LEARNING PHYSICS FROM AN IMPERFECT ANCESTOR

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural operators evaluate parametric partial differential equations (PDEs) cheaply but degrade sharply outside their training distribution. Physics-informed neural networks (PINNs) avoid dependence on labeled data, yet their optimization can be basin-fragile: when the governing residual admits multiple solutions, a PINN trained from scratch may converge to a physically incorrect state despite achieving a small residual. We show that these failure modes can be addressed jointly: an imperfect NO provides the structural prior needed to place a PINN in the correct solution basin, while the PDE residual refines the solution beyond the operator's accuracy. We introduce a three-stage framework that freezes the spatial basis of a physics-informed NO, extrapolates its solution branch to an out-of-distribution parameter using a polynomial continuation prior, and distills the resulting field into a fresh PINN. The NO (teacher) need not be accurate at the target; it transfers solution-branch information, while PDE residual minimization in the PINN (student) governs convergence. We evaluate the framework on three nonlinear PDEs: 1D viscous Burgers, 2D steady Allen–Cahn near a pitchfork bifurcation, and 2D steady lid-driven cavity flow. For Allen–Cahn, where the trivial solution satisfies the PDE residual exactly, a standard PINN consistently collapses to the trivial zero branch, whereas distillation from the crude extrapolated operator recovers the non-trivial branch that matches the finite-difference reference. For the lid-driven cavity, extrapolating to a Reynolds number of Re = 3200 accelerates convergence to the correct physical state, achieving competitive accuracy using fewer parameters and optimization steps than recent literature baselines. These results establish a simple principle: an NO need not accurately predict the solution to be useful; it only needs to identify the correct basin from which PINN optimization can recover it.

1 INTRODUCTION

Parametric partial differential equations (PDEs) underpin many scientific and engineering workflows, where the same governing equations must be solved repeatedly across varying parameters. Neural operators (NOs) amortize this cost by learning parameter-to-solution maps, enabling inexpensive inference after training. However, their accuracy can deteriorate sharply outside the training distribution, while adapting them to new parameters typically requires additional simulations or labeled data. Physics-informed neural networks (PINNs) provide a complementary alternative by enforcing the governing equations directly, but their optimization is sensitive to initialization and can converge to physically incorrect solutions when the PDE residual admits multiple minima. In such cases, a small residual does not guarantee selection of the physically relevant solution branch. These failure modes suggest a complementary division of roles: an NO can provide structural information about the solution manifold, while a PINN can enforce the governing physics at the target parameter. We therefore ask whether an imperfect, out-of-distribution NO can guide a PINN without constraining its final accuracy. We show that it can. The key is to use the NO not as a predictor of the final solution, but as a prior for solution-basin selection.

To this end, we introduce a three-stage framework: a physics-informed NO is trained and its spatial basis frozen; the branch is extrapolated to an out-of-distribution target using a polynomial continuation prior, without any reference solution; and the extrapolated field is distilled into a freshly initialized PINN, whose subsequent residual minimization governs final accuracy. The resulting NO acts as a teacher, but unlike conventional knowledge distillation, it is not required to provide an

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.

accurate prediction. Its role is to transfer solution-branch information and place the PINN in a favorable basin of attraction; subsequent minimization of the PDE residual allows the PINN to refine the field and, in principle, surpass the accuracy of the teacher. This separation of roles is central to the proposed framework.

We evaluate the framework on three nonlinear PDEs. First, the 1D viscous Burgers equation establishes the baseline on a problem with a single solution branch. Second, the 2D steady Allen–Cahn equation probes basin selection near a pitchfork bifurcation, a scenario where the trivial solution $u \equiv 0$ satisfies the residual exactly. Third, to test substantial out-of-distribution capabilities, we apply the framework to a 2D steady lid-driven cavity, extending an operator trained on a Reynolds number range of $\mathrm{Re} \in [100, 800]$ to $\mathrm{Re} = 3200$. Finally, an ablation study isolates the underlying mechanism by removing the extrapolation stage, designed to determine whether the critical challenge in these failure modes is strictly residual optimization or physical basin selection.

2 Related work

Operator learning. NOs learn maps between function spaces and are, in principle, discretization-invariant (Kovachki et al., 2023). DeepONet (Lu et al., 2021a; Haghighat et al., 2025) realizes this through a trunk–branch decomposition; the Fourier neural operator (Li et al., 2020a) and the earlier graph-kernel network (Li et al., 2020b) operate in spectral and graph domains respectively, with Wavelet (Tripura & Chakraborty, 2023),Laplace (Cao et al., 2024a) and other variants targeting localized features (Lu et al., 2022; Cao et al., 2024b); no single architecture dominates across problem classes. Physics-informed DeepONets (Wang et al., 2021c; Mandl et al., 2025) replace data losses with residual losses but retain the fixed-parameter-box limitation, and transfer-learning approaches under conditional shift (Weiss et al., 2016; Goswami et al., 2022) extend a trained operator to new domains only with labeled target data at the shifted parameters. Our Part 2 reuses a frozen basis at an extrapolated parameter without any labeled data and without retraining the trunk.

PINNs and their failure modes. Physics-informed neural networks (Raissi et al., 2019) (alongside related residual-minimization schemes such as the deep Galerkin method (Sirignano & Spiliopoulos, 2018), and surveyed broadly in (Karniadakis et al., 2021)) solve PDEs without labeled data but are notoriously hard to optimize. A substantial literature documents training pathologies in such coordinate-based and physics-informed networks: spectral bias (Tancik et al., 2020; Wang et al., 2021b), unbalanced loss gradients (Wang et al., 2021a), a neural-tangent-kernel account of why training stalls (Wang et al., 2022), causality violations in time (Wang et al., 2024b), and instability at depth (Wang et al., 2024a). Remedies span loss re-weighting and self-adaptive weights (Wang et al., 2021a; 2022; McClenny & Braga-Neto, 2023), residual-based adaptive sampling (Wang et al., 2026; Wu et al., 2023), adaptive activations (Jagtap et al., 2020), gradient-enhanced objectives (Yu et al., 2022), architectural changes (Wang et al., 2024a), and consolidated training recipes (Wang et al., 2023a). Less studied are failures of basin selection: on problems whose residuals admit multiple physically distinct minima (bifurcating branches, symmetry-related states, trivial constant solutions) a from-scratch PINN can converge to a spurious solution branch that does not match the reference field, even when the residual is near zero. While temporal and adaptive sampling remedies address erroneous local minima in specific PDEs like the Allen–Cahn equation (Wang et al., 2024b; 2026), our results address broader failures of basin selection that the other remedies above do not target.

Solving the lid-driven cavity. The lid-driven cavity is the canonical incompressible-flow benchmark: multigrid finite-difference tables (Ghia et al., 1982), spectral benchmarks (Botella & Peyret, 1998), high-Reynolds fine-grid studies (Erturk et al., 2005), and an 8192×8192 Richardson-extrapolated solution (Marchi et al., 2021) remain the standard yardstick. PINNs struggle far more: rising Reynolds number sharpens the corner singularities and destabilizes residual minimization. Wang et al. (2023b) show vanilla PINNs at $\mathrm{Re} = 2000$–5000 admit two residual minimizers (one matching DNS, one unphysical) restored to uniqueness via an added entropy-viscosity term. State-of-the-art single-case accuracy comes from the residual-adaptive PirateNet architecture (Wang et al., 2024a), our point of comparison at $\mathrm{Re} = 3200$. These works solve one cavity at a time; we instead extrapolate a frozen operator basis, letting a crude teacher supply the physical basin.

Knowledge distillation. Traditionally, knowledge distillation originates in model compression (Hinton et al., 2015), where a smaller student network is trained to match a larger, highly accurate teacher. In this classical setup, the student's performance is inherently capped by a near-perfect teacher. Our approach differs fundamentally: the teacher is a neural operator providing only a crude out-of-distribution estimate, and the student is explicitly expected to surpass it through subsequent physics-based optimization. Accordingly, the distillation weight decays and ultimately vanishes rather than remaining active throughout training. This shares intuition with scientific computing approaches that use learned models as initial guesses, including warm-start strategies for PINNs (Wang et al., 2024a) and iterative solvers (Eshaghi et al., 2026). However, in contrast to methods that either apply the prior only at initialization (Eshaghi et al., 2026) or rigidly freeze the prior representations (Desai et al., 2021; Goswami et al., 2020; Chakraborty, 2021), our framework retains the teacher as a persistent but decaying constraint. This provides necessary guidance while the student explores the solution landscape, progressively relaxing the imperfect teacher's influence before transferring full control to the governing physics.

3 METHODOLOGY

3.1 SHARED ARCHITECTURE

Let $s(\boldsymbol{x};\boldsymbol{p})$ denote the unknown solution field on domain $\Omega(\boldsymbol{p})$ for parameters $\boldsymbol{p} \in \mathbb{R}^{d_p}$. We employ strong boundary enforcement by defining the ansatz as:

$$s(\boldsymbol{x};\boldsymbol{p}) \;=\; g(\boldsymbol{x};\boldsymbol{p}) \;+\; \mathcal{M}(\boldsymbol{x}) \sum_{k=1}^{K} b_k(\boldsymbol{p})\, \varphi_k(\boldsymbol{x}), \tag{1}$$

where $\{\varphi_k\}_{k=1}^{K}$ is a spatial basis produced by a trunk network, K is the number of trunk basis functions (equivalently, the number of branch outputs), and $b_k(\boldsymbol{p})$ are the branch coefficients, collected into the vector $\boldsymbol{b}(\boldsymbol{p}) = (b_1(\boldsymbol{p}), \ldots, b_K(\boldsymbol{p}))$. Here, $g(\boldsymbol{x};\boldsymbol{p})$ is a known function satisfying the boundary conditions (the boundary lift), and $\mathcal{M}(\boldsymbol{x})$ is a distance multiplier that evaluates to zero on the boundary $\partial\Omega$. Enforcing boundary data exactly via this construction eliminates the need for boundary loss terms and stabilizes training (Lu et al., 2021b).

For the Allen–Cahn problem, the lift g also incorporates a non-zero initial profile. Because the trivial solution ($s = 0$) satisfies the Allen–Cahn PDE exactly, this addition is required to prevent the network from collapsing into the non-physical zero branch. Consequently, no boundary loss term enters training, and the objective relies strictly on the mean-squared interior PDE residual, $\mathcal{R}$ (the problem-specific residual operator, given explicitly for each of the three PDEs in Appendix A):

$$\mathcal{L}_{\mathrm{PDE}}(\boldsymbol{p}) \;=\; \Big\| \mathcal{R}\big[s(\cdot;\boldsymbol{p})\big] \Big\|^2_{L^2(\Omega(\boldsymbol{p}))}, \tag{2}$$

averaged over parameter batches drawn from a bounded base box $\mathcal{B} \subset \mathbb{R}^{d_p}$. Throughout, we report a relative-L^2 accuracy against a reference field s_{ref} (analytical or finite-difference, per problem):

$$\mathrm{Acc} \;=\; 1 - \varepsilon_{L^2}, \qquad \varepsilon_{L^2} \;=\; \frac{\|\hat{s} - s_{\mathrm{ref}}\|_{L^2}}{\|s_{\mathrm{ref}}\|_{L^2}}, \tag{3}$$

where $\hat{s}$ is the predicted field. Table 2 reports ε_{L^2} (labeled error) for the operator and both PINNs, alongside the absolute performance gain (labeled Improvement), defined as $\varepsilon_{L^2,\mathrm{baseline}} - \varepsilon_{L^2,\mathrm{distilled}}$. Everywhere else in the paper, performance is reported in terms of Acc (labeled accuracy). All three problems in §4 instantiate the ansatz equation 1 and train against the residual objective equation 2; they differ in three architectural aspects summarized in Table 1: (i) the procedure used to construct the anchor coefficients after freezing the spatial basis; (ii) the order to which the multiplier $\mathcal{M}$ vanishes on the boundary; and (iii) the spatial bandwidth of the trunk features. Additional implementation-level differences, including residual discretization and a problem-specific regularizer, are specified in Appendix A.6.

3.2 PART 1: BASE OPERATOR

Trunk and branch are trained jointly by minimizing equation 2 over $\boldsymbol{p} \sim \mathrm{Unif}(\mathcal{B})$. Upon convergence, the trunk is frozen, rendering $\{\varphi_k\}$ a fixed function space. An anchor set $\{c_i\}$ is then

Table 1: Problem-specific design choices across the evaluated PDEs.

	Burgers	Allen–Cahn	Cavity
Field / operator	scalar u; $uu_x - \nu u_{xx} = 0$	scalar u; $\nabla^2 u + \lambda(u - u^3) = 0$	stream ψ; vorticity transport
Parameters $\boldsymbol{p}$	(ν, L)	(λ, H)	(Re, H)
Order / dim.	2nd-order, 1D	2nd-order, 2D	4th-order, 2D
Design choice (i): Part 1 anchors	self-supplied from branch	data-free Gauss–Newton on frozen basis	FD reference reprojected onto basis (8 solves)
Design choice (ii): boundary lift	1st-order multiplier	seed lift ℓ_0 + 1st-order mult.	2nd-order multiplier + lid lift
Design choice (iii): trunk bandwidth	moderate, isotropic $\sigma = 6$	mild, isotropic (σ=[3, 3])	moderate, anisotropic $\sigma = [\sigma_\xi, \sigma_\eta]$
Part 2 and 3 target	larger L	λ below base box (near onset)	larger Re (a geometry variant is reported in Appendix E)

constructed at a small collection of base points $\{p_i\} \subset B$. These anchors act as trusted reference coefficients within the training domain, which will later be interpolated to form a structural prior for out-of-distribution extrapolation in Part 2. The three problems differ only in how these anchors are formed, exhibiting a spectrum of increasing reliance on external information (Table 1, Design choice i). Burgers stores the trained branch output directly. Allen–Cahn refines it through a damped Gauss–Newton solve of the residual on the frozen basis, utilizing only the residual and its exact Jacobian (with no reference field). The cavity computes eight finite-difference reference fields at $\mathrm{Re} \in \{100, 200, \ldots, 800\}$ with fixed H=1, reprojecting each onto the frozen basis via ridge least squares in the velocity metric. These eight solves represent the only external data the main cavity pipeline ever consumes.[1]

3.3 Part 2: extrapolation beyond the base box

While parameter continuation is common in PINN curricula to avoid failure modes (Krishnapriyan et al., 2021), unguided extrapolations often drift into non-physical configurations that spuriously minimize the residual. To prevent this, the base anchors are interpolated by a low-degree polynomial $\boldsymbol{c}_{\mathrm{cont}}(\boldsymbol{p})$ fit independently for each of the K modes, acting as a continuation prior. To reach a target $\boldsymbol{p}^\star \in \mathbb{R}^{d_p} \setminus \mathcal{B}$, the branch is retrained at $\boldsymbol{p}^\star$ under two regularizing mechanisms that require no reference solution, keeping the extrapolation entirely data-free.

Bounded correction. At the target, the branch network is re-optimized to output a raw coefficient correction $\Delta\boldsymbol{b}(\boldsymbol{p})$ (analogous to $\boldsymbol{b}(\boldsymbol{p})$ in equation 1, but produced by a fresh Part-2 branch head rather than the Part-1 branch). Rather than using $\Delta\boldsymbol{b}(\boldsymbol{p})$ directly as the coefficients, it is added to the continuation prior as a bounded perturbation,

$$\boldsymbol{b}_{\mathrm{comp}}(\boldsymbol{p}) \;=\; \boldsymbol{c}_{\mathrm{cont}}(\boldsymbol{p}) \;+\; \alpha \,\tanh\big(\Delta\boldsymbol{b}(\boldsymbol{p})\big), \tag{4}$$

with α representing a hard cap on the departure from the prior. If the prior is correct, $\Delta\boldsymbol{b} = 0$ is optimal.

Gram-weighted continuation penalty. Training minimizes

$$\mathcal{L}_{\mathrm{total}} \;=\; \mathcal{L}_{\mathrm{PDE}}(\boldsymbol{b}_{\mathrm{comp}}) \;+\; \beta \left\|\boldsymbol{b}_{\mathrm{comp}} - \boldsymbol{c}_{\mathrm{cont}}\right\|^2_{\boldsymbol{G}(H)}, \tag{5}$$

where $\boldsymbol{G}(H)$ is the field-metric Gram matrix of the frozen basis, assembled on the physical geometry at aspect ratio H via the isoparametric map of Appendix A (for the 1D and fixed-geometry problems, $\boldsymbol{G}(H)$ reduces to a constant matrix $\boldsymbol{G}$). Penalizing coefficient deviations in the field metric (rather than the raw coefficient space) ensures the objective strictly penalizes modifications that alter the

[1] The aspect-ratio extrapolation reported separately in Appendix E trains an independent base operator over the full (Re, H) box with 12 anchors on $\{100, 200, 300, 400\} \times \{1.0, 1.5, 2.0\}$; see Table 7 for both configurations.

physical field. The initial scalar weight β is calibrated once per target so that the penalty term contributes a fixed, predetermined fraction of the total initial objective loss. From that calibrated value, β is either held constant for the remainder of the step or annealed exponentially toward a small non-zero floor over the target's training budget, letting the PDE residual increasingly govern convergence as training proceeds; the problem-specific choices are given in Appendix C.

The fundamental extrapolation operation projects the base anchors to a new target parameter. If the target $\boldsymbol{p}^\star$ is sufficiently close to the training domain, this is executed in a single shot: the prior is fit once on the Part 1 anchors, and the branch is retrained directly at $\boldsymbol{p}^\star$ using equation 4 and equation 5. However, when a single step spans too large a parameter distance, the extrapolation employs an incremental self-supply strategy. In this multi-stage approach, the model extrapolates to intermediate targets; upon convergence, these intermediate solutions are appended to the anchor pool, and the prior is refitted before advancing to the next stage. For example, the principal cavity Re-extrapolation utilizes a four-stage continuation (with a 1D prior over $1/\mathrm{Re}$), the geometry variant fits a 2D prior over $(1/\mathrm{Re}, H)$ (Appendix A), and Allen–Cahn requires a two-stage sequence (Appendix A.6).

The goal of this stage is not high target accuracy, but to generate a coefficient field sufficient to initialize Part 3, which then refines the solution. The Allen–Cahn problem illustrates this: near the pitchfork bifurcation, the true branch amplitude scales as $\sqrt{\lambda - \lambda_1}$ (with λ_1 the critical rate), a form the analytic polynomial prior cannot represent. As a result, the extrapolated field degrades predictably at the near-onset target.

3.4 Part 3: Guarded Distillation into a Fresh Solver

At the target $\boldsymbol{p}^\star$, the composited field from equation 4 is passed to a fresh MLP-PINN u_ϕ trained from scratch on the physics residual. Crucially, this student network abandons the trunk-branch decomposition and the specialized boundary lift utilized in Part 1. The training loss over optimization step t is defined as:

$$\mathcal{L}(\phi; t) \;=\; \big\|\mathcal{R}[u_\phi]\big\|^2_{L^2(\Omega)} + \lambda_{\mathrm{BC}}\mathcal{L}_{\mathrm{BC}} + w(t)\big\|u_\phi - u_{\mathrm{teacher}}\big\|^2_{\Gamma(\boldsymbol{x})}, \tag{6}$$

where $\mathcal{L}_{\mathrm{BC}}$ penalizes boundary-condition violations, u_{teacher} is the Part 2 prediction, $w(t)$ is the distillation schedule, and $\Gamma(\boldsymbol{x})$ is an optional spatial reliability gate (Appendix B). To demonstrate modularity, we employ two strategies. Allen–Cahn and the primary Re-extrapolated cavity use an ungated ($\Gamma \equiv 1$) piecewise-linear $w(t)$ that vanishes during the final training phase. Conversely, the Burgers and geometry-extrapolated cavity configurations utilize a physics-adaptive gate: $w(t)$ tracks the student's residual, while $\Gamma(\boldsymbol{x})$ locally attenuates the teacher's influence at high-residual points to tolerate localized teacher inaccuracies. Both schedules strictly vanish before training concludes, leaving the PDE and boundary constraints to govern final convergence. Figure 1 summarizes the complete three-stage framework.

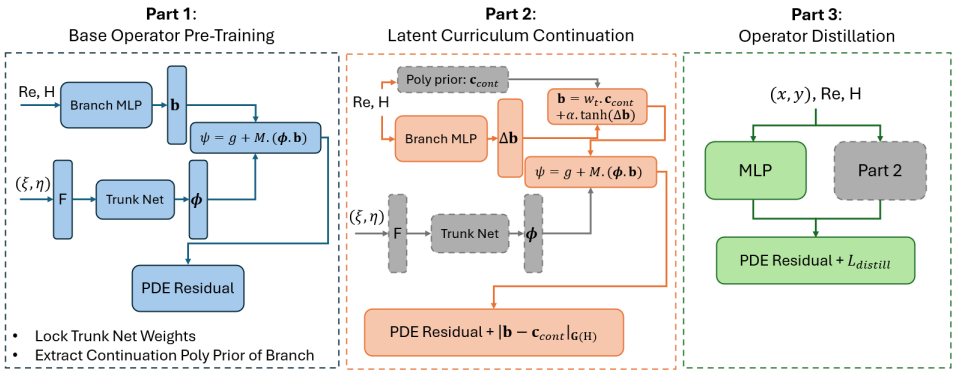

Figure 1: Three-stage framework schematic.

Table 2: Performance summary across the three evaluated PDEs.

	Burgers	Allen–Cahn	Cavity
Training range	$(\nu, L) \in [0.10, 0.20] \times [1, 2]$	$(\lambda, H) \in [35, 80] \times [1, 2]$	$(\mathrm{Re}, H) \in [100, 800] \times \{1\}$
Target	$(\nu, L) = (0.05, 4)$	$(\lambda, H) = (12.5, 3)$	$(\mathrm{Re}, H) = (3200, 1)$
OOD* distance	$2\times$	$0.35\times$ lower box edge	$4\times$
Operator: in-distribution error	0.003 ± 0.002	0.0015 ± 0.0004	0.2586 ± 0.1294
Operator: OOD error	0.12	0.78	0.53
Baseline PINN error	0.4123 ± 0.1601	1.00 ± 0.00	0.9800 ± 0.0248
Distilled PINN error (**Ours**)	0.0209 ± 0.0043	0.0037 ± 0.001	0.0663 ± 0.0099
Improvement	0.3914 ± 0.1635	0.9963 ± 0.001	0.9137 ± 0.0341

*OOD stands for out of distribution.

4 EXPERIMENTS

We evaluate the framework through a progressive sequence of experiments designed to address four core objectives. First, we use the viscous Burgers equation to establish whether an approximate, out-of-distribution teacher can accelerate a PINN (§4.1). Second, we analyze the Allen–Cahn equation to determine why this transfer succeeds in basin selection where pure residual minimization fails (§4.2). Third, we evaluate the lid-driven cavity to test if the mechanism scales to complex, high-dimensional systems far outside their training distribution (§4.3). Finally, an ablation study isolates precisely which component of the framework drives this success (§4.4).

4.1 EXPERIMENT 1 (BASIC TRANSFER): VISCOUS BURGERS (1D)

The 1D viscous Burgers equation establishes the fundamental transfer mechanism in a controlled setting featuring a unique monotone solution manifold, one internal transition layer, and a Cole–Hopf closed form for exact evaluation. The base operator is trained on $L \in [1, 2]$ in Part 1 and extrapolated to $L = 4$ at $\nu = 0.05$ in Part 2, extending twice beyond the maximum training length. Figure 2(a) illustrates the frozen-trunk operator at this out-of-distribution target: it captures the correct shock location and outer branches but exhibits an overshoot at the transition layer. This represents the precise approximate-yet-structurally-accurate prior the framework is designed to exploit. Across $n = 5$ independent seeds, the distilled student removes the overshoot and matches the analytical solution to graphical accuracy in every run (Figure 2(f)). It reaches $97.91\% \pm 0.43\%$ mean accuracy versus $58.77\% \pm 16.01\%$ for the from-scratch baseline, an improvement of $39.14\% \pm 16.35\%$ that holds in all 5/5 seeds. Beyond the higher mean, distillation markedly reduces run-to-run variance: the baseline's final accuracy is highly seed-dependent and occasionally collapses to a poor local solution, while the distilled student consistently converges to a tight neighborhood of the true solution regardless of initialization. Distillation also converges faster and to a lower final residual, with the reliability gate closing as the student's residual falls (Figure 2 d).

4.2 EXPERIMENT 2 (BASIN SELECTION): STEADY ALLEN–CAHN (2D)

This experiment demonstrates how the teacher resolves a solution-branch ambiguity that pure residual minimization cannot. At λ close to the pitchfork bifurcation $\lambda_1(H) = \pi^2(1 + 1/H^2)$, the principal positive branch $u^\star > 0$ exhibits an $O(\sqrt{\lambda - \lambda_1})$ amplitude, while the trivial solution ($u \equiv 0$) satisfies the residual exactly. We evaluate the framework at $\lambda = 12.5$ and $H = 3$ (where $\lambda_1(3) \approx 10.97$). This target resides approximately 14% above the onset, with a true peak amplitude of ≈ 0.45. Figure 3 (a) illustrates this dynamic. Baseline PINN collapses to the flat $u \equiv 0$ field, identifying a mathematically correct but physically trivial global minimizer; across $n = 5$ seeds, this collapse is total and deterministic, with $0.00\% \pm 0.00\%$ accuracy and a peak amplitude of only 0.0002 ± 0.0002 against a true peak of 0.4530. The Part 2 operator (second panel) correctly identifies the single-hump structure but over-predicts the amplitude, as the polynomial prior cannot precisely capture the $\sqrt{\lambda - \lambda_1}$ onset behavior, giving it only 32% accuracy on its own. Distilling this operator provides the necessary prior to pull the student into the correct basin of attraction despite the teacher's own imprecision. Subsequent residual minimization sharpens the field: Distilled PINN

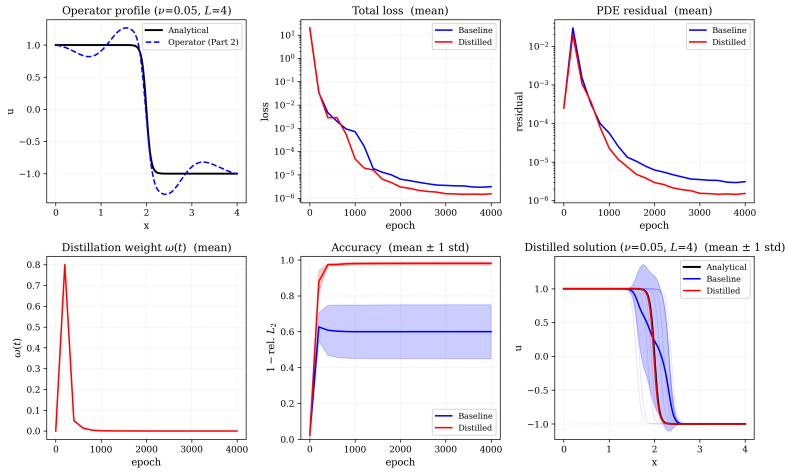

Figure 2: Burgers equation training metrics aggregated over $n = 5$ seeds. Panel (a) illustrates the Part 2 operator extrapolation versus the analytical reference. The remaining five panels detail the Part 3 distilled PINN performance.

matches the finite-difference reference in both shape and amplitude, reaching $99.63\% \pm 0.10\%$ mean accuracy across seeds. Figure 3 (b) confirms this quantitatively: accuracy climbs as the distillation weight anneals to zero, and the centerline profile aligns with the reference. This confirms the central hypothesis of the framework: an imperfect operator is sufficient because its role is solely to identify the correct physical basin, not to provide the exact solution. Here, a teacher accurate to only 32% still reliably rescues every seed from the trivial minimizer.

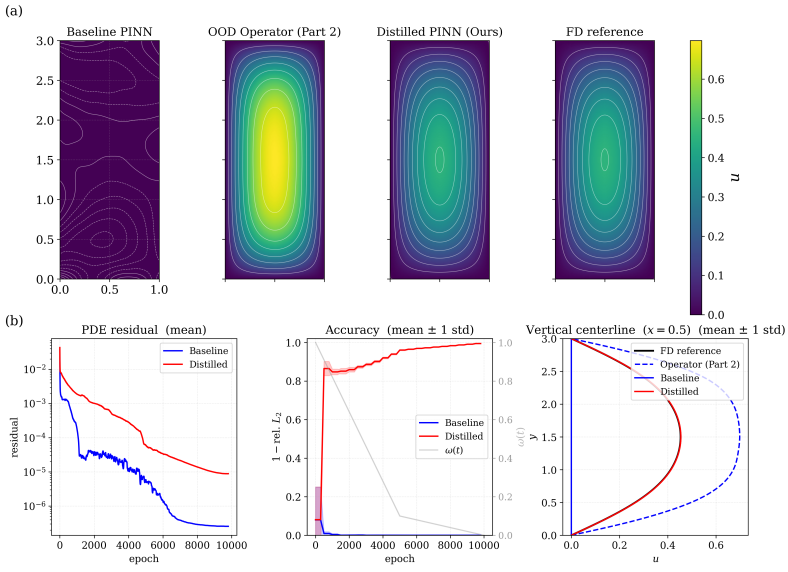

Figure 3: Allen–Cahn training metrics ($\lambda = 12.5$, $H = 3$), aggregated over $n = 5$ seeds, versus finite difference reference.

4.3 Experiment 3 (Large OOD extrapolation): Lid-driven cavity (2D)

To demonstrate scalability, the lid-driven cavity introduces a fourth-order, two-dimensional system extrapolated substantially outside its base box. The primary target is set to $\mathrm{Re} = 3200$ (far outside the $\mathrm{Re} \in [100, 800]$ training domain) to match the state-of-the-art single-case PINN benchmark

established by Wang et al. (2024a). A secondary experiment, extrapolating aspect ratio rather than Reynolds number, is reported in Appendix E.

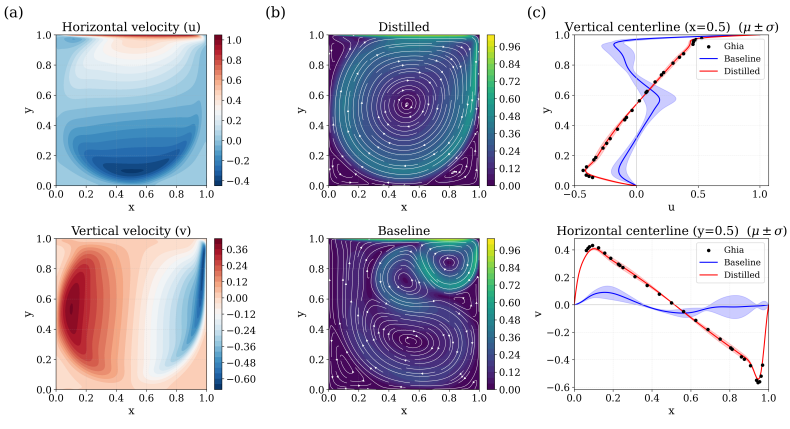

Figure 4: Lid-driven cavity at $\mathrm{Re} = 3200$. (a) Predicted u and v fields. (b) Velocity magnitude and streamlines for the distilled and baseline students. (c) Centerline profiles at $x = 0.5$ and $y = 0.5$ against Ghia et al. (1982). Fields and streamlines show the ensemble mean across $n = 5$ seeds.

Figure 4 illustrates the distilled student at $\mathrm{Re} = 3200$. Aggregated across $n = 5$ independent seeds, the u and v fields (a) successfully capture the primary vortex and thin wall jets. The streamlines (b) reproduce the primary recirculation and the two lower corner vortices in the distilled model, features that the baseline completely fails to capture. Consequently, the centerlines (c) tightly align with the benchmark points established by Ghia et al. (1982) and Marchi et al. (2021), whereas the baseline deviates severely.

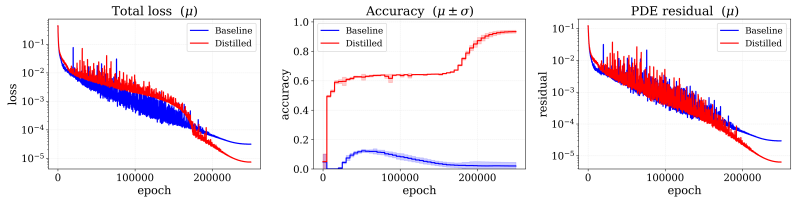

Figure 5: Lid-driven cavity ($\mathrm{Re} = 3200$) training metrics. Solid lines and shaded regions represent $\mu \pm \sigma$ across $n = 5$ independent seeds.

Figure 5 details the optimization trajectory: the baseline stalls at near-zero accuracy with a noisy residual, while the distilled student steadily climbs to high accuracy at a lower final residual, utilizing the identical architecture and step budget. Quantitatively, the baseline PINN entirely fails to converge to the correct physical state, yielding a mean accuracy of just $2.00\% \pm 2.48\%$. In contrast, the distilled student consistently resolves the solution-branch ambiguity, achieving a mean accuracy of $93.37\% \pm 0.99\%$. This represents a robust accuracy improvement of $91.4\% \pm 3.4\%$, an acceleration that holds consistently across all $5/5$ initialized seeds. Beyond the higher mean performance, distillation markedly reduces run-to-run variance, confirming that the operator initialization effectively guards against the inherent sensitivity of PINNs to random parameter initialization.

For computational context, Table 3 compares the Part 3 student with Wang et al. (2024a): our student uses roughly 20% of the parameters and 40% of the optimization steps (250K vs. 630K), but more point evaluations. End-to-end, Parts 1–3 require 546K optimization steps, but note that Part 1 is a one-time base-box cost reused across subsequent targets.

4.4 Experiment 4 (Mechanistic ablation): The role of Part 2

Section 4.3 presents the full framework, where Part 2 extrapolates the frozen-trunk operator over a four-stage Re sequence (§3.3) before distillation. This raises a key ablation: does incremental

Table 3: Part 3 computational comparison at $\mathrm{Re} = 3200$ with Wang et al. (2024a).

	Distilled PINN (**Ours**)	PirateNet
Rel. L^2 error against Ghia et al. (1982)	0.0663 ± 0.0099	0.0421
Depth $\times$ width	5×256	18×256
Trainable parameters	297 K	$\sim$1.32 M
Total training steps	250 K	630 K
Total point-evaluations	6.5×10^9	2.6×10^9

continuation uniquely determine basin selection, or is a direct evaluation of the Part 1 operator at $\mathrm{Re} = 3200$ an adequate prior? To test this, we keep the Part 3 architecture, the piecewise-linear distillation schedule, and the 250K-step training budget fixed, but initialize the teacher coefficients by directly evaluating the Part 1 branch at the target: $c = b_{\mathrm{Part\ 1}}(\mathrm{Re}{=}3200, H{=}1)$. This extrapolates the neural operator fourfold beyond its training range ($\mathrm{Re} \in [100, 800]$) without the regularizing polynomial continuation prior equation 4. We then compare this ablated model with the full framework and an unguided baseline PINN.

Table 4 and Figure 8 summarize the ensemble results across $n = 5$ independent seeds. Relying exclusively on the unguided Part 1 teacher consistently fails to identify the correct physical basin. As depicted in the mean velocity fields, the student's primary vortex is mislocated and the lower corner vortices are entirely absent, resulting in severe deviations from the benchmark (Ghia et al., 1982). Quantitatively, the ablated model yields a highly variable mean accuracy of $27.09\% \pm 13.49\%$. While this slightly outperforms the catastrophic collapse of the baseline PINN ($2.00\% \pm 2.48\%$), it falls drastically short of the full framework's $93.37\% \pm 0.99\%$.

Table 4: Ablation of the Part 2 continuation stage for the cavity at $\mathrm{Re} = 3200$, $H = 1$.

	Full framework	No Part 2	Baseline PINN
Teacher	Continued operator	Direct Part-1 operator	None
Relative L^2 accuracy	$93.37\% \pm 0.99\%$	$27.09\% \pm 13.49\%$	$2.00\% \pm 2.48\%$
PDE residual	6.3×10^{-6}	1.3×10^{-5}	2.9×10^{-5}

Crucially, this represents a basin-selection failure rather than an optimization failure. The mean PDE residual for the ablated student converges cleanly to 1.3×10^{-5}, which is strictly comparable to both the full framework's 6.3×10^{-6} and the baseline PINN's 2.9×10^{-5} (Table 4). This demonstrates that all three networks effectively minimize the governing physical equations, yet both the unguided baseline and the ablated model converge to steady states that deviate substantially from the Ghia et al. (1982) reference. This behavior perfectly characterizes the basin-selection vulnerability the proposed framework is designed to circumvent (§4.2).

5 Conclusion

We introduced a three-stage framework that decouples the distinct failure modes of operator learning and physics-informed neural networks. The central finding of this work, demonstrated across three nonlinear PDEs, is that an approximate neural operator need not be highly accurate to be effective; it needs only to initialize a student network within the correct basin of attraction, after which residual minimization governs final optimization. Our analyses confirm this mechanism, demonstrating that distilled students robustly recover the true physics in scenarios where standard PINNs deterministically collapse or stall at non-physical steady states. Future directions include extending the continuation prior to better capture bifurcation onsets, evaluating the framework in time-dependent multi-physics environments, and integrating this distillation strategy with state-of-the-art architectures (Wang et al., 2024a) to yield compounding improvements in optimization stability.

REPRODUCIBILITY STATEMENT

All experimental hyperparameters are given in Appendix C. The finite-difference reference solvers used for scoring and, for the cavity problem, anchor formation are standard, and their configurations are specified in Appendix D. An anonymized implementation and pretrained models are available at `https://anonymous.4open.science/r/PINN-Distillation-prototype-D3AF/`.

AI USE STATEMENT

During the preparation of this work, the authors used Claude (Anthropic) for assistance in writing the code used for analysis and evaluation, and to draft and revise manuscript text; Gemini (Google) was used to help improve manuscript text. The authors did not use generative AI tools for research ideation or experimental design; the three-stage framework and the extrapolation experiments described in this paper reflect the authors' own conceptual contributions. The authors also did not use generative AI tools for data analysis or interpretation of results. All AI-assisted work was reviewed before inclusion: LLM-generated code was verified and tested for correctness by the authors, and AI-drafted and AI-revised manuscript text was reviewed and edited by all the authors for accuracy and clarity. The authors take full responsibility for the content of the published article.

REFERENCES

O. Botella and R. Peyret. Benchmark spectral results on the lid-driven cavity flow. *Computers & Fluids*, 27(4):421–433, 1998. doi: 10.1016/S0045-7930(98)00002-4.

Qianying Cao, Somdatta Goswami, and George Em Karniadakis. Laplace neural operator for solving differential equations. *Nature Machine Intelligence*, 6(6):631–640, 2024a.

Qianying Cao, Somdatta Goswami, Tapas Tripura, Souvik Chakraborty, and George Em Karniadakis. Deep neural operators can predict the real-time response of floating offshore structures under irregular waves. *Computers & Structures*, 291:107228, 2024b.

Souvik Chakraborty. Transfer learning based multi-fidelity physics informed deep neural network. *Journal of Computational Physics*, 426:109942, 2021.

Shaan Desai, Marios Mattheakis, Hayden Joy, Pavlos Protopapas, and Stephen Roberts. One-shot transfer learning of physics-informed neural networks. *arXiv preprint arXiv:2110.11286*, 2021.

Ercan Erturk, Thomas C Corke, and Cihan Gökçöl. Numerical solutions of 2-d steady incompressible driven cavity flow at high reynolds numbers. *International journal for Numerical Methods in fluids*, 48(7):747–774, 2005.

Mohammad Sadegh Eshaghi, Cosmin Anitescu, Navid Valizadeh, Yizheng Wang, Xiaoying Zhuang, and Timon Rabczuk. Nows: Neural operator warm starts for accelerating iterative solvers. *Computer Methods in Applied Mechanics and Engineering*, 458:118989, 2026.

UKNG Ghia, Kirti N Ghia, and CT Shin. High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method. *Journal of computational physics*, 48(3):387–411, 1982.

Somdatta Goswami, Cosmin Anitescu, Souvik Chakraborty, and Timon Rabczuk. Transfer learning enhanced physics informed neural network for phase-field modeling of fracture. *Theoretical and Applied Fracture Mechanics*, 106:102447, 2020.

Somdatta Goswami, Katiana Kontolati, Michael D Shields, and George Em Karniadakis. Deep transfer operator learning for partial differential equations under conditional shift. *Nature Machine Intelligence*, 4(12):1155–1164, 2022.

Ehsan Haghighat, Mohammad Hesan Adeli, S Mohammad Mousavi, and Ruben Juanes. Stonet: A neural operator for modeling solute transport in micro-cracked reservoirs. *Advances in Water Resources*, pp. 105046, 2025.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Ameya D Jagtap, Kenji Kawaguchi, and George Em Karniadakis. Adaptive activation functions accelerate convergence in deep and physics-informed neural networks. *Journal of Computational Physics*, 404:109136, 2020.

George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

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

Aditi Krishnapriyan, Amir Gholami, Shandian Zhe, Robert Kirby, and Michael Mahoney. Characterizing possible failure modes in physics-informed neural networks. *Advances in neural information processing systems*, 34:26548–26560, 2021.

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

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Graph kernel network for partial differential equations. *arXiv preprint arXiv:2003.03485*, 2020b.

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

Lu Lu, Raphael Pestourie, Wenjie Yao, Zhicheng Wang, Francesc Verdugo, and Steven G Johnson. Physics-informed neural networks with hard constraints for inverse design. *SIAM Journal on Scientific Computing*, 43(6):B1105–B1132, 2021b.

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

Luis Mandl, Somdatta Goswami, Lena Lambers, and Tim Ricken. Separable physics-informed deeponet: Breaking the curse of dimensionality in physics-informed machine learning. *Computer Methods in Applied Mechanics and Engineering*, 434:117586, 2025.

Carlos Henrique Marchi, Cosmo Damião Santiago, and Carlos Alberto Rezende de Carvalho, Jr. Lid-driven square cavity flow: A benchmark solution with an $8192\times$ 8192 grid. *Journal of Verification, Validation and Uncertainty Quantification*, 6(4):041004, 2021.

Levi D McClenny and Ulisses M Braga-Neto. Self-adaptive physics-informed neural networks. *Journal of Computational Physics*, 474:111722, 2023.

Ken Perlin. Improving noise. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pp. 681–682, 2002.

Maziar Raissi, Paris Perdikaris, and George E 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.

Justin Sirignano and Konstantinos Spiliopoulos. Dgm: A deep learning algorithm for solving partial differential equations. *Journal of computational physics*, 375:1339–1364, 2018.

Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020.

Tapas Tripura and Souvik Chakraborty. Wavelet neural operator for solving parametric partial differential equations in computational mechanics problems. *Computer Methods in Applied Mechanics and Engineering*, 404:115783, 2023.

Sifan Wang, Yujun Teng, and Paris Perdikaris. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM Journal on Scientific Computing*, 43(5):A3055–A3081, 2021a.

Sifan Wang, Hanwen Wang, and Paris Perdikaris. On the eigenvector bias of fourier feature networks: From regression to solving multi-scale pdes with physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 384:113938, 2021b.

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, 2021c.

Sifan Wang, Xinling Yu, and Paris Perdikaris. When and why pinns fail to train: A neural tangent kernel perspective. *Journal of Computational Physics*, 449:110768, 2022.

Sifan Wang, Shyam Sankaran, Hanwen Wang, and Paris Perdikaris. An expert's guide to training physics-informed neural networks. *arXiv preprint arXiv:2308.08468*, 2023a.

Sifan Wang, Bowen Li, Yuhan Chen, and Paris Perdikaris. Piratenets: Physics-informed deep learning with residual adaptive networks. *Journal of Machine Learning Research*, 25(402):1–51, 2024a.

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

Wei Wang, Tang Paai Wong, Haihui Ruan, and Somdatta Goswami. Causality-respecting adaptive refinement for pinns: enabling precise interface evolution in phase field modeling. *Machine Learning for Computational Science and Engineering*, 2(1):10, 2026.

Zhicheng Wang, Xuhui Meng, Xiaomo Jiang, Hui Xiang, and George Em Karniadakis. Solution multiplicity and effects of data and eddy viscosity on navier-stokes solutions inferred by physics-informed neural networks. *arXiv preprint arXiv:2309.06010*, 2023b.

Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):9, 2016.

Chenxi Wu, Min Zhu, Qinyang Tan, Yadhu Kartha, and Lu Lu. A comprehensive study of non-adaptive and residual-based adaptive sampling for physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 403:115671, 2023.

Jeremy Yu, Lu Lu, Xuhui Meng, and George Em Karniadakis. Gradient-enhanced physics-informed neural networks for forward and inverse pde problems. *Computer Methods in Applied Mechanics and Engineering*, 393:114823, 2022.

A Per-problem instantiations of the shared architecture

A.1 Domains and the isoparametric map

The two-dimensional problems are posed on $\Omega_H = (0,1)\times(0,H)$; the one-dimensional problem on $(0,L)$. A reference domain $(\xi,\eta)\in[0,1]^2$ is mapped to physical coordinates by $(x,y)=(\xi,H\eta)$, yielding $\partial_x = \partial_\xi$ and $\partial_y = H^{-1}\partial_\eta$. Analogously, the one-dimensional reference coordinate $\xi\in[0,1]$ is mapped to the physical domain by $x = L\xi$, yielding $\partial_x = L^{-1}\partial_\xi$. The trunk basis is defined exclusively on the reference domain, such that geometry enters solely through algebraic chain-rule weights. Reference-space partial derivatives of the frozen basis remain geometry-independent; consequently, assembling the physical operator at a new geometry (i.e. a new L or H) reduces to algebraic scaling.

A.2 BURGERS: PDE, ANSATZ, RESIDUAL

On $(0, L)$ with $u(0) = +A$, $u(L) = -A$ (we use $A = 1$ in all experiments),

$$u\,\partial_x u - \nu\,\partial_{xx} u = 0, \tag{7}$$

with ansatz $u(\xi; \boldsymbol{p}) = A(1-2\xi)+4\xi(1-\xi)\sum_k b_k(\boldsymbol{p})\varphi_k(\xi)$ and residual $r = L^{-1}u\partial_\xi u - \nu L^{-2}\partial_{\xi\xi} u$. In the notation of equation 1, the boundary lift is $g(\xi; \boldsymbol{p}) = A(1-2\xi)$ and the multiplier is $\mathcal{M}(\xi) = 4\xi(1-\xi)$, which vanishes to first order at $\xi \in \{0, 1\}$ (Table 1, Design choice ii).

A.3 ALLEN–CAHN: PDE, ANSATZ, RESIDUAL, BIFURCATION

On Ω_H with $u = 0$ on $\partial\Omega_H$,

$$\nabla^2 u + \lambda(u - u^3) = 0. \tag{8}$$

The critical rate is $\lambda_1(H) = \pi^2(1 + 1/H^2)$, representing the first Dirichlet eigenvalue of $-\nabla^2$ on the domain. Below $\lambda_1(H)$, the only solution is the trivial state $u \equiv 0$; above it, a pair $\pm u^\star$ bifurcates along the principal (nodeless, sign-definite) eigendirection of $-\nabla^2$, scaling as $\|u^\star\|_\infty \sim \sqrt{\lambda - \lambda_1}$ near the onset and saturating to $\|u^\star\|_\infty \to 1$ with boundary layers of width $\sim 1/\sqrt{\lambda}$ further above the critical rate. Because the seed lift $\ell_0 > 0$ (defined below) biases the ansatz toward the positive member of this pair, we take $u^\star > 0$, the principal positive branch, as the target solution throughout. The base training domain $\lambda \in [35, 80], H \in [1, 2]$ is restricted to the comfortably supercritical regime.

The ansatz employs a fixed non-trivial seed lift $\ell_0(\xi, \eta) = A_0 \sin(\pi\xi)\sin(\pi\eta)$, with $A_0 = 0.3$ used in all experiments:

$$u(\xi, \eta; \boldsymbol{p}) = A_0 \sin(\pi\xi)\sin(\pi\eta) + 16\xi(1-\xi)\eta(1-\eta)\sum_k b_k(\boldsymbol{p})\varphi_k(\xi, \eta). \tag{9}$$

In the notation of equation 1, $g(\xi, \eta; \boldsymbol{p}) = \ell_0(\xi, \eta)$ (constant in $\boldsymbol{p}$) and $\mathcal{M}(\xi, \eta) = 16\xi(1-\xi)\eta(1-\eta)$, which vanishes to first order on $\partial\Omega_H$ (Table 1, Design choice ii). This formulation explicitly breaks the $u \equiv 0$ degeneracy at initialization; setting $b \equiv 0$ yields a positive interior profile rather than a trivial zero field. The seed function vanishes on $\partial\Omega_H$ and satisfies $\nabla^2\ell_0 = -\pi^2(1 + H^{-2})\ell_0 = -\lambda_1(H)\,\ell_0$: it is an eigenfunction of the linear Laplacian, with eigenvalue exactly equal to the critical bifurcation rate, but it is not an eigenfunction of the nonlinear residual equation 8, so it is never itself a spurious fixed point of training.

A.4 CAVITY: PDE, ANSATZ, RESIDUAL

The streamfunction ψ satisfies the vorticity-transport equation:

$$-u\,\partial_x(\nabla^2\psi) - v\,\partial_y(\nabla^2\psi) + \mathrm{Re}^{-1}\nabla^4\psi = 0, \quad u = \partial_y\psi,\ v = -\partial_x\psi, \tag{10}$$

enforcing velocity Dirichlet conditions on all walls alongside a prescribed lid velocity u_{lid}. The ansatz $\psi = g(\boldsymbol{x}) + \mathcal{M}(\boldsymbol{x})\sum_k b_k\varphi_k(\boldsymbol{x})$ employs a lift function $g(\xi, \eta) = H u_{\mathrm{lid}}(\xi)(\eta^3 - \eta^2)$ and a multiplier $\mathcal{M} = (16\xi(1-\xi)\eta(1-\eta))^2$. Because $\mathcal{M}$ vanishes to the second order on all boundaries (Table 1, Design choice ii), both ψ and its normal derivative $\partial_n\psi$ exactly satisfy the boundary conditions independent of the network coefficients.

Regularized lid profile. A uniformly driven lid ($u = 1$) introduces a velocity discontinuity at the two top corners where the moving boundary intersects the stationary side walls. This corner singularity generates unbounded vorticity, which frequently destabilizes high-Re residual training (Wang et al., 2023b). To circumvent this, we regularize the boundary condition. The lid speed is maintained at exactly 1 over the interior of the top wall and transitions smoothly to 0 at each corner across a margin of width $\delta = 0.12$:

$$u_{\mathrm{lid}}(\xi) = S\Big(\tfrac{\xi}{\delta}\Big)\,S\Big(\tfrac{1-\xi}{\delta}\Big), \qquad \begin{aligned} &S(t) = 0, && t < 0,\\ &S(t) = 6t^5 - 15t^4 + 10t^3, && 0 \le t \le 1,\\ &S(t) = 1, && t > 1. \end{aligned} \tag{11}$$

where S is Perlin's quintic smootherstep function (Perlin, 2002). This formulation ensures matching values, slopes, and curvatures ($S(0)$=0, $S(1)$=1, S'=S''=0 at both domain ends), guaranteeing that

the transition into the flat core introduces no gradient discontinuities up to the second derivative. With $\delta = 0.12$, the lid remains exactly 1 over the central 76% of the span, modifying the physical driving force only within the narrow corner margins. This identical u_{lid} profile is applied to the lift g, the finite-difference reference solver, and the Part 3 student's boundary loss, ensuring the teacher and student solve the exact same regularized boundary-value problem.

Part 3 primitive-variable formulation. Unlike Parts 1–2, which operate on the streamfunction ψ, the Part 3 cavity student is a primitive-variable PINN $u_\phi(\boldsymbol{x}) = (u_\phi, v_\phi, p_\phi)$ trained directly on the steady incompressible Navier–Stokes equations:

$$u_\phi\,\partial_x u_\phi + v_\phi\,\partial_y u_\phi = -\partial_x p_\phi + \mathrm{Re}^{-1}\big(\partial_{xx} u_\phi + \partial_{yy} u_\phi\big), \tag{12}$$

$$u_\phi\,\partial_x v_\phi + v_\phi\,\partial_y v_\phi = -\partial_y p_\phi + \mathrm{Re}^{-1}\big(\partial_{xx} v_\phi + \partial_{yy} v_\phi\big), \tag{13}$$

$$\partial_x u_\phi + \partial_y v_\phi = 0. \tag{14}$$

Dirichlet velocity conditions are enforced softly: $u_\phi = v_\phi = 0$ on the left, right, and bottom walls, and $u_\phi = u_{\mathrm{lid}}(x)$, $v_\phi = 0$ on the top wall, using the identical regularized lid profile equation 11 applied throughout the pipeline. Because p_ϕ is determined only up to an additive constant, we impose a soft gauge penalty pinning its value at the domain center to zero, $p_\phi(0.5, H/2) \approx 0$. The complete Part 3 objective is

$$\mathcal{L}(\phi; t) \;=\; \mathcal{L}_{\mathrm{mom},x} + \mathcal{L}_{\mathrm{mom},y} + \mathcal{L}_{\mathrm{cont}} \;+\; \lambda_{\mathrm{BC}}\,\mathcal{L}_{\mathrm{BC}} \;+\; \lambda_{\mathrm{gauge}}\,p_\phi(0.5, H/2)^2 \;+\; w(t)\,\mathcal{L}_{\mathrm{distill}}(t), \tag{15}$$

where $\mathcal{L}_{\mathrm{mom},x}$, $\mathcal{L}_{\mathrm{mom},y}$, and $\mathcal{L}_{\mathrm{cont}}$ are the mean-squared residuals of equation 12–equation 14 over interior collocation points, $\mathcal{L}_{\mathrm{BC}}$ is the mean-squared boundary-velocity violation, and λ_{BC}, λ_{gauge} are the momentum-normalized BC and gauge weights reported in Table 7 (10.0 and 10.0 respectively for the main Re-extrapolation target; the momentum terms carry unit weight). The distillation term is applied only to velocity:

$$\mathcal{L}_{\mathrm{distill}}(t) \;=\; \big\|u_\phi - u_{\mathrm{teacher}}\big\|^2_{\Gamma(\boldsymbol{x})} \;+\; \big\|v_\phi - v_{\mathrm{teacher}}\big\|^2_{\Gamma(\boldsymbol{x})}, \tag{16}$$

where $(u_{\mathrm{teacher}}, v_{\mathrm{teacher}}) = (\partial_y \psi_{\mathrm{teacher}}, -\partial_x \psi_{\mathrm{teacher}})$ are the induced velocities of the Part 2 streamfunction field, obtained by automatic differentiation of the frozen Part 2 network. No distillation target is imposed on p_ϕ: the streamfunction teacher carries no pressure information, so the pressure field is constrained solely by the momentum residuals and the gauge penalty. This primitive-variable formulation for Part 3 is used identically for both the Re-extrapolation (§4.3) and H-extrapolation (Appendix E) cavity experiments; the Burgers and Allen–Cahn students, by contrast, retain the scalar-field loss equation 6 directly on u_ϕ, since those problems have no pressure or continuity constraint.

Interior collocation points for the momentum and continuity residuals are not sampled uniformly: the wall-layer thickness scales as $\delta_{\mathrm{wall}} \sim \mathrm{Re}^{-1/2}$, comparable to or smaller than the spacing of a uniform sample, which otherwise lets the student smooth away the near-wall velocity extrema. We instead draw 40% of interior points from thin near-wall bands (width $4\mathrm{Re}^{-1/2}$, biased toward each wall) and the remaining 60% uniformly, applied identically to the baseline and distilled arms so it does not bias their comparison. Distillation query points are sampled uniformly rather than wall-clustered, since the teacher's velocity error is largest near the walls and clustering there would concentrate the distillation pull on its least reliable signal.

A.5 Spatial bandwidth and spectral bias

Standard coordinate-based networks exhibit spectral bias, so a random Fourier embedding is used to shift the burden of representing high-frequency features onto the analytical derivatives of the basis sinusoids. The required bandwidth scales with the PDE's order: a frequency-k component entering an n-th-order residual is amplified by a factor of $\sim k^n$, bounded by k^2 for the second-order Burgers and Allen–Cahn systems but reaching k^4 for the fourth-order cavity system, further exacerbated by a factor of H^{-4} at high aspect ratios. The cavity's trunk bandwidth is accordingly the most constrained of the three, using an anisotropic $\sigma = [\sigma_\xi, \sigma_\eta]$; Allen–Cahn and Burgers use a milder, isotropic bandwidth. Exact values for all three are given in Tables 5–7.

A.6 WHERE THE THREE INSTANTIATIONS DIFFER

Beyond the three primary design choices outlined in Table 1, the implementations differ across four specific algorithmic parameters (detailed in Appendices B through C):

(a) The Burgers base operator employs a pure autograd-derived residual. Conversely, Allen–Cahn and the cavity utilize a weighted finite-difference stencil residual, which significantly reduces computational overhead for the biharmonic cavity operator and the batched Allen–Cahn Laplacian. **(b)** The Allen–Cahn framework incorporates a soft amplitude floor into the Part 1 loss, penalizing batch samples whose fields collapse toward $u{\equiv}0$. This serves as a complementary regularization to the seed lift ℓ_0, mitigating the risk of the network encountering the competing trivial minimizer during early training iterations. **(c)** The extrapolation sequence detailed in §3.3 requires one stage for the Burgers equation and the cavity's aspect-ratio variant (Appendix E), two stages for the Allen–Cahn problem, and four stages for the principal cavity Re-extrapolation. **(d)** Allen–Cahn and the primary Re-extrapolated cavity deploy the ungated, piecewise-linear distillation schedule. Burgers and the cavity H-extrapolation use the physics-adaptive, reliability-gated schedule (§3.4).

B DISTILLATION SCHEDULE AND RELIABILITY GATE

The Part 3 loss equation 6 weights the distillation term using two distinct components: a temporal scalar schedule $w(t)$ and an optional spatial per-point gate $\Gamma(\boldsymbol{x})$. The Allen–Cahn system and the cavity's Re-extrapolation (main text) utilize an ungated, piecewise-linear temporal schedule ($\Gamma \equiv 1$, the identity); the Burgers equation and the cavity's H-extrapolation (Appendix E) utilize a physics-adaptive temporal schedule coupled with the spatial reliability gate defined below.

Piecewise-linear weight. Over a fraction $f_{\min}$ of the total training budget T, $w(t)$ decays from an initial value w_0 to an intermediate value $w_{\min}$, and subsequently decays to $w_{\mathrm{end}}{=}0$ over the remainder of training:

$$w(t) = \begin{cases} w_0 + (w_{\min} - w_0)\,\dfrac{t}{f_{\min}T}, & t \le f_{\min}T, \\[2ex] w_{\min} + (w_{\mathrm{end}} - w_{\min})\,\dfrac{t - f_{\min}T}{(1 - f_{\min})T}, & t > f_{\min}T. \end{cases} \tag{17}$$

For the cavity's Re-extrapolation ($\mathrm{Re}{=}3200$), the parameters are set to $(w_0, f_{\min}, w_{\mathrm{end}}) = (0.7, 0.7, 0)$ with $w_{\min}{=}0$. This constitutes a linear decay to zero over the first 70% of training, followed by exclusive residual minimization. Truncating $w(t)$ to exactly zero during the final training window allows the student network to refine the solution independent of the teacher's structural errors.

Physics-adaptive weight. While the piecewise-linear schedule is sufficient for Allen–Cahn and the cavity's Re-extrapolation, we use a physics-adaptive schedule for the Burgers equation and the cavity's H-extrapolation to demonstrate the framework's flexibility in accommodating alternative distillation strategies. After an initial warm-up period of T_{warm} epochs, during which $w(t)$ increases linearly to $w_{\max}\alpha_w$, the weight dynamically tracks the convergence of the student network: $w = w_{\max}\alpha \cdot \min(1, \bar{r}_\phi / \bar{r}_\phi^{\,\mathrm{ref}})$. Here, $\bar{r}_\phi$ represents an exponential moving average (with a decay factor of 0.5) of the student's PDE loss, and $\bar{r}_\phi^{\,\mathrm{ref}}$ is its reference value at the conclusion of the warm-up phase. The weight $w(t)$ is strictly zeroed once training surpasses a specified cutoff fraction of T. For the Burgers equation, the parameters are configured as $(\alpha_w, w_{\max}, T_{\mathrm{warm}}, \text{cutoff frac.}) = (0.8, 1.0, 200, 0.5)$; the cavity's H-extrapolation uses the same $(\alpha, w_{\max}, T_{\mathrm{warm}}, \text{cutoff frac.})$ values. This formulation dynamically attenuates the teacher's influence in response to the student's improving physics residual, rather than relying on a predetermined epoch schedule.

Spatial reliability gate $\Gamma(\boldsymbol{x})$. When enabled (as in the Burgers and cavity H-extrapolation configurations), the spatial weight Γ acts as a per-collocation-point factor that attenuates the distillation constraint in regions where the student's PDE residual $r_\phi(\boldsymbol{x})$ is large:

$$\Gamma(\boldsymbol{x}) \;=\; \big(1 - \tanh(\tau\, r_\phi(\boldsymbol{x}))\big) \;\vee\; \Gamma_{\mathrm{floor}}, \qquad \Gamma_{\mathrm{floor}} = 0.1. \tag{18}$$

This value is clamped to a minimum floor Γ_{floor} to ensure the teacher's prior is never entirely deactivated. In regions where the student adequately satisfies the governing physics (r_ϕ is small), the gate

Table 5: Burgers hyperparameters (§4.1).

Part 1 — base operator	
Base box (ν, L)	$[0.10, 0.20] \times [1.0, 2.0]$
Anchors	$\{0.10, 0.15, 0.20\} \times \{1.0, 1.5, 2.0\}$ (9)
Trunk / branch / basis K	128×5 tanh / 128×4 tanh / 96
Fourier features m / σ	64 / 6.0
Operator steps / LR ($\to$ min)	2×10^4 / $10^{-3} \to 10^{-6}$
Collocation pts / param. batch / grad clip	256 / 16 / 1.0
Part 2 — continuation	
Target	$(\nu, L) = (0.05, 4.0)$
Bounded-correction cap α	0.63
β range	$[10^{-3}, 10^{8}]$
Steps / LR / batch	1.2×10^4 / 8×10^{-4} / 8
Part 3 — guarded distillation	
Student / Fourier m, σ	128×4 tanh / 32, 8.0
Epochs / LR ($\to$ min)	4000 / $5 \times 10^{-4} \to 10^{-6}$
Collocation / BC / distil pts	1000 / 256 / 1000
BC weight / grad clip	10.0 / 1.0
Distillation schedule	physics-adaptive + gate
Gate τ/floor; $(\alpha_w, w_{\max}, T_{\text{warm}}, \text{cutoff})$	$2.0/0.1$; $(0.8, 1.0, 200, 0.5)$
Seeds (n / start)	5 / 1729

approaches 1 and the teacher's field is strictly enforced. Conversely, where the residual is large, the gate decays toward the floor value. For the Burgers equation, we utilize a lowered temperature of $\tau = 2.0$ as the residual magnitude $|r_\phi|$ peaks inside the internal transition layer. Because this layer contains the critical structural prior provided by the 1D teacher, a high-temperature gate would inappropriately deactivate distillation in this region and restrict knowledge transfer exclusively to the trivial outer domains. Lowering τ ensures the gate remains sufficiently open at moderate residual values, a behavior we empirically verified by monitoring the gate-open fraction during training. The cavity's H-extrapolation instead uses $\tau = 20.0$ (with the same floor $\Gamma_{\text{floor}} = 0.1$), reflecting the different residual scale of the two-dimensional momentum system.

C Hyperparameters and training details

Table 5 gives the Burgers hyperparameters (§4.1), Table 6 the Allen–Cahn hyperparameters (§4.2), and Table 7 the cavity hyperparameters for both the main-text Re-extrapolation (§4.3) and the Appendix E H-extrapolation, which are independently trained base operators. All values are taken directly from the released configuration; code is released alongside the paper for further implementation details.

D Reference-solver configurations

Reference fields are used only for post-hoc scoring and, for the cavity, for forming the eight main-text Part 1 anchors (twelve for the separate Appendix E base operator); they never enter any training loss. For Burgers the reference is the Cole–Hopf closed form, evaluated exactly. For Allen–Cahn the reference is a Newton solve of the residual on a five-point Laplacian stencil. For the cavity the reference is a regularized-lid, streamfunction–vorticity finite-difference solver on a uniform N_x=129 grid (isoparametrically stretched to $N_y = \lceil (N_x - 1)H \rceil + 1$ for $H \neq 1$), iterated to a steady state; it uses the same regularized lid profile u_{lid} as the operator (Appendix A.4). Each cavity anchor solve is gate-checked against the Marchi et al. (Marchi et al., 2021) high-resolution benchmark where tabulated; the few-percent offset expected from the regularized-versus-singular lid is accounted for in that check.

Table 6: Allen–Cahn hyperparameters (§4.2).

Part 1 — base operator	
Base box (λ, H)	$[35, 80] \times [1.0, 2.0]$
Anchors	$\{35, 50, 65, 80\} \times \{1.0, 1.5, 2.0\}$ (12), GN-polished
Trunk / branch / basis K	128×4 tanh / 128×4 tanh / 128
Fourier features m / σ	32 / $[3.0, 3.0]$
Operator steps / LR ($\to$ min)	3×10^4 / $10^{-3} \to 10^{-6}$
FD residual grid / param. batch	48 / 8
Anti-collapse amplitude floor / weight	$\mathrm{rms}(u) \geq 0.25$ / 100
GN iters / tol / damping / grad clip	60 / 10^{-10} / 10^{-4} / 1.0
Part 2 — continuation	
Targets	two-rung ladder: λ=18 $\to$ λ=12.5 (both at H=3)
Bounded-correction cap α	0.63
β range	$[1.0, 2000]$
Poly. degree	2
Steps / LR / batch	1.2×10^4 / 5×10^{-4} / 8
Part 3 — guarded distillation	
Student / Fourier m, σ	64×2 tanh / 32, 2.0
Epochs / LR ($\to$ min)	10^4 / $5 \times 10^{-4} \to 10^{-7}$
Collocation / BC / distil pts	2000 / 800 / 2000
BC weight / grad clip	10.0 / 1.0
Distillation schedule	piecewise-linear
$(w_0, f_{\min}, w_{\min}, w_{\mathrm{end}})$	$(1.0,\ 0.5,\ 0.1,\ 0.0)$
Seeds (n / start)	5 / 1729

E ADDITIONAL RESULTS: ASPECT-RATIO EXTRAPOLATION

Section 4.3 reports the cavity's Part 3 result at the Re-extrapolated target ($\mathrm{Re} = 3200$, $H = 1$), reached from a base operator trained on $\mathrm{Re} \in [100, 800]$ at fixed H=1 with 8 anchors. Here we report a separate experiment stressing the complementary axis: a base operator trained on the full (Re, H) box with 12 anchors (Table 7), single-shot extrapolated from $H \in [1, 2]$ out to $H = 3$ at fixed $\mathrm{Re} = 400$, followed by Part 3 distillation of a fresh student PINN at that target. Every other mechanism is exactly as in the main pipeline (§3.3–§3.4). This example shows that the framework's benefit from distillation is not specific to the Re-extrapolation axis reported in the main text; the same guarded-distillation mechanism accelerates convergence when the extrapolation instead stresses geometry.

Figure 6 compares the baseline and distilled students against the finite-difference reference across training progress (25/50/75/100% of the training budget) for both velocity components. Figure 7 reports the corresponding total loss, accuracy against the FD reference, and PDE residual histories. As in the main-text $\mathrm{Re} = 3200$ case, the distilled student converges faster and reaches a lower final residual and higher final accuracy than the baseline trained under an identical architecture and step budget.

Table 7: Cavity hyperparameters: main-text Re-extrapolation (Re=3200, H=1, §4.3) vs. the Appendix E H-extrapolation (H=3, Re=400).

	Main: Re=3200, H=1	App. E: Re=400, H=3
Part 1 — base operator		
Base box (Re, H)	$[100, 800] \times \{1.0\}$	$[100, 400] \times [1.0, 2.0]$
Anchors	8	12
Trunk / branch / basis K	128×8 tanh / 128×8 tanh / 192	256×6 tanh / 256×4 tanh / 192
Fourier features m / σ	64 / [4.0, 7.0]	48 / [4.0, 7.0]
Operator steps / LR ($\to$ min)	2×10^5 / $10^{-3} \to 10^{-6}$	3×10^5 / $6 \times 10^{-4} \to 10^{-6}$
FD residual grid / param. batch	160 / 4	64 / 8
Ridge μ / poly degree	10^{-2} / 2	10^{-2} / 3
Grad clip	1.0	1.0
Part 2 — continuation		
Targets	4-rung Re ladder: 1400→2000→2500→3200	single-shot: (Re, H)=(400, 3)
Bounded-correction cap α	0.63	0.63
β calibration clamp $[\beta_{\min}, \beta_{\max}]$	[20, 2000]	[20, 2000]
β within-step decay (start $\to$ end)	$\beta_0 \to \beta_0/20$	$\beta_0 \to \beta_0/20$
Steps / LR / batch	2.4×10^4 / 5×10^{-4} / 8	2.4×10^4 / 5×10^{-4} / 8
Part 3 — guarded distillation		
Student / Fourier m, σ	256×5 tanh / 64, 6.0	128×3 tanh / 32, 4.0
Epochs / LR ($\to$ min)	2.5×10^5 / $10^{-3} \to 10^{-8}$	5×10^4 / $5 \times 10^{-4} \to 10^{-6}$
Collocation / BC / distil pts	6000 / 4000 / 4000	4000 / 800 / 4000
Momentum / BC / gauge weight	1.0 / 10 / 10.0	1.0 / 10 / 1.0
Grad clip	1.0	1.0
Distillation schedule	piecewise-linear	physics-adaptive + gate
Seeds (n / start)	5 / 1729	single run

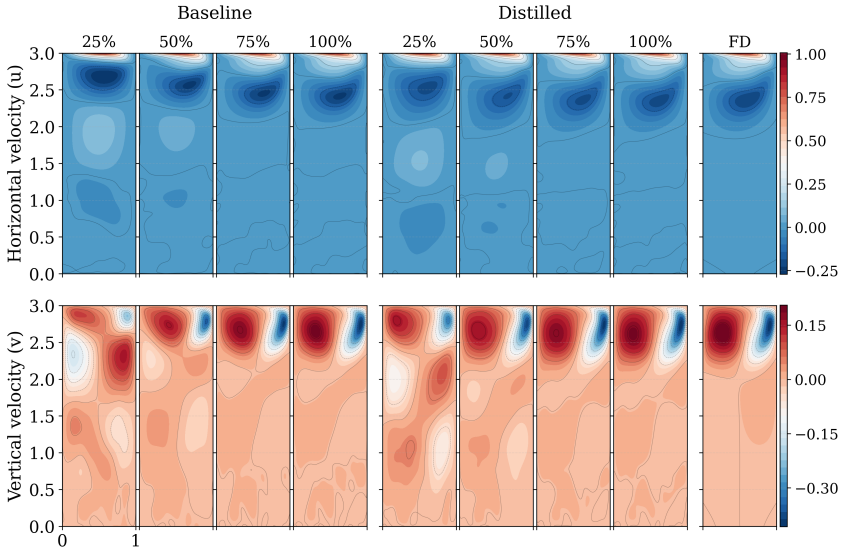

Figure 6: Lid-driven cavity, H-extrapolation example at Re $= 400$, $H = 3$ (base box $H \in [1, 2]$). Predicted horizontal (u, top row) and vertical (v, bottom row) velocity fields at 25/50/75/100% of training for the baseline (left block) and distilled (right block) students, against the finite-difference reference (rightmost column).

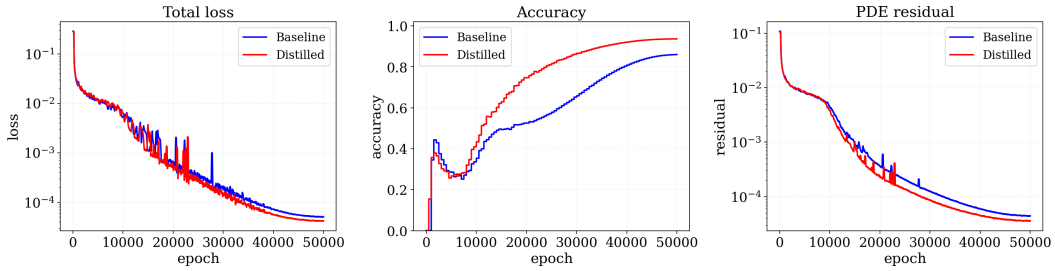

Figure 7: Lid-driven cavity, H-extrapolation example at $\mathrm{Re} = 400$, $H = 3$.

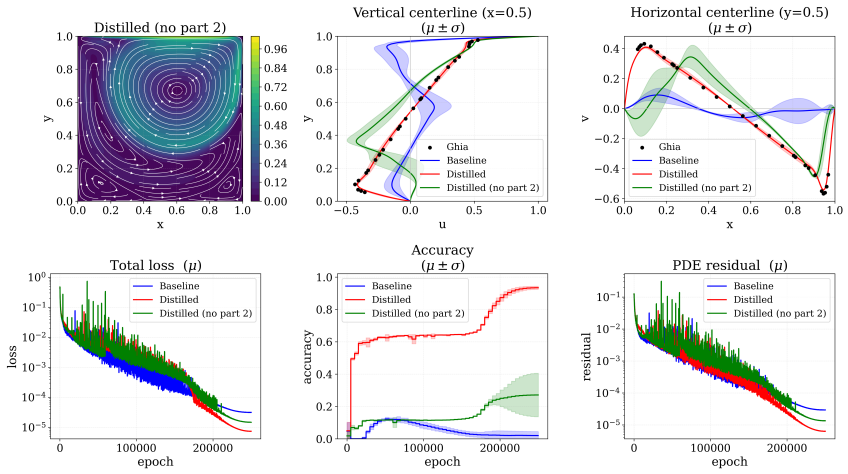

Figure 8: Cavity Part-2 removal ablation at $\mathrm{Re} = 3200$, $H = 1$. Solid lines and shaded regions represent $\mu \pm \sigma$ over $n = 5$ seeds.