde{s}_i, h] \subseteq [\ell, h]$. The total added is λC_r. If $M_t \leq Nh$ then $d \leq C_r$, so $\lambda = d/C_r$ and the total added is exactly d, giving $\sum \mathcal{R}_{M_t}(\widetilde{s}) = M_t$. If $M_t > Nh$ then $\lambda = 1$ and every cell equals h, the nearest admissible state. If $C_r = 0$ every cell is already at h and the update is taken to be the identity. The map is a composition of elementary operations with a single $\min$, so it is differentiable except on the set $|d| = C_r$. For the uniform offset, by contrast, a cell with $\widetilde{s}_i = h$ receives $h + d/N > h$ whenever $d > 0$, so that restoration cannot guarantee the bound.

B PDE DEFINITIONS AND TEACHER SCOPE

The experiments concern the numerical one-step map for each PDE. The governing forms below define the structural distinctions used in the main text; they do not replace the full teacher configuration. Domain sizes, coefficients, solver substeps, initial-condition distributions, and numerical precision must be taken from the corresponding experiment configuration to reproduce a particular score.

Heat.

$$\partial_t u = \alpha \nabla^2 u. \tag{8}$$

With periodic boundaries and constant diffusivity, spatial integration removes the Laplacian and conserves total heat. This is the smooth local-diffusion test.

Advection–diffusion.

$$\partial_t u = -\mathbf{c} \cdot \nabla u + \alpha \nabla^2 u. \tag{9}$$

Constant-velocity periodic transport conserves the integral as well as moving the field. Its FNO result should not be interpreted as evidence that conservation is absent; it shows that a correct invariant alone need not determine the best model.

Allen–Cahn.

$$\partial_t u = \varepsilon^2 \nabla^2 u - W'(u), \qquad W'(u) = u^3 - u. \tag{10}$$

The reaction term generally changes the spatial integral. Evolving interfaces provide a setting in which both non-conservation and spatial detail matter.

Wave.

$$\partial_{tt} u = c^2 \nabla^2 u. \tag{11}$$

The emulator represents this as a first-order system in $(u, \partial_t u)$. The initial velocity channel is identically zero, so $\partial_t u \equiv 0$ at $t = 0$ by construction for every evaluated trajectory. Propagation and coupling between fields distinguish it from scalar diffusion.

Shallow-water. The source-free periodic shallow-water system is represented in conserved variables (h, hu, hv). Per-field conservation concerns height and momentum, not arbitrary primitive velocity channels. This variable choice is part of the architectural prior.

Nagumo and FitzHugh–Nagumo. The scalar bistable Nagumo equation is

$$\partial_t u = D \nabla^2 u + u(1 - u)(u - a). \tag{12}$$

FitzHugh–Nagumo is the two-field activator–inhibitor reaction system; its coefficients are those of the benchmark solver configuration. For both problems the reaction terms change each field's total, so neither conserves a field.

Cahn–Hilliard.

$$\partial_t u = \nabla^2 \big(-\varepsilon^2 \nabla^2 u + W'(u)\big), \qquad W'(u) = u^3 - u. \tag{13}$$

This fourth-order equation conserves mass on a periodic domain. The solver clips its state to $[-1, 1]$, so the distillation target is that clipped map rather than the unconstrained continuum equation. Bounded PI-NCA conserves mass, not the Cahn–Hilliard free energy, and no discrete energy law is claimed for it.

Gray–Scott.

$$\partial_t u = D_u \nabla^2 u - uv^2 + F(1 - u), \tag{14}$$

$$\partial_t v = D_v \nabla^2 v + uv^2 - (F + k)v. \tag{15}$$

The feed and removal terms change field totals. The original configuration at $\Delta t = 2.0$ exceeds the explicit diffusion limit and diverges; the benchmark uses $\Delta t = 1.0$. Errors remain large for every model.

Navier–Stokes. The vorticity formulation reconstructs the flow through the Poisson relation $\nabla^2 \psi = -\omega$, which is the source of global coupling. Viscosity, forcing, the treatment of the zero Fourier mode, and the time discretization are those of the benchmark solver configuration.

Three-dimensional suite. Heat, advection–diffusion, Allen–Cahn, Nagumo, Gray–Scott, and FHN are implemented in NDHWC layout and evaluated on 24^3 grids in this run. The implementation uses a 7-point Laplacian, flux divergence along all three axes, and spectral convolution through `rfftn`; the explicit stability bound tightens to $\Delta t \leq \Delta x^2/(6D)$ in three dimensions, which is why Gray–Scott uses $\Delta t = 0.5$ here against 1.0 in 2-D. The 3-D correctness subset contains 11 checks. It does not include 3-D versions of the wave, shallow-water, Navier–Stokes, or Cahn–Hilliard problems.

C Architecture Details and Schematics

The model configurations used in the paper are as follows. **PI-NCA** uses a 3×3 circular perception convolution with 32 channels, a per-cell MLP with 64 and 32 channels, and a zero-initialized flux head with $2C$ outputs, followed by the per-field divergence. **Bounded PI-NCA** is the same network with the clip and headroom projection appended; the projection has no parameters, so both configurations have 4,576 parameters on a single field. **MC-PI-NCA** is the per-field construction with 48 perception and 96 hidden channels. On a single field, PI-NCA and MC-PI-NCA differ only in width.

NCA is the standard residual NCA of Equation 1, with 48 perception and 64 hidden channels. **ResNet** is a residual CNN with circular padding (width 32, depth 4) and **U-Net** a two-level U-Net

Table 5: Structural properties of the two proposed architectures and the baselines. Perception extent is the reach of a single update, not the full receptive field after the divergence. Parameter counts are for the single-field heat configuration; on C-field problems the flux head grows by $2C$ output channels only. "Exact" conservation holds for any weights, at initialization and after training, up to floating-point round-off.

Property	NCA	PI-NCA	MC-PI-NCA	ResNet / U-Net	FNO
Perception extent	± 1 cell	± 1 cell	± 1 cell	local / pooled	global
Mass conservation	none	**exact**	**exact, per field**	none	none
Bound projection	no	**optional, 0 params**	optional	no	no
Params (heat)	6,784	**4,576**	9,936	74k / 265k	5.9×10^5
Shared local rule	yes	yes	yes	no	no
Boundary handling	periodic	periodic	periodic	periodic (circ.)	periodic (FFT)

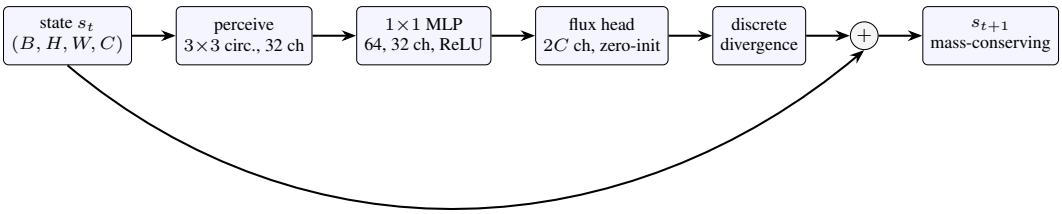


Figure 1: PI-NCA. The head predicts a flux per axis; its periodic discrete divergence sums to zero over the grid, so the residual update conserves mass by construction.

with mean-pool downsampling (width 24); **ResNet-S** (width 12, depth 2) and **U-Net-S** (width 4) match the NCA parameter budget. **FNO** has width 24, 8 modes and depth 4, and **FNO-S** width 8, 4 modes and depth 2. Every learned model has a zero-initialized output head and therefore starts as the identity map. For a fixed width and stencil, the cost of a local convolutional update scales linearly with the number of cells; parameter count alone is not a runtime model.

FNO. A spectral layer combines a pointwise transformation with learned Fourier weights:

$$v_{\ell+1} = \sigma\big(W v_\ell + \mathcal{F}^{-1}\big(R_\ell \odot \mathcal{F}[v_\ell]\big)\big)\,. \tag{16}$$

The implementation truncates spectral mixing to the lowest $m \times m$ modes in 2-D and uses the FNO as a residual one-step emulator, so it is trained and evaluated through the same interface as every other model.

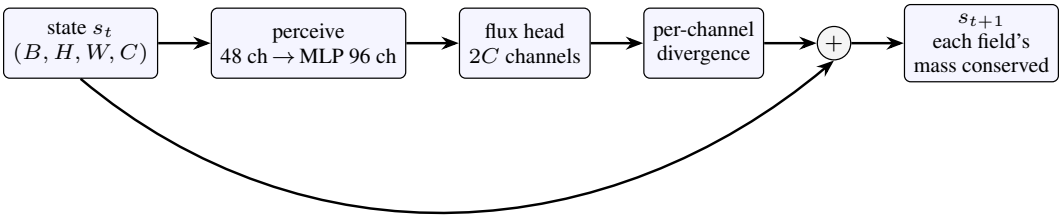


Figure 2: MC-PI-NCA. Perception sees all fields jointly; the flux head emits a flux pair (f_x, f_y) per field, and a per-channel divergence conserves each field's total independently.

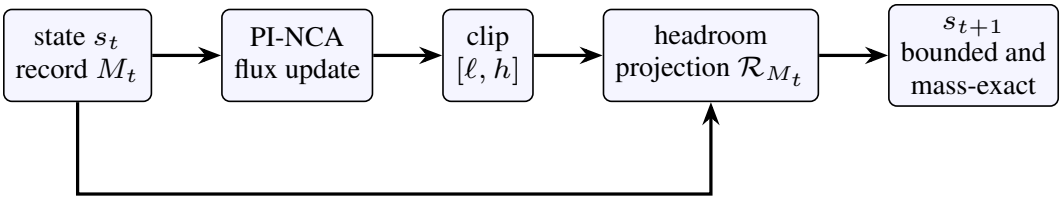


Figure 3: PI-NCA with the bound projection. The clip enforces the range; the headroom projection of Equation 5 restores the recorded mass without moving any cell outside it (Proposition 3). The projection has no parameters.

Table 6: What the distillation target itself gets wrong, at the benchmark's own grid (48^2) and horizon. For the linear constant-coefficient members the periodic problem is diagonal in Fourier space, so the continuum solution is available in closed form and the error splits into the stencil's *spatial* truncation and forward Euler's *temporal* error; for the rest the timestep is refined and the Cauchy differences bound the teacher's own error, with the observed order of accuracy reported. Every teacher converges. The last two columns report the reference solver's and fastest emulator's wall-clock times per step.

Phenomenon	Estimate	Teacher rel-L^2	Spatial	Temporal	Order	Solver (ms)	Emulator (ms)
Heat	closed form	0.00114	0.00181	0.00070	—	1.64	3.86
Wave	closed form	0.00061	0.00048	0.00082	—	3.79	4.23
Advection–diff.	closed form	0.00472	0.00480	0.00103	—	3.27	4.61
Allen–Cahn	Δt refinement	0.00017	—	—	0.97	3.63	6.44
Gray–Scott	Δt refinement	0.00497	—	—	1.03	1.70	1.26
Shallow water	Δt refinement	9.7×10^{-7}	—	—	n/a[†]	4.92	4.72
Cahn–Hilliard	Δt refinement	0.0441	—	—	0.94	3.08	6.28
FitzHugh–Nagumo	Δt refinement	0.00601	—	—	1.00	4.06	6.78
Nagumo	Δt refinement	0.00282	—	—	1.00	3.81	6.36
Navier–Stokes	Δt refinement	0.0251	—	—	1.54	1.15	1.28

[†]Shallow water's Cauchy differences reach machine precision, so floating-point cancellation prevents a meaningful convergence-rate estimate.

D Training, Initialization, and Measurement

Each training trajectory starts with a randomly drawn initial condition and a teacher pre-seeding rollout of ten steps, except on Cahn–Hilliard where pre-seeding is disabled. The teacher and emulator then evolve from the resulting common state over $K = 12$ steps. The optimizer receives the gradient of trajectory MSE through a `lax.scan` rollout, with all epochs inside a single jitted scan and donated buffers. Evaluation uses a distinct PRNG stream for held-out initial conditions and a four-times longer horizon.

One protocol, recorded per file. Every 2-D number in this paper comes from one protocol: grid 48^2, 12 training steps, 48 evaluation steps, 1,200 gradient steps, AdamW with learning rate 10^{-3} and weight decay 10^{-5}, He/Kaiming-normal initialization before ReLU, zero-initialized update heads, a 30-epoch linear warmup from 2×10^{-4}, divergence-guard safety factor 1.25, and ten pre-seeding steps. Full-size ResNet uses batch 16; all other models use batch 32. Each epoch samples a fresh batch from a newly split PRNG key, giving 38,400 freshly sampled training initial conditions for batch-32 models and 19,200 for ResNet, with no reuse. Evaluation uses eight held-out conditions per seed from the distinct stream `seed + 10000`. The full matrix uses five seeds $\{0, \ldots, 4\}$ and the three regime representatives use ten $\{0, \ldots, 9\}$. Each results file records the JAX version, backend, device list, and the complete configuration dataclass, so a table cannot be assembled from runs that differ in any of these without the difference being visible.

Cahn–Hilliard is the one exception: teacher pre-seeding is disabled because its solutions coarsen rapidly from fresh initial conditions, so developed states mismatch the evaluation distribution. Models within the CH comparison share the setting, so the comparison remains internally controlled. This setting evaluates the spinodal and early-coarsening transient, not the slow Ostwald-ripening regime.

Teacher timestep corrections. Two reference solvers required stability corrections that are recorded rather than silently applied. Gray–Scott ships at $\Delta t = 2.0$, above the explicit-diffusion bound $\Delta x^2/(4D_u) = 1.25$; experiments use $\Delta t = 1.0$. Cahn–Hilliard is fourth order, so its explicit bound is $\Delta t \leq 2/(|\lambda_{\text{lap}}| + \varepsilon^2\lambda_{\text{lap}}^2) = 0.231$, and the original configuration used $\Delta t = 0.5$. It avoided visible blow-up only because the stepper clips the state each step; removing the clip at that timestep causes the state to reach NaN, and timestep refinement gives an observed order of accuracy of zero. Experiments therefore use $\Delta t = 0.02$, where the scheme converges at the expected first order (Table 6). Table 6 therefore reports convergence separately for each equation rather than assuming that the original solver settings are stable.

Table 7: Diagnostic panel on heat (grid 48^2, ten seeds). High-frequency error fraction is the share of squared residual energy in the upper half of the spatial spectrum: it describes *where* the residual sits, not how large it is. Mass drift is the absolute change in total heat at the end of the 48-step rollout. Error growth is the ratio of relative error at the final step to that at one quarter of the horizon.

Model	rel-L^2	PSNR (dB)	HF frac.	Mass drift	Err. growth	Train (s)
FNO	0.001447	71.7	0.036	2.04	2.11	107.6
MC-PI-NCA	0.003766	65.3	0.358	6.1×10^{-5}	5.47	18.2
U-Net	0.004081	62.4	0.146	2.74	3.07	57.6
ResNet-S	0.005559	60.1	0.063	7.83	2.74	20.7
PI-NCA	0.006057	60.4	0.553	8.2×10^{-5}	5.52	12.9
NCA	0.05614	46.1	0.331	65.3	22.87	15.3
Identity (floor)	0.1889	29.3	0.000	0	3.70	—

Reported metric inventory. We report relative L^2 error, MSE, RMSE, MAE, L^∞, PSNR, an SSIM-style structural score, high-frequency error fraction, relative error at $T/4, T/2, 3T/4, T$, error-growth ratio, per-channel mass drift and its drift curve, periodic-boundary residual, gradient energy, parameter count, training wall time, inference time per step, and throughput. The primary metric is per-initial-condition relative L^2, because pairing requires one value per initial condition; the batch-reduced ratio of norms is also recorded and is a different estimator of the same quantity, so the two do not print equal numbers.

Mass drift is $|\mathrm{mass}(\mathrm{prediction}) - \mathrm{mass}(\mathrm{initial\ condition})|$, computed *per channel*: a lumped total would score a model that adds mass to one field and removes it from another as perfectly conserving. For non-conservative targets, a low drift score indicates a wrong constraint rather than fidelity. Gradient energy is a diagnostic proxy, not a proof of nonlinear stability, and a normalized high-frequency error fraction need not decrease when total error decreases.

E Additional Results and Resource Accounting

Heat. Table 8 gives the complete ten-seed ranking. The main text compresses three observations. First, the standard NCA is an order of magnitude behind every other learned model (0.0561 against 0.0074 for the next worst) and its error-growth ratio is $22.9\times$, against about $5.5\times$ for PI-NCA and MC-PI-NCA and 2–3$\times$ for the convolutional and spectral baselines; this shows faster error accumulation over the measured horizon, not a general instability result. Second, MC-PI-NCA is second overall with $9{,}936$ parameters, ahead of a U-Net with $265{,}104$. Third, at $n = 80$, every difference against the FNO is Holm-significant.

Training and inference cost. On heat, PI-NCA trains in 12.9 s and MC-PI-NCA in 18.2 s, compared with 107.6 s for the FNO. Their recorded inference latencies are 1.22, 1.32, and 4.17 ms per step, respectively, but the identity emulator measures 3.00 ms on heat and 1.68 ms on shallow water under the same procedure. These short-call latency values are therefore dispatch-dominated and descriptive rather than evidence of a speedup. Every model emulates the same solver step at the same Δt and rolls out the same 48 steps. The local architectures' arithmetic scales linearly with the number of cells and involves no Fourier transform. At 48^2, the reference solver costs 1.2–4.9 ms per step (Table 6).

Resource accounting. Table 9 groups models into compact and full-size regimes; parameter counts within a group are similar but not exactly matched. Among the compact models, MC-PI-NCA (0.00377) leads ResNet-S (0.00556), FNO-S (0.00562) and U-Net-S (0.00738). Across budgets, a $60\times$ increase in parameters from MC-PI-NCA to the FNO buys a factor of 2.6 in error. Inference latency does not track parameter count: the $592{,}897$-parameter FNO costs 4.17 ms per step against 1.22 ms for PI-NCA and 1.32 ms for MC-PI-NCA, while the $265{,}104$-parameter U-Net costs 1.80 ms, because the FFT rather than the weight count dominates. These values are medians of 10 jitted calls after two warmups at batch 8 and grid 48^2. The identity emulator measures 3.00 ms on heat and 1.68 ms on shallow water despite doing no model computation, so the short-call timing

Table 8: Heat, ten seeds $\times$ 8 held-out initial conditions ($n = 80$ paired evaluations per model), grid 48^2, 1,200 epochs. Intervals are 10,000-sample percentile bootstraps on the per-initial-condition errors. Every model is compared with the FNO by a Wilcoxon signed-rank test on the same initial conditions, Holm-corrected; every difference is significant. The solver's own error on this task is 0.00114, so the FNO has nearly saturated the distillation target and the gap to the next rows should not be over-interpreted.

Model	rel-L^2	95% CI	Params	vs. FNO
FNO	0.001447	[0.00137, 0.00154]	592,897	reference
MC-PI-NCA	0.003766	[0.00310, 0.00453]	9,936	worse
U-Net	0.004081	[0.00383, 0.00434]	265,104	worse
ResNet-S	0.005559	[0.00520, 0.00594]	5,364	worse
FNO-S	0.005616	[0.00531, 0.00592]	8,433	worse
PI-NCA (bounded)	0.005720	[0.00505, 0.00644]	4,576	worse
PI-NCA	0.006057	[0.00529, 0.00687]	4,576	worse
ResNet	0.006070	[0.00542, 0.00677]	74,336	worse
U-Net-S	0.007381	[0.00683, 0.00797]	7,464	worse
NCA	0.05614	[0.0405, 0.0725]	6,784	worse
Identity (floor)	0.1889	[0.185, 0.193]	1	worse

Table 9: Heat, grouped by parameter budget rather than ranked across budgets (ten seeds). Within a budget the comparison isolates the architectural prior; across budgets it shows what two orders of magnitude more parameters buy. Inference is per rollout step at batch 8 and grid 48^2 on the evaluation GPU, reported as the median of 10 jitted calls after two warmups. The identity row is 3.00 ms here but 1.68 ms on shallow water despite doing no model computation, showing that these short timings are dispatch-dominated.

Budget	Model	rel-L^2	Params	Infer. (ms/step)
floor	Identity	0.1889	1	3.00
small	MC-PI-NCA	0.003766	9,936	1.32
	ResNet-S	0.005559	5,364	1.27
	FNO-S	0.005616	8,433	3.39
	PI-NCA (bounded)	0.005720	4,576	1.25
	PI-NCA	0.006057	4,576	1.22
	U-Net-S	0.007381	7,464	1.58
	NCA	0.05614	6,784	1.48
medium	ResNet	0.006070	74,336	1.67
large	FNO	0.001447	592,897	4.17
	U-Net	0.004081	265,104	1.80

is dispatch-dominated. Accuracy, size, and latency must be read separately, and none of these ratios establishes acceleration over the solver (McGreivy and Hakim, 2024).

Ablation details. A4 runs at the main protocol with five seeds on a matched backbone, changing only the output head. Its parameter counts differ by 0.7% ($4{,}576$ against $4{,}544$), the cost of a two-channel flux head against a one-channel residual head, which is far too small to explain a $4.6\times$ or $24\times$ effect. A1 compares PI-NCA with bounded PI-NCA from the main matrix: identical networks, identical training, and the same measured bounds, so the only difference is the projection. The Cahn–Hilliard comparison uses the ten-seed runs and the other phenomena the five-seed runs; the paired tests are recomputed from the saved per-initial-condition errors.

The headroom form of the projection is chosen on the strength of Proposition 3. The uniform offset provably fails to preserve the bound (Appendix A), and a constructed state exhibiting the failure is part of the test suite; a separate experiment comparing the two corrections on PI-NCA is not part of this study.

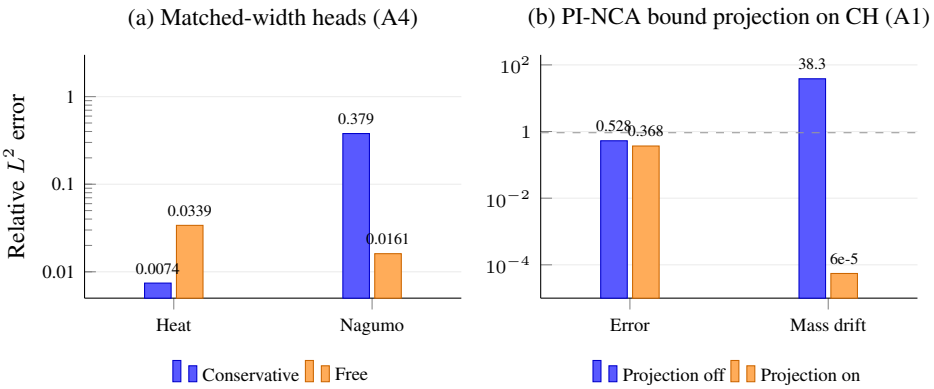


Figure 4: Two interventions on the PI-NCA design, each changing one component. **(a)** The conservative flux head improves heat by $4.6\times$ and worsens Nagumo by $24\times$ against a residual head at matched backbone (five seeds): the prior helps exactly where the quantity is conserved. **(b)** On Cahn–Hilliard, enabling PI-NCA's bound projection lowers error by 30% (identity reference 0.918, dashed) *and* cuts mass drift by nearly six orders of magnitude, with no added parameters (ten seeds). The axes are logarithmic because the effects span several decades. In panel (b), relative error and mass drift are different quantities on one display; only within-quantity changes are meaningful.

Scope of the head attribution. A4 is deliberately a within-backbone intervention. It identifies the flux output and fixed divergence as the source of exact conservation and of the measured change in error, while the shared local NCA trunk determines the representation and computational profile. A stronger architecture-agnostic claim would require otherwise identical FNO and U-Net models with residual versus flux outputs. Likewise, FINN is a valid conservative baseline: its facewise flux modules and PI-NCA's cellwise flux rule impose related finite-volume bookkeeping through different parameterizations. These controls are not included in the common-protocol result matrix, so the paper does not use A4 to rank PI-NCA against other possible flux-carrying backbones.

F Evaluation Axes Not Measured at This Scale

Three planned evaluation axes produced no usable results at the reported scale. We mark them as unmeasured rather than substituting reduced-scale results obtained under a different compute setting.

- **Out-of-distribution generalization.** The harness defines held-out axes before training—amplitude, structure count, length scale, added broadband spectral content, a random torus translation as a negative control, PDE coefficient, and horizon—and evaluates a trained model on each. This evaluation did not complete at the reported scale.
- **Long-horizon stability with the guard disabled.** The accuracy tables clamp rollouts to the teacher's measured physical range so that a diverging model cannot emit a meaningless metric. That guard is appropriate for accuracy but inappropriate for stability, so the stability study disables it and counts failures. This evaluation did not complete at the reported scale.
- **Resolution transfer.** No result under the common compute configuration is available, so results from a different backend are excluded.

Because no stability numbers are available at this scale, the paper makes no long-horizon robustness claim. The one stability-adjacent quantity that *is* measured is the error-growth ratio in Table 7, which is computed inside the guarded accuracy protocol and is a weaker statement.

Other scope limitations. All evaluated domains are periodic and regular, so Proposition 1 does not cover non-periodic boundaries, which require explicit boundary fluxes. On heat and advection–diffusion, leading errors approach the solver's discretization error, so small differences do not measure continuum accuracy. A 3×3 rule moves information by one cell per step, limiting globally coupled flow; wider or dilated perception is compatible with flux form but is not evaluated. The

Table 10: Rank stability. Each axis is swept with everything else fixed at the main operating point (grid 48^2, rollout 12, 1,200 epochs, 48 evaluation steps, one seed), for five models: NCA, PI-NCA, FNO, ResNet-S, and the identity floor. Kendall's τ compares the ordering at the smallest and largest setting ($\tau = 1$: unchanged). The winner is reported at all three settings of each axis. The analysis is restricted to the five prespecified architectures listed above.

Phenomenon	Axis	τ	Winner at all three settings
Heat	grid $24, 32, 48$	0.80	FNO
	rollout $4, 8, 12$	0.80	FNO
	epochs $300, 600, 1200$	0.80	FNO
Cahn–Hilliard	grid $24, 32, 48$	0.80	PI-NCA
	rollout $4, 8, 12$	0.60	PI-NCA
	epochs $300, 600, 1200$	0.60	PI-NCA
Navier–Stokes	grid $24, 32, 48$	1.00	FNO
	rollout $4, 8, 12$	0.00	FNO
	epochs $300, 600, 1200$	0.40	FNO

Table 11: Three-dimensional results (24^3, 600 epochs, **single seed**). The 3-D suite compares PI-NCA (plain and bounded), MC-PI-NCA, the standard NCA, and the FNO; it carries neither the convolutional controls nor the identity floor, and it does not include shallow water or Cahn–Hilliard, so it supports only the weaker statement that the 2-D trends recur. Rank is among these five models.

Phenomenon	Best in 3-D	Best proposed (rank)	3-D rel-L^2	Reading
Heat	FNO	PI-NCA (bounded) (2)	0.0107	second, as in 2-D
Advection–diff.	FNO	PI-NCA (2)	0.00379	second, as in 2-D
Allen–Cahn	FNO	PI-NCA (3)	0.0436	level with NCA
Gray–Scott	NCA	MC-PI-NCA (2)	0.847	reaction system
Nagumo	NCA	PI-NCA (3)	0.197	reaction system
FitzHugh–Nagumo	FNO	MC-PI-NCA (3)	0.981	reaction system

fixed stencil is tied to its training Δx and offers no resolution-invariance guarantee. Conservation constrains the field total rather than its spectrum: on heat, the high-frequency error fractions are 0.553 for PI-NCA and 0.358 for MC-PI-NCA, against 0.036 for the FNO (Table 7). The projection is compared with no alternative correction, and the 3-D suite is single-seed, omits convolutional controls, and excludes shallow water and Cahn–Hilliard. Flux-output variants of the FNO and U-Net are not evaluated at the reported scale; PINO, FINN, and Neural Conservation Laws are also not evaluated as external baselines. Results from the study's own solvers are not directly comparable with published PDEBench or APEBench values.

G Rank Stability Under Scale

Each sweep varies one axis with the others fixed at the main operating point, using five models that span the unconstrained local, conservative local, global spectral, and physics-free families, using one seed. Kendall's τ compares only the endpoints and treats all rank inversions as equal, including inversions between models whose errors are close; the per-setting errors in the anonymized supplementary results distinguish the two cases.

The winners are stable: FNO on heat and Navier–Stokes, and PI-NCA on Cahn–Hilliard, at all three settings of every axis. On Cahn–Hilliard, PI-NCA leads at every setting without the bound projection, which is not part of the sweep. The orderings below the winner move most on Navier–Stokes, where changing the training horizon from 4 to 12 steps reorders the models ($\tau = 0$).

H Three-Dimensional Results

The 3-D suite uses 24^3 grids, 600 epochs, eight-step training and 24-step evaluation horizons, and a single seed. It omits the convolutional baselines and the identity floor. Without that floor, "better than nothing" is not checkable; the suite also excludes shallow water and Cahn–Hilliard.

Read with those caveats, the 3-D results follow the 2-D pattern. The FNO leads heat, advection–diffusion, Allen–Cahn and FitzHugh–Nagumo, and a proposed model is second on heat and advection–diffusion, ahead of the standard NCA by $5\times$ and $3\times$. On the reaction systems, the standard NCA leads Nagumo and Gray–Scott, and on Nagumo PI-NCA and MC-PI-NCA sit at 0.197 against the standard NCA's 0.0273, the same sign as the 2-D A4 ablation. Neither 3-D Navier–Stokes nor 3-D Cahn–Hilliard is tested.

I Reproducibility Details

Every reported result was produced under the common configuration on one consumer GPU. The anonymized supplementary archive records the software versions, full configuration, and per-stage timings; planned evaluations without usable outputs are listed in Appendix F and are not used in the paper.

The run begins with a correctness gate in which 204 tests passed and one was skipped (205 collected) before any benchmark executes. The gate asserts that the JAX ports match the verbatim PyTorch references within tolerance. Regression tests cover the periodic Laplacian axes for the training array layout and the Cahn–Hilliard timestep stability condition described in Appendix D. It also asserts properties on which the statistical claims rest—that the bootstrap interval covers at its nominal rate, that Holm correction is strictly more conservative than uncorrected testing, that the closed-form solver references are correct, and that the headroom projection is simultaneously mass-exact, bound-preserving, and differentiable.

The rankings and paired tests quoted in the tables are recomputed from the saved per-initial-condition errors by an analysis script included in the anonymized supplementary archive. A bibliography check resolves every cited arXiv identifier against the arXiv API.