BEYOND PARAMETER-SPACE CONFLICT: INTRINSIC FUNCTION-SPACE GEOMETRY FOR PHYSICS-CONSTRAINED FLOW MATCHING

Anonymous authors
Paper under double-blind review

ABSTRACT

Physics-constrained generative modeling requires balancing distribution fidelity with physical consistency, two objectives that can induce conflicting optimization directions. Existing approaches typically resolve this conflict using parameter gradients, but we show that the resulting geometry depends on the chosen parameterization: function-preserving reparameterizations can substantially alter measured conflict without changing the represented model. Because multi-objective rules use this geometry to determine the update itself, equivalent parameterizations can induce different optimization decisions for the same model function. Based on this observation, we introduce Intrinsic Geometry Flow Matching (IGFM), which defines objective conflict over realizable first-order changes in function space and constructs a common-descent direction within this intrinsic geometry. For a fixed observation map and output metric, the resulting ideal direction is invariant to locally invertible changes of parameter coordinates and admits a scale-free common-descent guarantee. Across Darcy flow, Kolmogorov flow, and Structural Topology Optimization, IGFM reduces the primary physics or engineering error by 40.0%, 22.4%, and 55.7%, respectively, relative to the strongest PBFM baseline, while maintaining comparable task-specific fidelity. These theoretical and empirical findings support realizable function-space geometry as a principled framework for resolving physics–distribution conflicts in generative modeling.

1 INTRODUCTION

Physics-constrained generative modeling aims to learn complex data distributions while ensuring that generated samples satisfy physical requirements (Shu et al., 2023). Flow matching provides an efficient framework for learning such distributions (Lipman et al., 2023; Tong et al., 2024), but incorporating physical constraints creates an optimization tension: improving physical consistency can interfere with distribution matching, while prioritizing generative fidelity can leave substantial physical error (Baldan et al., 2026). Recent approaches formulate this tension as a multi-objective optimization problem and resolve it using loss-specific parameter gradients. ConFIG (Liu et al., 2025) constructs common-descent updates from conflicting gradients, while PBFM (Baldan et al., 2026) applies such gradient-based conflict resolution to physics-constrained flow matching.

These methods implicitly treat the geometry of parameter gradients as a faithful description of how the objectives interact. Yet parameter gradients are tied to a particular parameterization of the model, whereas neural networks admit multiple parameterizations that represent the same function. This raises a basic question: *is the measured conflict between objectives an intrinsic property of the represented model?* We show that, in general, it is not. For any two linearly independent objective gradients, a locally invertible reparameterization can change their Euclidean cosine to any prescribed value in $(-1, 1)$ without changing the represented function. The ambiguity is therefore not in the objectives themselves, but in the parameter-space geometry used to compare them.

This distinction directly affects optimization. Multi-objective methods use gradient geometry not only to diagnose conflict but also to determine the update, so equivalent parameterizations of the same model function can receive different optimization decisions from the same conflict-resolution procedure. Figure 1 illustrates this effect on a two-objective toy problem: although the represented

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.

objectives are unchanged, parameter-space conflict resolution produces parameterization-dependent update trajectories. In contrast, realizable function-space geometry is defined through changes of the represented function itself. The same phenomenon appears in trained networks: across function-preserving reparameterizations of a trained Darcy model, the parameter-gradient cosine varies substantially, while the intrinsic statistic stays stable under matched audits of the same model.

These observations suggest that objective conflict should be defined through changes of the represented function rather than through the coordinates used to parameterize it. Based on this principle, we introduce Intrinsic Geometry Flow Matching (IGFM), which represents each objective by its first-order effect on the model output, restricted to the changes that parameter perturbations can realize. These realizable directions define an intrinsic geometry of objective interaction, within which IGFM constructs a common-descent direction and lifts the desired functional change back into the parameter space. The resulting ideal direction is invariant to locally invertible reparameterizations of the model and positive objective rescaling, and guarantees simultaneous first-order descent for every nonzero objective whenever the realizable directions are not exactly opposed to one another.

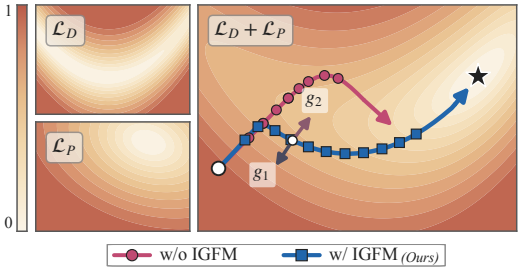


Figure 1: Parameter-space conflict depends on the chosen parameterization. On the same two-objective problem, parameter-space and function-space geometries induce different update trajectories when resolving this objective conflict.

We validate IGFM across three physics-constrained generation tasks: Darcy Flow, Kolmogorov Flow, and Structural Topology Optimization. In all three settings, IGFM improves physical consistency while maintaining generative fidelity, reducing the primary physical error by 40.0%, 22.4%, and 55.7%, respectively, relative to the strongest PBFM baseline. Controlled ablations further isolate the function-space geometry from the normalization, step-length, and lifting mechanics with architecture and objectives fixed, showing that these auxiliary choices alone do not account for the observed improvement in physical consistency under matched training conditions.

Our main contributions span the geometric diagnosis, the method, and its empirical validation:

- We identify a fundamental ambiguity in parameter-gradient-based conflict resolution: function-preserving reparameterizations can substantially alter the measured conflict, and therefore the function-space motion the rule induces, without changing the represented function itself.
- We introduce IGFM, which defines objective conflict over realizable first-order changes in function space and constructs a scale-free common-descent direction in this intrinsic geometry.
- We establish its invariance and common-descent properties and validate the framework through reparameterization audits and three physics-constrained generation tasks spanning PDE residuals, nonlinear fluid dynamics, and downstream engineering design with held-out evaluation.

2 Related Work

Physics-Constrained Generative Modeling. Physics-informed generative models incorporate PDE residuals, feasibility constraints, or differentiable physical objectives into diffusion- or flow-based models, with physics entering during training, through the model architecture, or through sampling and post-hoc correction (Tauberschmidt et al., 2026; Christopher et al., 2024). PBFM (Baldan et al., 2026) is the closest formulation to ours because it treats distribution matching and physical consistency as simultaneous training objectives. IGFM adopts this two-objective setting but defines their interaction through realizable changes of the model output rather than Euclidean parameter gradients. FunDiff (Wang et al., 2026) formulates physics-informed diffusion directly over random functions and fields, while functional flow matching (Kerrigan et al., 2024) defines the matching flow in an infinite-dimensional function space. These works use function space to model the generated random function or physical field, whereas IGFM uses the realizable function-space tangent of a parameterized model to characterize multi-objective interaction. Physics-informed neural networks embed governing equations through residual losses (Raissi et al., 2019; Wang et al., 2021), and the

difficulty of optimizing their competing loss terms (Krishnapriyan et al., 2021; Wang et al., 2022; Xiang et al., 2022) illustrates a closely related optimization tension in physics-informed learning.

Multi-Objective Gradient Geometry. Multi-objective methods combine, project, or normalize task gradients: MGDA (Sener & Koltun, 2018) provides the classical minimum-norm formulation, while ConFIG (Liu et al., 2025) constructs a normalized common-descent direction from conflicting parameter gradients, with PBFM (Baldan et al., 2026) applying this construction to physics-constrained flow matching. The Pareto construction underlying IGFM follows the classical multi-criteria steepest-descent view (Fliege & Svaiter, 2000). Existing methods formulate this geometry directly over parameter gradients; IGFM instead represents each objective differential by its realizable first-order change in function space before constructing the common-descent direction. This anchors the geometry of objective interaction to realizable changes of the represented model rather than to a particular parameter chart. A broader family of multi-task methods reweights or projects parameter gradients—GradNorm (Chen et al., 2018), PCGrad (Yu et al., 2020), CAGrad (Liu et al., 2021a), and Nash-MTL (Navon et al., 2022)—and loss balancing within physics-informed learning has likewise been formulated as multi-objective optimization (Bischof & Kraus, 2021; Hwang & Lim, 2024); these methods predominantly formulate objective interaction through parameter-space gradients, leaving the geometry of that interaction tied to the chosen coordinates.

Parameterization-Invariant Learning. Parameterization-invariant optimization has a long history: Path-SGD is invariant to node-wise rescalings of ReLU networks (Neyshabur et al., 2015), while natural-gradient methods replace the Euclidean parameter metric with a distribution-induced geometry (Amari, 1998; Martens & Grosse, 2015) and Jacobian descent generalizes gradient descent to vector-valued objectives while aggregating in parameter space (Quinton & Rey, 2024). Recent work has also parameterized neural models in function space (Scannell et al., 2024) or characterized learning rates through induced output changes (Milsom et al., 2025). Related function-level perspectives include Sobolev training, which supervises derivatives of the represented function (Czarnecki et al., 2017). IGFM is related to these ideas through its $J^\top M J$-type pullback structure, but uses the resulting realizable function-space geometry to characterize the interaction between multiple objectives and construct their joint descent direction. Further related work and geometric connections, including an explicit contrast with geometric optimizers, are discussed in Appendix H.

3 Method

3.1 Physics-Constrained Flow Matching

We build on flow matching (Lipman et al., 2023; Tong et al., 2024) and the two-objective formulation of PBFM (Baldan et al., 2026). Let $x_0, x_1 \in \mathbb{R}^{d_x}$ denote source and data samples drawn from p_0 and p_{data}, and let $t \in [0, 1]$. For the linear probability path,

$$x_t = (1 - t)x_0 + tx_1, \qquad u_t = x_1 - x_0, \tag{1}$$

the velocity field $v_\theta(x, t)$ minimizes

$$\mathcal{L}_D(\theta) = \mathbb{E}_{x_0, x_1, t}\left[\|v_\theta(x_t, t) - u_t\|_2^2\right]. \tag{2}$$

Following PBFM, physical consistency is evaluated near the generated endpoint by unrolling the learned velocity from x_t to an n-step approximation $\hat{x}_1^{(n)}$ (Baldan et al., 2026). For a differentiable physical functional Φ_P and time weight $w(t) = t^\beta$,

$$\mathcal{L}_P(\theta) = \mathbb{E}\left[w(t)\,\Phi_P\left(\hat{x}_1^{(n)}\right)\right]. \tag{3}$$

For the PDE tasks, Φ_P is the task's discretized residual on the endpoint field, and Structural Topology Optimization instead uses the differentiable log-compliance with its volume term (Appendix E.4), with per-task forms in Appendix E.1. All tasks use the linear time weight $w(t) = t$ ($\beta = 1$), applied identically to every physics-constrained method compared. IGFM leaves both objectives unchanged and instead changes how their interaction is represented during optimization.

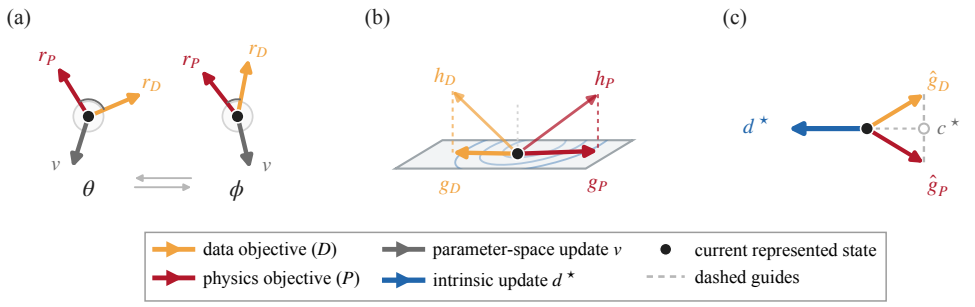


Figure 2: Intrinsic Geometry Flow Matching. (a) Equivalent charts induce different parameter-gradient geometries. (b) IGFM instead represents each objective by its realizable tangent component. (c) The common-descent direction is constructed from these intrinsic representatives.

3.2 Chart Dependence of Parameter-Space Conflict

For objective $i \in \{D, P\}$, let $r_i := \nabla_\theta \mathcal{L}_i(\theta)$. Under a first-order update $\theta^+ = \theta - \eta v$ with step size η and direction v, descent of objective i is determined by $r_i^\top v$, so parameter-space methods such as ConFIG (Liu et al., 2025) construct the update from the geometry of r_D and r_P, and a common alignment statistic is their Euclidean cosine γ_θ. A meaningful measure of objective interaction, however, should not change when only the coordinates used to represent the same model are changed. This requirement fails. Under a locally invertible reparameterization $\theta = \psi(\phi)$ with Jacobian $Q := D\psi(\phi)$, objective gradients transform as

$$r_i^{(\phi)} = Q^\top r_i^{(\theta)}, \tag{4}$$

and the Euclidean cosine is generally not preserved. More strongly, for any two nonzero linearly independent gradients and any target value $\bar{\gamma} \in (-1, 1)$, an invertible local chart can be chosen such that $\cos(Q^\top r_1, Q^\top r_2) = \bar{\gamma}$; a constructive derivation, together with the degenerate cases, is given in Appendix A.2. Parameter-gradient alignment is therefore a property of the chosen coordinates rather than of the represented function, so optimization rules that depend on this geometry can yield different decisions under equivalent parameterizations. Conflict should instead be defined from realizable changes of the model output rather than from the coordinates used to represent it.

3.3 Intrinsic Geometry Flow Matching

Realizable Function-Space Tangent. A change of parameter coordinates can alter the basis used to describe parameter perturbations, but it cannot alter the set of first-order output motions that the model can realize, once the observation map and output metric are fixed. We therefore build the geometry on this set. For a minibatch B, let $\mathcal{O}_B : \Theta \to \mathcal{H}_B$ collect the network quantities on which the two objectives depend: the flow-model velocity output on the minibatch and, through it, the unrolled endpoint and the physical quantities derived from that endpoint. We equip the observation space with the inner product $\langle a, b \rangle_M := a^\top M b$ for a fixed $M \succ 0$, and use normalized observation coordinates with $M = I$ in all experiments, each dataset standardized by fixed per-channel training-set statistics; the task-specific components of $\mathcal{O}_B$ are listed in Appendix E. At $y := \mathcal{O}_B(\theta)$ with Jacobian $J := D\mathcal{O}_B(\theta)$, the first-order output motions realizable by parameter perturbations form

$$T_\theta := \mathrm{range}(J) \subseteq \mathcal{H}_B. \tag{5}$$

This output-motion viewpoint is related to recent analyses that quantify neural-network updates by their induced change in function space rather than by parameter displacement (Milsom et al., 2025); here, the realizable tangent defines the geometry of multi-objective conflict.

Realizable Objective Gradients. The ambient output-space gradient of an objective can contain components that the current network cannot realize through any infinitesimal parameter change. Writing $\mathcal{L}_i(\theta) = \widetilde{\mathcal{L}}_i(y)$, let $h_i \in \mathcal{H}_B$ denote its output-space gradient, defined by $d\widetilde{\mathcal{L}}_i(y)[\delta y] = \langle h_i, \delta y \rangle_M$. Because h_i lives in the full observation space, directly comparing such gradients would assign geometric meaning to unattainable output motions. We therefore project each differential onto the realizable tangent,

$$g_i := P_{T_\theta}^M h_i, \tag{6}$$

and use g_i as the representation of objective i. The usual parameter gradient is $r_i = J^\top M h_i$; the projection shows that it appears only as the right-hand side of a tangent solve, whose exact form, normal equations, and rank-deficient cases are given in Appendix A.3. The alignment of the two objectives can be summarized by the intrinsic cosine γ_T of g_D and g_P—a chart-invariant tangent-space alignment statistic, unchanged by a locally invertible reparameterization because such a change preserves $\mathrm{range}(J)$; this statistic is used in the chart audit of Section 4.2.

Intrinsic Pareto Direction. Normalize $\hat{g}_i := g_i / \|g_i\|_M$. For two objectives, the classical minimum-norm multicriteria construction (Fliege & Svaiter, 2000) reduces to

$$d^\star = -\frac{\hat{g}_D + \hat{g}_P}{2}. \tag{7}$$

The corresponding formulation over the convex hull of the normalized realizable gradients is given in Appendix A.4. This normalized-bisector form is also used by ConFIG (Liu et al., 2025) and PBFM when operating on parameter gradients. The distinction is in the vectors entering the construction: (r_D, r_P) in parameter space versus the representatives (g_D, g_P) in the realizable tangent. Figure 2 summarizes the construction of the intrinsic direction from these representatives.

Properties. For a fixed observation map $\mathcal{O}_B$ and metric M, locally invertible reparameterizations preserve $\mathrm{range}(J)$ and therefore leave the realizable gradients g_i and the ideal direction $d^\star$ unchanged. Normalization makes the direction invariant to independent positive rescaling and permutation of the objectives. Whenever the two objectives are not exactly opposed in the tangent, $d^\star$ is a first-order common-descent direction for both, with equal normalized descent margins. Complete formal statements, proofs, and degenerate cases are given in Appendix A.4.

3.4 Practical Realization

Training uses a finite-dimensional approximation of the ideal field, implemented by two damped tangent solves that realize the projected representatives as parameter-space lifts. Unless stated otherwise, IGFM uses conjugate-gradient (CG) depth $k = 5$ and tangent damping 10^{-3} on the curvature scale of the data gradient. For each objective,

$$\left(J^\top M J + \lambda I\right) \xi_i = r_i, \qquad \widetilde{g}_i := J \xi_i, \tag{8}$$

and we use $\widetilde{g}_i$ in place of g_i to construct the practical intrinsic direction. The composite direction is assembled directly from the two lifts, and a projected step-length rule scales it by the mean projection of the objective gradients onto it; the scaled direction is stored as the parameter gradient, so that the AdamW step realizes the descent with optimizer state carried across steps. The combination rule, step-length rule, and optimizer interface are specified in Appendix G. Products with $J^\top M J$ are computed through Jacobian–vector (JVP) and vector–Jacobian (VJP) products, so neither J nor the tangent projector is formed explicitly. The experiments evaluate this fixed-depth realization; its numerical behavior under finite depth, damping, optimizer state, and step sizes is analyzed in Appendices C and D, together with a direction-fidelity audit against a high-accuracy reference solve, which the realized direction matches to cosine above 0.998 at every audited stage.

4 Experiments

In this section, we apply IGFM to three physics-constrained generation tasks—Darcy Flow with residual-based PDE constraints, Kolmogorov Flow as a nonlinear fluid-dynamics setting, and Structural Topology Optimization as a downstream engineering objective. Unless stated otherwise, IGFM denotes the fixed-depth realization with $k = 5$ and damping 10^{-3} of Section 3, trained with the optimized implementation of Section 4.5. For each task, every reported method is averaged over five training runs with different seeds. Per-seed values, standard deviations, and complete training and evaluation protocols are given in Appendix E. An ablation study (Appendix D) shows the gains are robust to the tangent-solve depth, damping, and objective rescalings, so they do not rely on tuning beyond these defaults; the same configuration serves all three tasks without per-task adjustment.

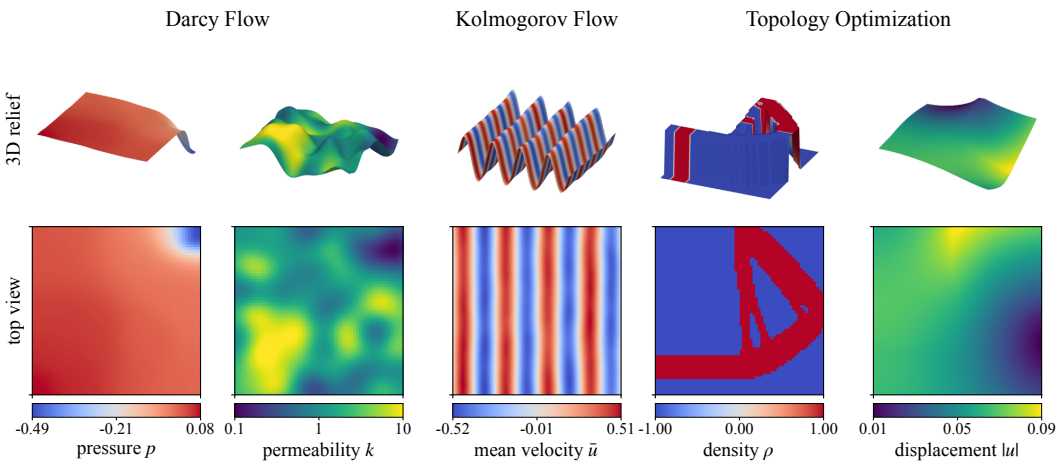


Figure 3: Benchmark overview by reference quantity: Darcy pressure p and permeability k (log color scale), Kolmogorov mean velocity $\bar{u}$, and Topology Optimization density ρ with displacement magnitude $|u|$, as 3D reliefs (top) and top views (bottom) of ground-truth samples.

4.1 Experimental Setup

All models are trained on a single NVIDIA H100 80GB GPU, and evaluated from frozen checkpoints on held-out physical conditions with deterministic sampling. Within each task, FM-OT (Tong et al., 2024), PBFM (Baldan et al., 2026), and IGFM use the same backbone, data split, physical functional, optimizer schedule, and evaluator; PBFM-4 denotes PBFM with the original unrolling curriculum, which ramps the unrolled physics steps from one to four over training (Baldan et al., 2026), and PBFM-1 the single-step variant used as the wall-clock reference.

Darcy Flow measures physical consistency with the discretized governing-equation residual R and distribution fidelity with the 2-Wasserstein distance W_2 (Villani et al., 2009). Kolmogorov Flow reports the endpoint divergence residual R_{div} together with the spectrum error. Structural Topology Optimization replaces the residual objective with the log-compliance objective of density-based topology optimization (Bendsøe et al., 2004), evaluated through compliance error E_C, volume violation ΔV, and topology Fréchet distance (FD) under held-out loads and boundary conditions; the task formulation is specified in Appendix E.4. Figure 3 overviews the benchmark through the reference quantity of each task, and all metrics are computed by frozen task-specific evaluators applied identically to every method within each task, for both the primary errors and the fidelity metrics.

4.2 Reparameterization Consistency

We first test the geometric distinction independently of downstream task performance: across 7040 function-preserving reparameterizations of the trained Darcy model, the parameter-space cosine spans 0.40–0.93, whereas the intrinsic statistic—evaluated on matched reparameterizations at condition numbers $\kappa(Q) \in \{100, 1000, 3000\}$—remains within 0.3093–0.3122, and the function-identity error stays below 6.3×10^{-5}. Figure 4 condenses this comparison: the median parameter-space change exceeds the matched intrinsic change by one to two orders of magnitude at every audited condition number; the restricted condition range reflects the costlier intrinsic audit procedure.

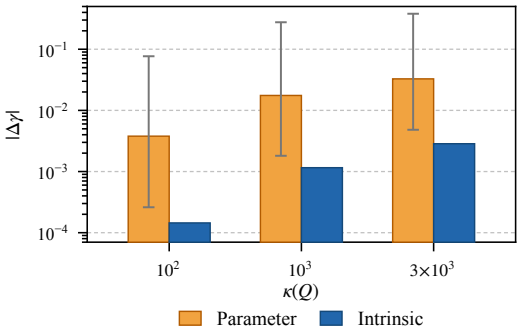


Figure 4: Chart audit on the trained Darcy model: the median parameter-space change $|\Delta\gamma_\theta|$ (orange, whiskers 10–90) varies across charts, while the intrinsic $|\Delta\gamma_T|$ (blue) stays stable.

The same distinction appears during optimization: under the default fixed-depth realization,

Table 1: Physics–distribution operating points on Darcy Flow. Mean ± std. across five seeds; lower is better, best in bold.

Method	$R \downarrow$	$W_2 \downarrow$
FM-OT	35.46±2.10	**0.012±0.001**
$\omega = 10^{-2}$	28.35±1.84	0.013±0.001
$\omega = 1$	19.42±1.15	0.019±0.002
$\omega = 10^{2}$	14.12±1.63	0.031±0.004
MGDA	35.28±1.42	0.013±0.001
ConFIG-VJP	16.13±0.95	0.021±0.002
PBFM-1	18.42±1.20	0.018±0.002
PBFM-4	11.55±0.57	0.024±0.003
IGFM	**6.93±1.24**	0.014±0.002

Table 2: Component ablation and matched control on Darcy Flow. Mean ± std. across five seeds; lower is better, best in bold; italic headers group the variants.

Variant	$R \downarrow$	$W_2 \downarrow$
Representation & Geometry		
ConFIG-VJP (parameter gradients)	16.13±0.95	0.021±0.002
Ambient gradients (h_D, h_P)	19.24±2.11	0.018±0.003
Mechanics Controlled		
Matched Parameter-Space Control (step & norm.)	14.20±1.05	0.019±0.002
Step realization		
w/o normalization	12.45±1.18	0.019±0.002
w/o step-length rule	9.82±0.88	0.016±0.002
Full IGFM (g_D, g_P)	**6.93±1.24**	**0.014±0.002**

equivalent-chart twin runs—initialized from the same represented function and sharing the data order, optimizer settings, and sampling stream—diverge less under IGFM than under PBFM-1 over the evaluated horizon. Full diagnostics and the matched intrinsic audits are reported in Appendix B.

4.3 Isolating Intrinsic Geometry on Darcy Flow

Table 1 reports the operating points, and Figure 5 places them in the joint physics–distribution plane. Weighted scalarization, trained with $\mathcal{L}_D + \omega\mathcal{L}_P$, trades physical consistency against distribution fidelity, with MGDA (Sener & Koltun, 2018) near the unconstrained operating point. PBFM and ConFIG-VJP (Liu et al., 2025) reduce the physical residual at increased distribution error, whereas IGFM reaches $R = 6.93$ while retaining $W_2 = 0.014$, a further 40% below PBFM-4.

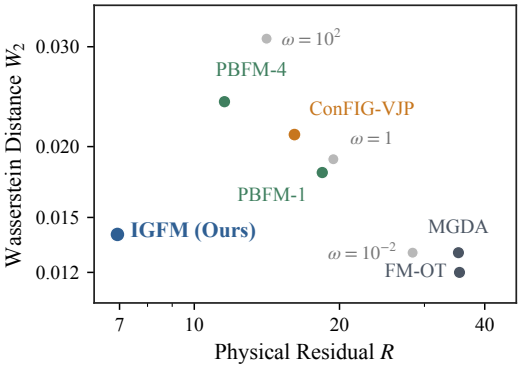


Figure 5: Physics–distribution operating plane on Darcy Flow (five-seed means; lower is better). IGFM reaches the lowest residual at distribution fidelity close to that of the FM-OT reference.

To isolate the contribution of the intrinsic representation from auxiliary numerical mechanics, we compare IGFM with ConFIG-VJP and a *Matched Parameter-Space Control*. The matched control applies the numerical mechanics specified by IGFM—the same unit-norm normalization, bisector combination, and projected step-length rule—to the parameter gradients (r_D, r_P) in place of the tangent representatives; since the substituted directions already live in parameter space, no tangent solve is performed and the lift degenerates to the identity, so the comparison isolates the effect of the representative space under the specified matched update mechanics. The first row of Table 2 is this ConFIG-VJP entry, identical to the corresponding row of Table 1. As shown in Table 2, these mechanics improve the parameter-space control, but the result remains substantially above Full IGFM; removing either unit-norm normalization or the projected step-length rule yields intermediate operating points. Together, these controls show the auxiliary numerical components contribute to optimization behavior but do not by themselves account for the full improvement of the realizable tangent representation under these matched training conditions.

4.4 Transfer Across Objectives

Table 3 summarizes the quantitative results across the three tasks. Figure 6 provides qualitative evidence through Darcy residual maps, Kolmogorov velocity fields, and held-out topology designs,

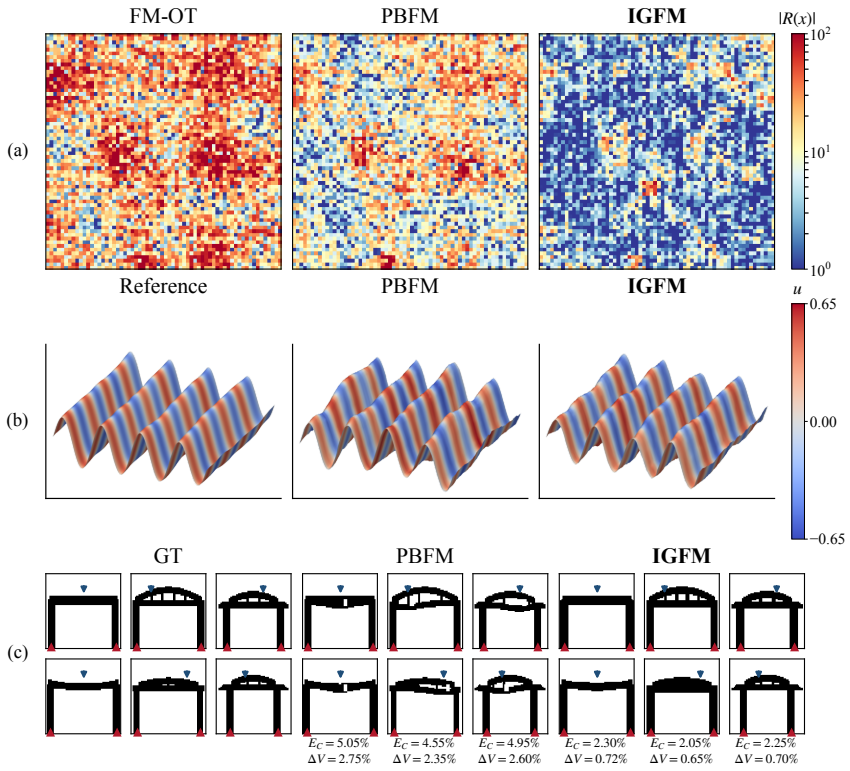


Figure 6: Cross-task qualitative comparison. (a) Darcy pointwise residual $|R(x)|$ (mean over PDE components) at one held-out sample for all methods, on a shared log color scale. (b) Kolmogorov mean-velocity fields at one held-out flow condition as 3D reliefs under a shared color range. (c) Topology Optimization designs for the six held-out bridge-type load cases (triangles: supports; arrow: load), annotated with E_C and ΔV for each compared method.

Table 3: Main results across three physics-constrained generation tasks. Values are mean $\pm$ standard deviation across five training seeds per method; the Kolmogorov spectrum error is reported as an aggregate mean. Lower is better, best in bold; FM-OT is the unconstrained baseline for all three tasks, with Topology Optimization evaluated under held-out loads and boundary conditions.

	Darcy Flow		**Kolmogorov Flow**		**Topology Optimization**		
Method	R	W_2	R_{div}	Spec. Err.	E_C	ΔV	FD
FM-OT	35.46±2.10	**0.012±0.001**	14.28±1.15	5.10×10^{-4}	8.57±0.90%	4.20±0.30%	**1.42±0.11**
PBFM-1	18.42±1.20	0.018±0.002	8.92±0.85	3.15×10^{-4}	5.12±0.60%	2.80±0.20%	1.85±0.15
PBFM-4	11.55±0.57	0.024±0.003	5.81±1.64	2.19×10^{-4}	4.85±0.50%	2.50±0.20%	2.10±0.18
IGFM	**6.93±1.24**	0.014±0.002	**4.51±0.49**	$\mathbf{2.16\times10^{-4}}$	**2.15±0.30%**	**0.60±0.10%**	1.48±0.12

while Figure 7 reports the relative reduction in each primary error. Across all three settings, the reduction in primary physical error is accompanied by fidelity close to or better than FM-OT and better than both PBFM baselines on the reported fidelity metrics of each task.

Darcy Flow. In the pointwise residual maps of Figure 6(a), the IGFM field is substantially flatter than those of FM-OT and PBFM under the shared log color scale; the full field grids and the complete per-sample residual distributions are reported in Figures 13 and 14 of Appendix F.

Kolmogorov Flow. IGFM improves the divergence residual with nearly unchanged spectrum error, extending the gain from the elliptic Darcy setting to nonlinear fluid dynamics. In Figure 6(b), the PBFM field overshoots the reference extrema despite a similar spectrum error, whereas the IGFM field more closely matches the reference amplitude while retaining its coherent large-scale structure; the per-condition velocity profiles and per-sample fields are reported in Figure 17.

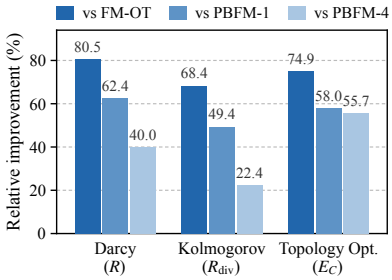


Figure 7: IGFM improvement over each baseline on the primary error: R (Darcy), R_{div} (Kolmogorov), E_C (Topology Optimization); 50% means IGFM halves the baseline error.

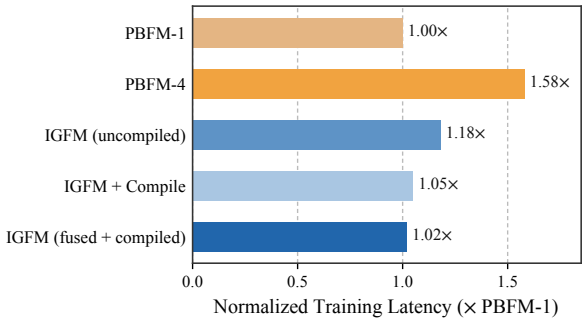


Figure 8: Training latency of the k=5 realization and its optimized variants on a single H100, normalized by the PBFM-1 wall-clock reference; the fused-and-compiled configuration used for all main experiments is within 2% of this reference (Appendix G.2).

Structural Topology Optimization. The primary engineering error is the compliance error E_C, reduced by 55.7% relative to the strongest PBFM baseline, while the topology Fréchet distance stays close to the unconstrained FM-OT reference (1.48 vs. 1.42; Table 3). In the held-out bridge-type designs of Figure 6(c), the PBFM examples show characteristic defects—a sagging deck, a missing hanger, and a spurious hole—whereas the IGFM designs stay close to the reference layout (equilibrium residuals for all compared methods in Figure 19 of Appendix F).

Long-Horizon Stability. We extend both PDE schedules by 50% to examine late-stage behavior. IGFM remains stable over the extended horizon, with the Darcy residual still decreasing throughout the final 30% of training without a late-stage reversal (trajectories in Appendix E.6).

4.5 Computational Cost

TorchCompile and a custom fused kernel reduce JVP/VJP overhead. On one H100 (Figure 8), the unoptimized k=5 realization costs $1.18\times$ PBFM-1 at 28.6 GB, and the fused-and-compiled combination reaches $1.02\times$ at 23.5 GB, below PBFM-4 ($1.58\times$, 29.3 GB); unoptimized latency grows from $1.06\times$ at k=1 to $1.27\times$ at k=8. Timing and memory audits appear in Appendix G.2.

5 Discussion and Conclusion

Discussion. Parameter-gradient conflict varies across function-preserving reparameterizations; the realizable tangent is stable across them. Matched parameter-space controls show that these mechanics alone do not explain the full improvement, and one solver configuration serves all three tasks.

Limitations and Outlook. The invariance result is local and first-order, relative to a fixed observation map and output metric: changing either induces a different geometry, and the finite-depth, damped solve does not preserve the ideal trajectories. The closed-form construction and all experiments target the two-objective setting; the many-objective extension (Appendix A.4) is not evaluated empirically. Ablations under one solver configuration suggest limited sensitivity to the numerical choices and the observation metric (Appendix D); the observation map itself is not varied, and adaptive observation maps, alternative metrics, and cheaper solvers are left to future work.

Conclusion. We presented IGFM, addressing the chart dependence of parameter-gradient conflict by defining objective interaction through realizable function-space changes. Invariance to reparameterization and objective rescaling, and scale-free common descent, are properties of the ideal direction; the fixed-depth realization approximates it with one solver configuration, within 2% of the single-step reference. Across the three tasks, IGFM reduces the primary physics or engineering error by 22.4–55.7% over the strongest PBFM baseline at comparable or improved fidelity, supporting realizable function-space geometry as a basis for physics-constrained generative modeling.

AI USE STATEMENT

Generative AI tools were used solely to assist with drafting, structural refining, and language polishing of this manuscript. All conceptual framework development, experimental designs, code implementations, data processing, and figure generation were conducted entirely by the authors. All AI-assisted text was thoroughly reviewed, critically evaluated, and revised by the authors, who retain full and complete responsibility for the entire content of the final manuscript.

REPRODUCIBILITY STATEMENT

All quantitative comparisons use fixed task-specific evaluators and frozen checkpoints, with the same evaluation protocol applied to all methods within each task. Aggregate results are reported as mean $\pm$ standard deviation across five training seeds per method. Darcy and Kolmogorov use deterministic flow sampling, while Topology Optimization evaluates compliance, volume violation, and topology fidelity under matched held-out structural conditions. Per-run bootstrap intervals are used only for diagnostic uncertainty and are kept separate from seed-level variation. The appendix and supplementary material provide per-run provenance, task protocols, evaluation details, theorem proofs, reparameterization studies, ablation sweeps, and finite-step solver diagnostics.

REFERENCES

Shun-ichi Amari. Natural gradient works efficiently in learning. *Neural Comput.*, 10(2):251–276, 1998. doi: 10.1162/089976698300017746. URL https://doi.org/10.1162/089976698300017746.

Giacomo Baldan, Qiang Liu, Alberto Guardone, and Nils Thuerey. Physics vs distributions: Pareto optimal flow matching with physics constraints. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=tAf1KI3d4X.

Jan-Hendrik Bastek and Dennis M. Kochmann. Inverse design of nonlinear mechanical metamaterials via video denoising diffusion models. *Nat. Mac. Intell.*, 5(12):1466–1475, 2023. doi: 10.1038/S42256-023-00762-X. URL https://doi.org/10.1038/s42256-023-00762-x.

Christian Beck, Martin Hutzenthaler, Arnulf Jentzen, and Benno and Kuckuck. An overview on deep learning-based approximation methods for partial differential equations. *Discrete and Continuous Dynamical Systems - B*, 28(6):3697–3746, 2023. ISSN 1553-524X. doi: 10.3934/dcdsb.2022238. URL http://doi.org/10.3934/dcdsb.2022238.

Martin P Bendsøe, Ole Sigmund, et al. *Topology optimization: theory, methods, and applications*, volume 2. Springer Berlin, 2004.

Rafael Bischof and Michael A. Kraus. Multi-objective loss balancing for physics-informed deep learning. *CoRR*, abs/2110.09813, 2021. URL https://arxiv.org/abs/2110.09813.

Steven L. Brunton and J. Nathan Kutz. Promising directions of machine learning for partial differential equations. *Nature Computational Science*, 4(7):483–494, 2024. ISSN 2662-8457. doi: 10.1038/s43588-024-00643-2. URL http://doi.org/10.1038/s43588-024-00643-2.

Markus J. Buehler. Modeling atomistic dynamic fracture mechanisms using a progressive transformer diffusion model. *Journal of Applied Mechanics*, 89(12), October 2022. ISSN 1528-9036. doi: 10.1115/1.4055730. URL http://doi.org/10.1115/1.4055730.

Wenqian Chen, Qian Wang, Jan S. Hesthaven, and Chuhua Zhang. Physics-informed machine learning for reduced-order modeling of nonlinear problems. *J. Comput. Phys.*, 446:110666, 2021. doi: 10.1016/J.JCP.2021.110666. URL https://doi.org/10.1016/j.jcp.2021.110666.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In Jennifer G. Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings*

of Machine Learning Research, pp. 793–802. PMLR, 2018. URL http://proceedings.mlr.press/v80/chen18a.html.

Zhao Chen, Jiquan Ngiam, Yanping Huang, Thang Luong, Henrik Kretzschmar, Yuning Chai, and Dragomir Anguelov. Just pick a sign: Optimizing deep multitask models with gradient sign dropout. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/16002f7a455a94aa4e91cc34ebdb9f2d-Abstract.html.

Jacob K Christopher, Stephen Baek, and Ferdinando Fioretto. Constrained synthesis with projected diffusion models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=FsdB3I9Y24.

Salvatore Cuomo, Vincenzo Schiano Di Cola, Fabio Giampaolo, Gianluigi Rozza, Maziar Raissi, and Francesco Piccialli. Scientific machine learning through physics-informed neural networks: Where we are and what's next. *J. Sci. Comput.*, 92(3):88, 2022. doi: 10.1007/S10915-022-01939-Z. URL https://doi.org/10.1007/s10915-022-01939-z.

Wojciech M. Czarnecki, Simon Osindero, Max Jaderberg, Grzegorz Swirszcz, and Razvan Pascanu. Sobolev training for neural networks. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 4278–4287, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/758a06618c69880a6cee5314ee42d52f-Abstract.html.

Taco de Wolff, Hugo Carrillo Lincopi, Luis Martí, and Nayat Sánchez-Pi. Mopinns: an evolutionary multi-objective approach to physics-informed neural networks. In Jonathan E. Fieldsend and Markus Wagner (eds.), *GECCO '22: Genetic and Evolutionary Computation Conference, Companion Volume, Boston, Massachusetts, USA, July 9 - 13, 2022*, pp. 228–231. ACM, 2022. doi: 10.1145/3520304.3529071. URL https://doi.org/10.1145/3520304.3529071.

Christian Düreth, Paul Seibert, Dennis Rücker, Stephanie Handford, Markus Kästner, and Maik Gude. Conditional diffusion-based microstructure reconstruction. *Materials Today Communications*, 35:105608, 2023. ISSN 2352-4928. doi: 10.1016/j.mtcomm.2023.105608. URL http://doi.org/10.1016/j.mtcomm.2023.105608.

Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11, 2018. doi: 10.1016/J.NEUNET.2017.12.012. URL https://doi.org/10.1016/j.neunet.2017.12.012.

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

Jörg Fliege and Benar Fux Svaiter. Steepest descent methods for multicriteria optimization. *Math. Methods Oper. Res.*, 51(3):479–494, 2000. doi: 10.1007/S001860000043. URL https://doi.org/10.1007/s001860000043.

Stefania Fresca, Luca Dedè, and Andrea Manzoni. A comprehensive deep learning-based approach to reduced order modeling of nonlinear time-dependent parametrized pdes. *J. Sci. Comput.*, 87(2):61, 2021. doi: 10.1007/S10915-021-01462-7. URL https://doi.org/10.1007/s10915-021-01462-7.

Han Gao, Matthew J. Zahr, and Jian-Xun Wang. Physics-informed graph neural galerkin networks: A unified framework for solving pde-governed forward and inverse problems. *Computer Methods in Applied Mechanics and Engineering*, 390:114502, February 2022. ISSN 0045-7825. doi: 10.1016/j.cma.2021.114502. URL http://doi.org/10.1016/j.cma.2021.114502.

Giorgio Giannone, Akash Srivastava, Ole Winther, and Faez Ahmed. Aligning optimization trajectories with diffusion models for constrained design generation. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/a2fe4bb50fc6f3564cee1551d6309fea-Abstract-Conference.html.

Luca Guastoni and Ricardo Vinuesa. A new perspective on the simulation of stochastic problems in fluid mechanics with diffusion models. *Nature Machine Intelligence*, 7(6):816–817, 2025. ISSN 2522-5839. doi: 10.1038/s42256-025-01060-4. URL http://doi.org/10.1038/s42256-025-01060-4.

Jiequn Han, Arnulf Jentzen, and Weinan E. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*, 115(34):8505–8510, August 2018. ISSN 1091-6490. doi: 10.1073/pnas.1718942115. URL http://doi.org/10.1073/pnas.1718942115.

Benjamin J. Holzschuh, Simona Vegetti, and Nils Thuerey. Solving inverse physics problems with score matching. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/c2f2230abc7ccf669f403be881d3ffb7-Abstract-Conference.html.

Jiahe Huang, Guandao Yang, Zichen Wang, and Jeong Joon Park. DiffusionPDE: Generative PDE-solving under partial observation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024a. URL https://openreview.net/forum?id=z0I2SbjN0R.

William Huang, Yifeng Jiang, Tom Van Wouwe, and C. Karen Liu. Constrained diffusion with trust sampling. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024b. URL http://papers.nips.cc/paper_files/paper/2024/hash/aa8164b8a8e484fe31885657fea14997-Abstract-Conference.html.

Youngsik Hwang and Dong-Young Lim. Dual cone gradient descent for training physics-informed neural networks. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/b2b781badeeb49896c4b324c466ec442-Abstract-Conference.html.

Christian Jacobsen, Yilin Zhuang, and Karthik Duraisamy. Cocogen: Physically consistent and conditioned score-based generative models for forward and inverse problems. *SIAM Journal on Scientific Computing*, 47(2):C399–C425, April 2025. ISSN 1095-7197. doi: 10.1137/24m1636071. URL http://doi.org/10.1137/24M1636071.

Adrián Javaloy and Isabel Valera. Rotograd: Gradient homogenization in multitask learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=T8wHz4rnuGL.

George Karniadakis and Spencer Sherwin. *Spectral/hp Element Methods for Computational Fluid Dynamics*. Oxford University Press, 06 2005. ISBN 9780198528692. doi: 10.1093/acprof:oso/9780198528692.001.0001. URL https://doi.org/10.1093/acprof:oso/9780198528692.001.0001.

Gavin Kerrigan, Giosue Migliorini, and Padhraic Smyth. Functional flow matching. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li (eds.), *International Conference on Artificial Intelligence and Statistics, 2-4 May 2024, Palau de Congressos, Valencia, Spain*, volume 238 of *Proceedings of Machine Learning Research*, pp. 3934–3942. PMLR, 2024. URL https://proceedings.mlr.press/v238/kerrigan24a.html.

Shervin Khalafi, Dongsheng Ding, and Alejandro Ribeiro. Constrained diffusion models via dual training. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/2eb31e84089286505c27dd5d5ea7f683-Abstract-Conference.html.

Aditi S. Krishnapriyan, Amir Gholami, Shandian Zhe, Robert M. Kirby, and Michael W. Mahoney. Characterizing possible failure modes in physics-informed neural networks. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 26548–26560, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/df438e5206f31600e6ae4af72f2725f1-Abstract.html.

Tianyi Li, Alessandra S. Lanotte, Michele Buzzicotti, Fabio Bonaccorso, and Luca Biferale. Multi-scale reconstruction of turbulent rotating flows with generative diffusion models. *Atmosphere*, 15 (1):60, December 2023. ISSN 2073-4433. doi: 10.3390/atmos15010060. URL http://doi.org/10.3390/atmos15010060.

Xinpeng Li, Enming Liang, and Minghua Chen. Gauge flow matching: Efficient constrained generative modeling over general convex set and beyond. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=vxq1OnaAMq.

Marten Lienen, David Lüdke, Jan Hansen-Palmus, and Stephan Günnemann. From zero to turbulence: Generative modeling for 3d flow simulation. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=Zh1woC1XaN.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 18878–18890, 2021a. URL https://proceedings.neurips.cc/paper/2021/hash/9d27fdf2477ffbff837d73ef7ae23db9-Abstract.html.

Bo Liu, Yihao Feng, Peter Stone, and Qiang Liu. FAMO: fast adaptive multitask optimization. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/b2fe1ee8d936ac08dd26f2ff58986c8f-Abstract-Conference.html.

Liyang Liu, Yi Li, Zhanghui Kuang, Jing-Hao Xue, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. Towards impartial multi-task learning. In *International Conference on Learning Representations*, 2021b. URL https://openreview.net/forum?id=IMPnRXEWpvr.

Qiang Liu and Nils Thuerey. Uncertainty-aware surrogate models for airfoil flow simulations with denoising diffusion probabilistic models. *AIAA Journal*, 62(8):2912–2933, August 2024. ISSN 1533-385X. doi: 10.2514/1.j063440. URL http://doi.org/10.2514/1.J063440.

Qiang Liu, Mengyu Chu, and Nils Thuerey. ConFIG: Towards conflict-free training of physics informed neural networks. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=APojAzJQiq.

James Martens and Roger B. Grosse. Optimizing neural networks with kronecker-factored approximate curvature. In Francis R. Bach and David M. Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pp. 2408–2417. JMLR.org, 2015. URL http://proceedings.mlr.press/v37/martens15.html.

François Mazé and Faez Ahmed. Diffusion models beat gans on topology optimization. In Brian Williams, Yiling Chen, and Jennifer Neville (eds.), *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pp. 9108–9116. AAAI Press, 2023. doi: 10.1609/AAAI.V37I8.26093. URL https://doi.org/10.1609/aaai.v37i8.26093.

Levi D. McClenny and Ulisses M. Braga-Neto. Self-adaptive physics-informed neural networks. *J. Comput. Phys.*, 474:111722, 2023. doi: 10.1016/J.JCP.2022.111722. URL https://doi.org/10.1016/j.jcp.2022.111722.

Edward Milsom, Ben Anson, and Laurence Aitchison. Function-space learning rates. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/milsom25a.html.

Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 16428–16446. PMLR, 2022. URL https://proceedings.mlr.press/v162/navon22a.html.

Behnam Neyshabur, Ruslan Salakhutdinov, and Nathan Srebro. Path-sgd: Path-normalized optimization in deep neural networks. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pp. 2422–2430, 2015. URL https://proceedings.neurips.cc/paper/2015/hash/eaa32c96f620053cf442ad32258076b9-Abstract.html.

Zhenguo Nie, Tong Lin, Haoliang Jiang, and Levent Burak Kara. Topologygan: Topology optimization using generative adversarial networks based on physical fields over the initial domain. *Journal of Mechanical Design*, 143(3):031715, 2021.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 8024–8035, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pp. 4172–4182. IEEE, 2023. doi: 10.1109/ICCV51070.2023.00387. URL https://doi.org/10.1109/ICCV51070.2023.00387.

Pierre Quinton and Valérian Rey. Jacobian descent for multi-objective optimization. *CoRR*, abs/2406.16232, 2024. doi: 10.48550/ARXIV.2406.16232. URL https://doi.org/10.48550/arXiv.2406.16232.

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. *J. Comput. Phys.*, 378:686–707, 2019. doi: 10.1016/J.JCP.2018.10.045. URL https://doi.org/10.1016/j.jcp.2018.10.045.

Maximilian Rixner and Phaedon-Stelios Koutsourelakis. A probabilistic generative model for semi-supervised training of coarse-grained surrogates and enforcing physical constraints through virtual observables. *J. Comput. Phys.*, 434:110218, 2021. doi: 10.1016/J.JCP.2021.110218. URL https://doi.org/10.1016/j.jcp.2021.110218.

M. Sardar, A. Skillen, M. J. Zimoń, S. Draycott, and A. Revell. Spectrally decomposed denoising diffusion probabilistic models for generative turbulence super-resolution. *Physics of Fluids*, 36 (11), November 2024. ISSN 1089-7666. doi: 10.1063/5.0231664. URL http://doi.org/10.1063/5.0231664.

Aidan Scannell, Riccardo Mereu, Paul Edmund Chang, Ella Tamir, Joni Pajarinen, and Arno Solin. Function-space parameterization of neural networks for sequential learning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=2dhxxIKhqz.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 525–536, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/432aca3a1e345e339f35a30c8f65edce-Abstract.html.

Yeonjong Shin, Jérôme Darbon, and George Em Karniadakis. On the convergence of physics informed neural networks for linear second-order elliptic and parabolic type pdes. *Communications in Computational Physics*, 28(5):2042–2074, 2020. ISSN 1991-7120. doi: 10.4208/cicp.oa-2020-0193. URL http://doi.org/10.4208/cicp.OA-2020-0193.

Dule Shu, Zijie Li, and Amir Barati Farimani. A physics-informed diffusion model for high-fidelity flow field reconstruction. *J. Comput. Phys.*, 478:111972, 2023. doi: 10.1016/J.JCP.2023.111972. URL https://doi.org/10.1016/j.jcp.2023.111972.

Gurjeet Sangra Singh, Frantzeska Lavda, Giangiacomo Mercatali, and Alexandros Kalousis. Variational grey-box dynamics matching. In *The 29th International Conference on Artificial Intelligence and Statistics*, 2026. URL https://openreview.net/forum?id=NMuUPLBc84.

Jan Tauberschmidt, Sophie Fellenz, Sebastian Josef Vollmer, and Andrew B. Duncan. Physics-constrained fine-tuning of flow-matching models for generation and inverse problems. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=khBHJz2wcV.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Trans. Mach. Learn. Res.*, 2024, 2024. URL https://openreview.net/forum?id=CD9Snc73AW.

Mario Lino Valencia, Tobias Pfaff, and Nils Thuerey. Learning distributions of complex fluid simulations with diffusion graph networks. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=uKZdlihDDn.

Remco van der Meer, Cornelis W. Oosterlee, and Anastasia Borovykh. Optimally weighted loss functions for solving pdes with neural networks. *J. Comput. Appl. Math.*, 405:113887, 2022. doi: 10.1016/J.CAM.2021.113887. URL https://doi.org/10.1016/j.cam.2021.113887.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Cédric Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2009.

Nikolaos N. Vlassis and WaiChing Sun. Denoising diffusion algorithm for inverse design of microstructures with fine-tuned nonlinear material properties. *Computer Methods in Applied Mechanics and Engineering*, 413:116126, August 2023. ISSN 0045-7825. doi: 10.1016/j.cma.2023.116126. URL http://doi.org/10.1016/j.cma.2023.116126.

Sifan Wang, Yujun Teng, and Paris Perdikaris. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM J. Sci. Comput.*, 43(5):A3055–A3081, 2021. doi: 10.1137/20M1318043. URL https://doi.org/10.1137/20M1318043.

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, January 2022. ISSN 0021-9991. doi: 10.1016/j.jcp.2021.110768. URL http://doi.org/10.1016/j.jcp.2021.110768.

Sifan Wang, Zehao Dou, Siming Shan, Tong-Rui Liu, and Lu Lu. Fundiff: diffusion models over function spaces for physics-informed generative modeling. *Nature Communications*, 17(1), April 2026. ISSN 2041-1723. doi: 10.1038/s41467-026-72292-0. URL http://doi.org/10.1038/s41467-026-72292-0.

Tsun-Hsuan Johnson Wang, Juntian Zheng, Pingchuan Ma, Yilun Du, Byungchul Kim, Andrew Spielberg, Joshua B. Tenenbaum, Chuang Gan, and Daniela Rus. Diffusebot: Breeding soft robots with physics-augmented generative diffusion models. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/8b1008098947ad59144c18a78337f937-Abstract-Conference.html.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for pdes on general geometries. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53681–53705. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/wu24r.html.

Zixue Xiang, Wei Peng, Xu Liu, and Wen Yao. Self-adaptive loss balanced physics-informed neural networks. *Neurocomputing*, 496:11–34, 2022. doi: 10.1016/J.NEUCOM.2022.05.015. URL https://doi.org/10.1016/j.neucom.2022.05.015.

Aobo Yang, Zhen Wei, Rhea P. Liem, and Pascal Fua. Physics-guided generative design with gradient-based source-space optimization on flow matching. *Structural and Multidisciplinary Optimization*, 69(5), May 2026. ISSN 1615-1488. doi: 10.1007/s00158-026-04318-6. URL http://doi.org/10.1007/s00158-026-04318-6.

Jiachen Yao, Chang Su, Zhongkai Hao, Songming Liu, Hang Su, and Jun Zhu. Multiadam: Parameter-wise scale-invariant optimizer for multiscale training of physics-informed neural networks. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 39702–39721. PMLR, 2023. URL https://proceedings.mlr.press/v202/yao23c.html.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/3fe78a8acf5fda99de95303940a2420c-Abstract.html.

Mingze Yuan, Pengfei Jin, Na Li, and Quanzheng Li. PIRF: physics-informed reward fine-tuning for diffusion models. *CoRR*, abs/2509.20570, 2025. doi: 10.48550/ARXIV.2509.20570. URL https://doi.org/10.48550/arXiv.2509.20570.

Yi Zhang and Difan Zou. Physics-informed distillation of diffusion models for pde-constrained generation. *CoRR*, abs/2505.22391, 2025. doi: 10.48550/ARXIV.2505.22391. URL https://doi.org/10.48550/arXiv.2505.22391.

A Additional Theory and Proofs

This appendix develops the geometric results used in Section 3. We follow the same order as the main text: we first formalize the coordinate dependence of parameter-gradient conflict, then characterize realizable objective gradients in function space, prove the intrinsic Pareto-field result, and finally separate the ideal first-order theory from the finite-step numerical realization. Only the notation needed for each argument is introduced below, so that each subsection can be read independently, with cross-references to the main text given where each result is used.

A.1 Geometric Setup and Notation

Fix a minibatch B and an observation map

$$\mathcal{O}_B : \Theta \to \mathcal{H}_B, \qquad y = \mathcal{O}_B(\theta), \qquad J = D\mathcal{O}_B(\theta), \tag{9}$$

where $\Theta \subseteq \mathbb{R}^{d_\theta}$ is parameter space and $\mathcal{H}_B$ is the finite-dimensional observation space used by the two objectives. The observation space is equipped with the inner product

$$\langle a, b\rangle_M = a^\top M b, \qquad \|a\|_M = \sqrt{a^\top M a}, \qquad M \succ 0. \tag{10}$$

The realizable first-order output motions form

$$T_\theta := \mathrm{range}(J). \tag{11}$$

For objective $i \in \{D, P\}$, write $\mathcal{L}_i(\theta) = \widetilde{\mathcal{L}}_i(y)$ and let $h_i \in \mathcal{H}_B$ denote its ambient output-space gradient,

$$d\widetilde{\mathcal{L}}_i(y)[\delta y] = \langle h_i, \delta y\rangle_M. \tag{12}$$

The corresponding parameter gradient is $r_i = J^\top M h_i$. We use g_i for the restriction of this differential to T_θ, $\hat{g}_i = g_i/\|g_i\|_M$ for the normalized realizable gradient, and $d^\star$ for the intrinsic Pareto field. When the parameter index is clear, we write T in place of T_θ, and likewise suppress the minibatch subscript of $\mathcal{O}_B$ and $\mathcal{H}_B$ when no confusion arises, which is the case in every argument that follows.

A.2 Coordinate Dependence of Parameter-Space Conflict

The first result shows that the Euclidean angle between parameter gradients is not an intrinsic property of the represented model, but a property of the chosen parameter chart.

Proposition 1 (Coordinate dependence of parameter-gradient angles)**.** *Let $r_1, r_2 \in \mathbb{R}^{d_\theta}$ be nonzero and linearly independent at θ_0. For any $\bar{\gamma} \in (-1, 1)$, there exists a locally invertible parameter chart $\theta = \psi(\phi)$ for the same local model function such that*

$$\cos\big(Q^\top r_1, Q^\top r_2\big) = \bar{\gamma}, \qquad Q = D\psi(\phi_0). \tag{13}$$

Proof of Proposition 1. Let $r_1, r_2 \in \mathbb{R}^{d_\theta}$ be nonzero and linearly independent. Extend them to a basis $\{r_1, \dots, r_{d_\theta}\}$ of $\mathbb{R}^{d_\theta}$. Let e_j denote the jth standard basis vector. For any target cosine $\bar{\gamma} \in (-1, 1)$, define a linear map L on this basis by

$$Lr_1 = e_1, \qquad Lr_2 = \bar{\gamma} e_1 + \sqrt{1 - \bar{\gamma}^2}\, e_2, \qquad Lr_j = e_j \quad (j \geq 3). \tag{14}$$

The image vectors are linearly independent, so L is invertible. Set $Q^\top = L$ and choose the local affine chart

$$\psi(\phi) = \theta_0 + Q(\phi - \phi_0), \tag{15}$$

which is locally invertible and represents the same model through a change of coordinates. By the chain rule,

$$\nabla_\phi \mathcal{L}_i = Q^\top \nabla_\theta \mathcal{L}_i = L r_i. \tag{16}$$

The two transformed gradients have unit norm and inner product $\bar{\gamma}$, hence

$$\cos\big(Q^\top r_1, Q^\top r_2\big) = \bar{\gamma}. \tag{17}$$

This proves Proposition 1. □

Degenerate Cases. If $r_2 = \alpha r_1$ for a nonzero scalar α, every invertible coordinate change preserves collinearity and the cosine remains $\operatorname{sign}(\alpha)$. If either gradient is zero, the cosine is undefined. The arbitrary-angle result therefore concerns the generic linearly independent case.

Interpretation. The proposition changes only the parameter chart, not the represented function. Consequently, a parameter-gradient cosine can vary even though the local input–output map is unchanged. This motivates measuring objective interaction through output motions that are realizable by the model rather than through Euclidean parameter coordinates, which is the construction developed in the next subsection, where the projection and the intrinsic direction are defined.

A.3 Realizable Gradients as Metric Projections

The realizable gradient is the Riesz representative of the objective differential after restricting it to T. For any realizable variation $v = J\zeta \in T$,

$$\begin{aligned} d\widetilde{\mathcal{L}}_i(y)[v] &= d\widetilde{\mathcal{L}}_i(y)[J\zeta] && (18)\\ &= \langle h_i, J\zeta \rangle_M && (19)\\ &= \zeta^\top J^\top M h_i && (20)\\ &= \zeta^\top r_i. && (21) \end{aligned}$$

We seek $g_i \in T$ such that

$$\langle g_i, v \rangle_M = d\widetilde{\mathcal{L}}_i(y)[v] \qquad \text{for all } v \in T. \tag{22}$$

Writing $g_i = J\xi_i$ and substituting $v = J\zeta$ gives

$$\zeta^\top J^\top M J \xi_i = \zeta^\top r_i \qquad \text{for every } \zeta, \tag{23}$$

which is equivalent to

$$J^\top M J\, \xi_i = r_i. \tag{24}$$

The same equation is obtained from the least-squares problem

$$\xi_i \in \arg\min_{\xi} \frac{1}{2} \|J\xi - h_i\|_M^2. \tag{25}$$

Hence $g_i = J\xi_i$ is precisely the M-orthogonal projection of h_i onto the realizable tangent:

$$g_i = P_T^M h_i, \qquad P_T^M = J\left(J^\top M J\right)^\dagger J^\top M. \tag{26}$$

If J is rank deficient, the parameter lift ξ_i need not be unique. Any two solutions differ by an element of $\ker(J)$, so their realized output vector $J\xi_i$ is identical. The function-space gradient g_i is therefore uniquely defined even when its parameter-space lift is not, so the intrinsic quantities of the next subsection are well posed regardless of the rank of J, independently of which parameter lift realizes them; only the lift, not the realized direction, depends on this choice.

This projection also clarifies the relation between the usual parameter gradient and the intrinsic geometry. The vector r_i is not discarded; it appears as the right-hand side of Eq. 24. What changes is the geometric object used to compare objectives: IGFM compares the realizable representatives g_i in T, rather than the coordinate-dependent vectors r_i in parameter space.

A.4 Invariance and Common-Descent Properties

This subsection gives the formal version of the properties summarized in Section 3.

Proposition 2 (Intrinsic Pareto field)**.** *Fix the observation map $\mathcal{O}_B$ and metric M. For nonzero realizable gradients, the field $d^\star$ in Eq. 28 below is invariant to locally invertible changes of parameter coordinates, independent positive rescaling of each objective, and permutation of the objectives. If $c^\star \neq 0$, then every objective decreases to first order along $d^\star$ and satisfies*

$$-\frac{d\widetilde{\mathcal{L}}_i(y)[d^\star]}{\|g_i\|_M} \geq \|c^\star\|_M^2 > 0. \tag{27}$$

For two objectives, both normalized margins equal $\mu = (1+\gamma_T)/2$.

Recall the normalized realizable gradients $\hat{g}_i = g_i/\|g_i\|_M$ and the minimum-norm construction

$$c^\star := \arg\min_{c\in\mathrm{conv}\{\hat{g}_i\}} \frac{1}{2}\|c\|_M^2, \qquad d^\star := -c^\star. \tag{28}$$

Equivalently, $d^\star$ is the unique solution of

$$d^\star = \arg\min_{v\in T}\left[\frac{1}{2}\|v\|_M^2 + \max_i \langle \hat{g}_i, v\rangle_M\right]. \tag{29}$$

We prove the properties in Proposition 2 in turn.

Chart Invariance. Let $\theta = \psi(\phi)$ be a locally invertible reparameterization with Jacobian $Q = D\psi(\phi)$. The observation Jacobian in the new chart is

$$J_\phi = D(\mathcal{O}_B \circ \psi)(\phi) = J_\theta Q. \tag{30}$$

Since Q is invertible,

$$\mathrm{range}(J_\phi) = \mathrm{range}(J_\theta) = T. \tag{31}$$

At corresponding parameter points, the observation y, the objective differentials, and the metric M are unchanged. Therefore the tangent T, the projector P_T^M, the realizable gradients g_i, their normalized directions $\hat{g}_i$, and the intrinsic cosine γ_T are all unchanged. The optimization problem in Eq. 28 is consequently identical in both charts, so $d^\star$ is parameter-chart invariant. Only the parameter-space lift used to realize $d^\star$ changes, and this lift ambiguity is precisely the rank-deficiency freedom noted in Section A.3, where any two lifts differ by an element of $\ker(J)$ and therefore produce the same realized output vector, so the chart-invariance argument below never depends on the particular parameter lift chosen, only on the direction realized in the tangent T at that state.

Positive Objective Rescaling and Permutation. Replacing objective i by $\omega_i\mathcal{L}_i$ with $\omega_i > 0$ multiplies both h_i and g_i by ω_i but leaves

$$\hat{g}_i = \frac{g_i}{\|g_i\|_M} \tag{32}$$

unchanged, since numerator and denominator scale identically. Therefore the convex hull in Eq. 28 and its minimum-norm point are unchanged. Permuting the objectives also leaves the same convex hull. Thus $d^\star$ is invariant to independent positive rescaling and permutation of the objectives, since both leave the normalized representatives unchanged, which completes the invariance claims of Proposition 2, leaving the scale-free common-descent property to be established next.

Existence, Uniqueness, and Common Descent. Let

$$\mathcal{C} := \mathrm{conv}\{\hat{g}_i\}. \tag{33}$$

Because $\mathcal{C}$ is compact and convex, the strictly convex function $\frac{1}{2}\|c\|_M^2$ has a unique minimizer $c^\star$ on $\mathcal{C}$. The projection optimality condition gives

$$\langle c^\star, \bar{c} - c^\star\rangle_M \geq 0 \qquad \text{for every } \bar{c}\in\mathcal{C}. \tag{34}$$

Setting $\bar{c} = \hat{g}_i$ yields

$$\langle \hat{g}_i, c^\star\rangle_M \geq \|c^\star\|_M^2 =: \mu. \tag{35}$$

Since $d^\star = -c^\star$ and $g_i = \|g_i\|_M \hat{g}_i$,

$$-\frac{d\widetilde{\mathcal{L}}_i(y)[d^\star]}{\|g_i\|_M} = -\frac{\langle g_i, d^\star\rangle_M}{\|g_i\|_M} \tag{36}$$

$$= \langle \hat{g}_i, c^\star\rangle_M \tag{37}$$

$$\geq \mu. \tag{38}$$

Therefore every objective decreases to first order whenever $c^\star \neq 0$. This proves the normalized common-descent statement in Proposition 2, with the two-objective case made explicit next.

Two-Objective Closed Form. For two objectives, write

$$c(\alpha) = \alpha\hat{g}_D + (1-\alpha)\hat{g}_P, \qquad \alpha \in [0,1]. \tag{39}$$

Using $\|\hat{g}_D\|_M = \|\hat{g}_P\|_M = 1$ and $\gamma_T = \langle \hat{g}_D, \hat{g}_P\rangle_M$,

$$\|c(\alpha)\|_M^2 = \alpha^2 + (1-\alpha)^2 + 2\alpha(1-\alpha)\gamma_T \tag{40}$$

$$= 1 - 2(1-\gamma_T)\alpha(1-\alpha). \tag{41}$$

For $\gamma_T < 1$, this is minimized at $\alpha^\star = 1/2$, so

$$c^\star = \frac{\hat{g}_D + \hat{g}_P}{2}, \qquad d^\star = -\frac{\hat{g}_D + \hat{g}_P}{2}, \qquad \mu = \frac{1+\gamma_T}{2}. \tag{42}$$

If $\gamma_T = 1$, the normalized gradients coincide, every convex combination is the same vector, and the closed form above applies unchanged with $\alpha^\star = 1/2$ and margin $\mu = 1 = (1+\gamma_T)/2$.

Relation to Two-Objective ConFIG. Let $a = U(r_D)$ and $b = U(r_P)$ with $U(z) = z/\|z\|_2$. For two objectives, ConFIG (Liu et al., 2025) reduces to the normalized bisector

$$q_{\mathrm{CFG}} = U(a+b) = U(U(r_D) + U(r_P))\,. \tag{43}$$

Thus ConFIG and IGFM use the same two-objective combination principle after normalization. Their distinction is geometric: ConFIG forms the bisector from parameter gradients, whereas IGFM forms it from realizable function-space gradients, so the contribution lies in the space in which objective conflict is defined; the combination is the classical minimum-norm one.

A.5 Approximate Solves and Finite-Step Consistency

The preceding results concern the ideal field $d^\star$. The implementation in Section 3 uses damped, truncated tangent solves and a finite parameter update. We therefore separate a local approximation statement from the behavior of an accumulated training trajectory over many optimization steps.

Local Robustness to an Approximate Tangent Solve. Let

$$\widetilde{v} = d^\star + e \in T \tag{44}$$

be an approximate realized direction. Combining Eq. 38 with Cauchy–Schwarz gives

$$-\frac{d\widetilde{\mathcal{L}}_i(y)[\widetilde{v}]}{\|g_i\|_M} = -\frac{\langle g_i, d^\star\rangle_M}{\|g_i\|_M} - \frac{\langle g_i, e\rangle_M}{\|g_i\|_M} \tag{45}$$

$$\geq \mu - \|e\|_M. \tag{46}$$

Hence

$$\|e\|_M < \mu \tag{47}$$

is sufficient to preserve strict common descent at that step. The statement is intentionally local: it does not imply that repeated approximate steps follow the same trajectory as the ideal field.

Accumulated Direction Error. To track repeated approximation error, let $\widetilde{v}_j$ denote the realized direction at optimization step j, let $v_j^\star$ denote the corresponding reference intrinsic direction, and let $\Delta s_j \geq 0$ be the functional travel of that step. We use the travel-weighted diagnostic

$$\mathcal{E}_{N_{\mathrm{step}}} := \sum_{j=0}^{N_{\mathrm{step}}-1} \Delta s_j \, \frac{\|\widetilde{v}_j - v_j^\star\|_M}{\|v_j^\star\|_M + \varepsilon}, \qquad \varepsilon > 0. \tag{48}$$

This quantity is not a convergence bound. It is used only to distinguish small stepwise error from error repeatedly accumulated over substantial function-space travel; the corresponding measurements are reported in the solver-dynamics and long-horizon analyses of Appendix C.

Finite-Step Chart Consistency. Let $\theta = \psi(\phi)$ be two equivalent parameter charts. Suppose parameter lifts ξ_θ and ξ_ϕ realize the same ideal function-space direction,

$$J_\theta \xi_\theta = J_\phi \xi_\phi = d^\star. \tag{49}$$

For a scalar step size $\eta \geq 0$, define

$$y_\theta(\eta) := \mathcal{O}_B(\theta + \eta\xi_\theta), \qquad y_\phi(\eta) := \mathcal{O}_B(\psi(\phi + \eta\xi_\phi)). \tag{50}$$

The two curves agree at $\eta = 0$ and have the same first derivative. Their second derivatives are

$$y''_\theta(0) = D^2\mathcal{O}_B[\xi_\theta, \xi_\theta], \tag{51}$$

$$y''_\phi(0) = D^2\mathcal{O}_B[Q\xi_\phi, Q\xi_\phi] + J_\theta D^2 g[\xi_\phi, \xi_\phi]. \tag{52}$$

The one-step difference is therefore $O(\eta^2)$. Under bounded second derivatives and a Lipschitz realized update field, these local discrepancies accumulate to $O(\eta)$ over a fixed horizon containing $O(1/\eta)$ explicit steps. Thus local chart invariance of the ideal field is compatible with nonzero finite-step chart divergence once damping, truncated solves, optimizer state, and explicit integration are introduced, in line with the twin-run observations of Appendix B, which quantify this accumulated divergence on trained Darcy models, where the twin-run separation is reported after twenty epochs.

A.6 A Minimal Coordinate-Change Example

A two-dimensional example isolates the distinction between parameter-space and function-space conflict. Let

$$\mathcal{O}(\theta) = \theta, \qquad M = I_2, \qquad h_D = e_1, \qquad h_P = e_2. \tag{53}$$

Because the observation map is the identity, $T = \mathbb{R}^2$ and

$$g_D = e_1, \qquad g_P = e_2, \qquad \gamma_T = 0, \qquad d^\star = -\frac{e_1 + e_2}{2}. \tag{54}$$

Now introduce coordinates $\phi \in \mathbb{R}^2$ through the invertible linear map $\theta = Q_\pm \phi$, with

$$Q_\pm^\top = \begin{pmatrix} 1 & \pm 1 \\ 0 & \sigma \end{pmatrix}, \qquad \sigma > 0. \tag{55}$$

The parameter gradients become

$$r_D^{(\phi)} = (1, 0)^\top, \qquad r_P^{(\phi)} = (\pm 1, \sigma)^\top, \tag{56}$$

so their cosine is

$$\gamma_\phi^\pm = \pm \frac{1}{\sqrt{1 + \sigma^2}}. \tag{57}$$

As $\sigma \to 0$, the same pair of orthogonal function-space objectives appears almost perfectly aligned in one parameter chart and almost perfectly opposed in the other. Meanwhile T, g_D, g_P, γ_T, and $d^\star$ remain unchanged. This is the finite-dimensional analogue of the trained-model reparameterization study in Appendix B, where the same parameter-space versus intrinsic contrast appears at trained-network scale, with the parameter-space cosine varying over a wide range and the intrinsic cosine nearly fixed, in line with the static-audit measurements of that appendix on the trained Darcy model.

B Reparameterization and Chart-Consistency Evidence

This appendix complements the geometric analysis in Section 3 with two empirical checks. We first test how conflict measurements behave under function-preserving changes of parameter coordinates while keeping the represented model fixed; function-preserving rescalings provide a familiar example of such parameterization non-uniqueness in neural networks (Neyshabur et al., 2015). We then ask whether the same distinction persists during optimization by training equivalent parameterizations from the same represented function, so that any divergence is attributable to the parameterization rather than to initialization, the training protocol being otherwise identical.

B.1 Synthetic Sanity Check

We begin with a small network in which large families of equivalent parameterizations can be generated directly. A two-layer network with dimensions $5 \to 6 \to 8$ is evaluated under 1000 random function-preserving reparameterizations with condition numbers in $\{1, 3, 10, 30, 100\}$. The parameter-space conflict changes sign in 21.6% of these trials while the represented function is unchanged. The synthetic condition numbers exceed those of the trained systems in the main experiments; the construction serves as a numerical sanity check for the coordinate-dependence result of Appendix A.2, which the next subsection complements on a trained Darcy model.

B.2 Static Audit on a Trained Darcy Model

We next repeat the analysis on the trained Darcy transformer used in the main experiments. We generate 7040 function-preserving reparameterizations with condition numbers up to 3000. Across this sweep, the parameter-space cosine varies from 0.40 to 0.93, while the maximum function-identity error remains below 6.3×10^{-5}. No sign flip occurs in this trained-model sweep, a substantially milder regime than the synthetic construction above, where the sign instability serves as a stress test of the coordinate-dependence mechanism rather than an expectation for trained networks.

The intrinsic statistic is evaluated separately in matched audits at condition numbers 100, 1000, and 3000. Across these matched points, γ_T remains within 0.3093–0.3122, corresponding to less than 1% relative variation. Figure 9 summarizes both measurements. The two statistics behave differently under the same function-preserving coordinate changes: the Euclidean parameter-gradient angle varies substantially, whereas the intrinsic cosine remains nearly unchanged.

Table 4: Static reparameterization audit on the trained Darcy model: the parameter-space range is measured over all 7040 function-preserving reparameterizations, the intrinsic range only at the matched audits (condition numbers 100, 1000, and 3000), within which γ_T varies by less than 1%.

quantity	observed range
number of reparameterizations	7040
condition number $\kappa(Q)$	up to 3000
parameter-space cosine γ_θ	0.40–0.93
intrinsic cosine γ_T	0.3093–0.3122
function-identity error	$\leq 6.3 \times 10^{-5}$

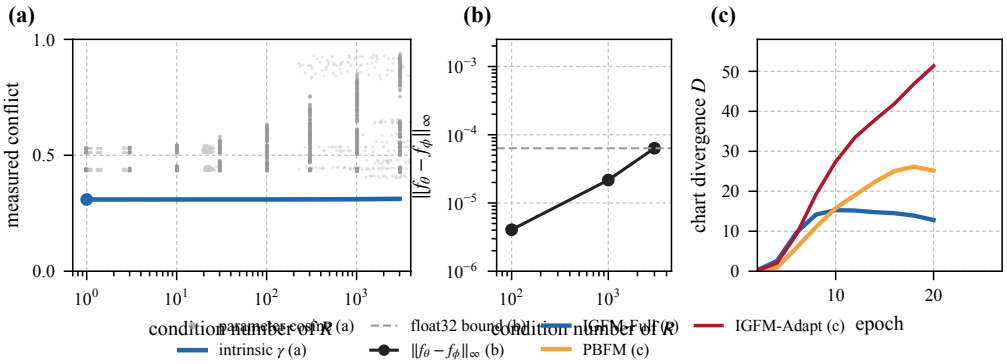


Figure 9: Reparameterization evidence on the trained Darcy model: (a) the parameter-gradient cosine varies with chart conditioning while matched intrinsic audits remain within 0.3093–0.3122, (b) the function-identity error stays below 6.3×10^{-5}, and (c) equivalent-chart training produces different finite-step trajectories for different practical solvers, analyzed below, with the corresponding separations quantified in Table 5 and discussed at the end of this appendix.

Together with the coordinate-dependence result of Appendix A.2, the sweep shows that the parameter-space angle is not an intrinsic property of the represented function: exact function-

preserving transformations of a trained model already induce a nontrivial change in the measured parameter conflict, with the function-identity error of Table 4 below 6.3×10^{-5}.

B.3 Training Under Equivalent Parameter Charts

The static audit concerns a single represented function. We next test whether equivalent parameter charts also lead to different finite-step optimization trajectories. For each twin-run pair, two parameterizations represent the same initial model function and use the same data order, optimizer settings, and sampling stream. We evaluate their function-space separation

$$D_{\mathrm{chart}}(\tau) := \left\| f_{\theta(\tau)}(Z) - f_{\phi(\tau)}(Z) \right\|_2 \tag{58}$$

on a frozen probe set Z of 64 samples, where $f_{\theta(\tau)}$ and $f_{\phi(\tau)}$ denote the model functions represented by the two equivalent parameter charts after τ epochs of training, with the probe set held fixed across all evaluations, so that changes in D_{chart} reflect the trained functions rather than the probe.

At epoch 20, PBFM-1 reaches $D_{\mathrm{chart}} = 25.2$, whereas the default fixed-depth IGFM realization reaches 12.8. Normalizing by accumulated functional travel gives relative divergences of 0.043 and 0.030, respectively. The adaptive tangent solver reaches $D_{\mathrm{chart}} = 51.3$ over the same horizon. These values are averaged over $N = 3$ twin-run pairs, each initialized from the same represented function, with identical data order and sampling stream, so the only difference is the parameter chart.

Table 5: Equivalent-chart training on Darcy after 20 epochs: D_{chart} measures function-space separation between twin runs initialized from the same represented function, and D_{rel} normalizes it by accumulated functional travel; values are averaged over $N = 3$ twin-run pairs.

method	$D_{\mathrm{chart}} \downarrow$	$D_{\mathrm{rel}} \downarrow$
PBFM-1	25.2	0.043
IGFM-k=5	12.8	0.030
IGFM-Adaptive	51.3	—

The twin-run experiment concerns the realized finite-step optimizer rather than the ideal intrinsic field defined in Section 3. The fixed-depth realization is more chart-consistent than PBFM-1 over this horizon, while the adaptive solver is less consistent despite using the same intrinsic geometric target. This distinction motivates the solver-dynamics analysis in Appendix C, where we separate local direction quality from the accumulated effects of damping, truncated solves, step-length choice, and finite-step integration, whose combined effect the twin-run divergence above illustrates.

C Practical Solver Dynamics

This appendix examines the gap between the ideal intrinsic field of Section 3 and its finite-step numerical realization. The analysis is organized around three ingredients that are distinct in theory but coupled in training: the direction realized by the tangent solve, the amount of function-space travel taken along that direction, and the parameter-space mechanism used to realize the step. We first validate the functional-travel diagnostic, then separate step-length and realization effects, and finally study why an adaptive tangent solve that is locally well behaved can diverge from the fixed-depth trajectory over long horizons, a reversal quantified in Section C.4 across all three evaluated seeds, with the crossing epoch varying between them, as Table 7 details for seeds 15, 16, and 17.

C.1 Measuring Functional Travel

Let $s(\tau)$ denote accumulated function-space travel along optimization time τ; this output-motion viewpoint is related to measuring optimization steps through their induced change in function space rather than parameter displacement (Milsom et al., 2025). In practice, we estimate its increments on the same frozen probe set used for the chart-consistency audit. A fixed-probe validation on 64 samples compares the JVP-based estimator against the corresponding explicit probe displacement on four runs; the two estimates induce the same ordering in all four cases, which supports its use throughout the audits below, from the matched-travel comparison to the long-horizon analysis.

The estimator also responds predictably to the step size. With the realized direction and update rule fixed, sweeping the scalar step size η over five orders of magnitude changes the terminal travel from $s_T = 0.0018$ to 18.11 and yields an exactly linear dependence on η, while the training residual remains approximately fixed at $R \approx 4400$. This intervention isolates the step size from the direction itself, since η is the only quantity that varies across the sweep, consistent with the exactly linear response reported above, as expected for a first-order estimate of the realized displacement.

C.2 Separating Direction, Step Length, and Realization

The practical update contains two choices beyond the intrinsic direction: how far to move in function space and how to lift that motion back to parameters. We cross these factors using a fixed scalar step size or the projected step-length rule, together with either direct integration of the realized intrinsic direction or the AdamW parameter realization used in training, giving the four-cell audit of Table 6.

Table 6: Crossed step-length–realization audit on Darcy at epoch 12 (seed 15, frozen evaluation). Each entry reports terminal residual R and accumulated functional travel s_T.

realization	fixed clock (η)	projected rule
direct	$R \approx 26308$ $(s_T \approx 18.1)$	$R = 2131.8$ $(s_T \approx 1.08 \times 10^5)$
AdamW	$R = 5429.5$ $(s_T \approx 5.70 \times 10^4)$	$R = 6617.0$ $(s_T \approx 1.05 \times 10^5)$

The interaction is strongly non-additive. Under direct realization, the projected step-length rule lowers the residual by $12.3\times$, whereas under AdamW it increases the residual by about $1.2\times$. Conversely, AdamW substantially improves the fixed-clock run but is worse than direct realization under the projected step-length rule. The practical trajectory therefore cannot be attributed to a step-length effect or a realization effect in isolation, as the crossed audit in Table 6 above shows.

A matched-travel comparison leads to the same conclusion. At approximately $s = 1.2 \times 10^5$ accumulated travel, fixed-depth IGFM reaches $R = 6617$, whereas PBFM-1 reaches $R = 10066$; PBFM-1 ultimately travels farther, with $s_T = 1.64 \times 10^5$. More accumulated travel is therefore not, by itself, a measure of more effective optimization. Figure 10 visualizes this functional-travel and realization audit, whose panels separate travel, residual, and the crossed interventions.

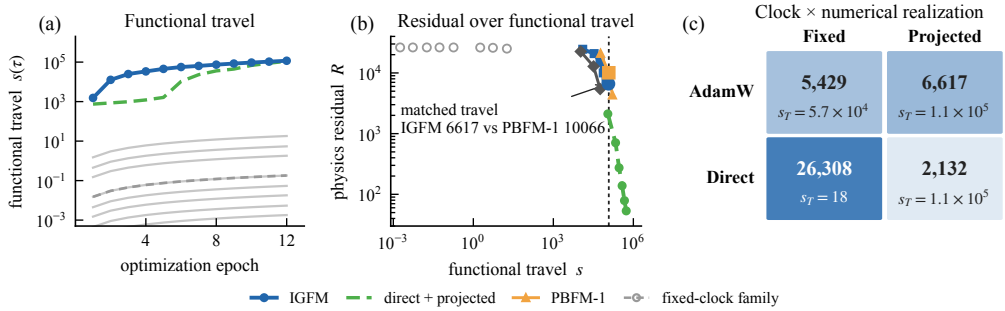


Figure 10: Functional-travel and realization audit. The panels separate accumulated function-space travel from residual reduction and compare the crossed step-length–realization interventions summarized in Table 6, all on the seed-15 Darcy run at epoch 12 under frozen evaluation.

The direct realization with the projected step-length rule also continues to improve beyond the short audit window: evaluations at epochs $12, 18, 24, 30, 36$, and 42 decrease to $R = 53$ without an observed plateau. This behavior motivates treating step length and realization as trajectory-level quantities rather than as one-step properties of the intrinsic field, a distinction developed further in the next subsections, which examine the fixed-depth and adaptive solves and their long-horizon behavior, together with the solver-switch interventions that localize the mechanism.

C.3 Fixed-Depth and Adaptive Tangent Solves

We next compare the default fixed-depth solve, IGFM-k=5, with an adaptive variant whose stopping rule is based on the local tangent-solve certificate. The adaptive rule passes the same-state direction audit and initially uses fewer CG iterations. At epoch 20, it achieves $R = 950 \pm 193$, compared with 2183 ± 274 for the fixed-depth solve, while the local common-descent margins remain favorable.

This short-horizon advantage does not persist. The two methods follow different finite-step trajectories even though both are derived from the same intrinsic geometry. The discrepancy is consistent with the distinction established in Appendix A: the local certificate controls the realized direction at an individual state, but does not imply equality of the accumulated training path.

C.4 Long-Horizon Reversal

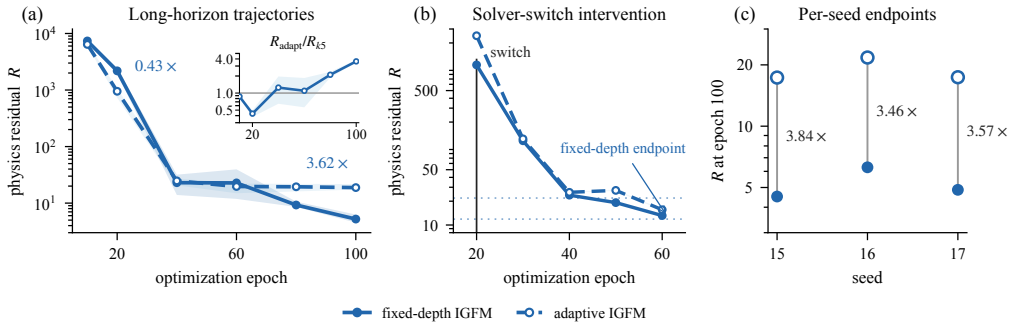


Figure 11: Long-horizon comparison of the fixed-depth and adaptive tangent solves on Darcy. The adaptive solve is ahead early in training, but the ordering reverses under continued optimization.

Figure 11 illustrates the trajectory-level reversal, and Table 7 summarizes it across three seeds. The adaptive solve is better at epochs 10 and 20. The ordering begins to cross between epochs 20 and 40 on seeds 15 and 16 and between epochs 60 and 80 on seed 17. By epoch 100, the fixed-depth solve reaches $R = 5.22 \pm 0.93$, whereas the adaptive solve reaches 18.82 ± 2.53. Thus the crossing epoch varies by seed, but the late-stage reversal occurs in all three runs, which is the regime that the solver-switch interventions below probe further with switches placed both before and after the crossing, separating early-state from late-phase effects, in both switch directions.

Table 7: Long-horizon behavior of the fixed-depth and adaptive tangent solves on Darcy. Trajectory values are mean ± standard deviation over seeds 15/16/17; the qualitative regime column summarizes the ordering at each horizon, with the crossing region between epochs 20 and 40.

horizon	IGFM-k=5	IGFM-Adaptive	qualitative regime
epoch 10	7376 ± 530	6407 ± 300	adaptive ahead
epoch 20	2183 ± 274	950 ± 193	adaptive ahead
epoch 40	23.0 ± 9.0	24.8 ± 3.6	crossing region
epoch 80	9.26 ± 1.19	19.45 ± 3.01	fixed-depth ahead
epoch 100	5.22 ± 0.93	18.82 ± 2.53	fixed-depth ahead

A same-state audit helps localize the mechanism. The adaptive solve maintains favorable local margins, but its travel-weighted direction discrepancy accumulates over training. Using the diagnostic of Appendix A,

$$\mathcal{E}_{N_{\text{step}}} = \sum_{j=0}^{N_{\text{step}}-1} \Delta s_j \frac{\|\widetilde{v}_j - v_j^\star\|_M}{\|v_j^\star\|_M + \varepsilon}, \tag{59}$$

the cumulative discrepancy grows together with the late-stage divergence. This observation is descriptive rather than a convergence claim: it shows that repeated small realization errors can accumulate over substantial functional travel even when individual steps satisfy a local certificate.

C.5 Solver-Switch Interventions

We perform two interventions to distinguish early-state effects from late-phase trajectory effects. First, on seed 15, we switch the solver at epoch 20 and continue training to epoch 60. The adaptive-to-fixed run ends at $R = 13.2$, close to the fixed-depth reference endpoint 11.9, whereas the fixed-to-adaptive run ends at $R = 15.6$. This asymmetry indicates that the late-phase update rule, rather than the early advantage alone, determines the final trajectory, consistent with the accumulated-discrepancy analysis below, in which the travel-weighted error grows together with the divergence.

Second, we switch a converged fixed-depth checkpoint to the adaptive rule at epoch 100. The residual rises immediately from 4.515 to 20.26 and recovers only to 8.18 by epoch 120. Near this regime, the adaptive stopping criterion activates less often: the mean CG depth is $\bar{k} = 4.12$, and 56% of steps reach $k_{\max}$. The intervention again points to accumulated finite-step realization as the source of the trajectory difference, in line with the reversal analysis above.

These experiments complement the local theory rather than modify it. The intrinsic field is defined by the realizable function-space geometry at a state; damping, truncated solves, the step-length rule, the parameter lift, and optimizer state determine how that local field is integrated over training. Computational cost and implementation-level efficiency are reported separately with the reproducibility details in Appendix G, which also records the instrumentation policy for the audits above.

D Additional Ablations

This appendix complements the representative ablations in Section 4.3 with the complete numerical sweeps used to characterize the practical IGFM realization, separating three questions: how accurately the tangent projection must be solved, how damping changes the realized correction, and whether the scale-invariance of the ideal field survives finite-step training. Two further audits measure how closely the practical direction tracks a high-accuracy reference solve, and how the operating point responds to the choice of observation metric. We close with a small learning-rate sensitivity check. Unless stated otherwise, all results use Darcy Flow, the default projected step-length rule, and the same frozen evaluation protocol as the main experiments.

D.1 Tangent-Solve Depth

The fixed-depth realization approximates each tangent projection with a prescribed number of conjugate-gradient iterations. Table 8 reports the full depth sweep at damping $\lambda = 10^{-3}$. Increasing the depth from $k = 2$ to $k = 5$ steadily reduces both the physics residual and the distribution distance. The improvement saturates around the default $k = 5$: increasing the solve to $k = 8$ does not improve either reported metric. Wall-clock cost grows monotonically with solve depth, from $1.06\times$ PBFM-1 at $k = 1$ to $1.27\times$ at $k = 8$, consistent with the timing audit of Appendix G.2.

Table 8: CG-depth sweep on Darcy with damping $\lambda = 10^{-3}$. Values are mean $\pm$ standard deviation across training seeds, under the frozen evaluation protocol of Appendix E.5.

CG depth	$R\downarrow$	$W_2\downarrow$
$k = 2$	19.03 ± 1.82	0.019 ± 0.002
$k = 3$	11.24 ± 1.10	0.016 ± 0.002
$k = 4$	8.15 ± 0.92	0.015 ± 0.001
$k = 5$ (default)	6.93 ± 1.24	0.014 ± 0.002
$k = 8$	7.28 ± 0.85	0.014 ± 0.002

The sweep supports a practical sufficiency interpretation rather than a monotonic "more CG is better" rule. Shallow solves under-resolve the tangent correction, while deeper solves beyond $k = 5$ add runtime cost with little change in the final operating point. This is why the main experiments use a fixed depth of $k = 5$, with the corresponding wall-clock costs reported in Appendix G.

D.2 Direction Fidelity of the Practical Realization

The depth and damping sweeps characterize the practical realization through its final operating point; a complementary audit measures how closely the realized direction tracks the intrinsic direction at the same model state. At three training stages (epochs 10, 50, and 100) of the Darcy Flow and Kolmogorov Flow training runs, we compute the practical direction $\tilde{d}^\star$, the fixed-depth realization of $d^\star$ (k=5, λ=10^{-3}), and a high-accuracy reference solve (k=100, λ=10^{-6}) of the same construction from the same objective tangents, and compare the two directions in the observation space, where the default $M = I$ coincides with the Euclidean metric; Table 9 reports the cosine agreement and the relative error between the two directions at every audited training stage of both tasks.

Table 9: Direction-fidelity audit on Darcy Flow and Kolmogorov Flow. At each training stage, the practical direction $\tilde{d}^\star$ (k=5, λ=10^{-3}) is compared with a high-accuracy reference $d^\star$ (k=100, λ=10^{-6}) at the same model state; both quantities are computed in the observation space.

Task	Checkpoint	$\cos(\tilde{d}^\star, d^\star)\uparrow$	$\|\tilde{d}^\star - d^\star\|_2/\|d^\star\|_2\downarrow$
Darcy Flow	epoch 10	0.9995	3.8%
	epoch 50	0.9991	5.2%
	epoch 100	0.9988	5.8%
Kolmogorov Flow	epoch 10	0.9993	4.5%
	epoch 50	0.9986	6.3%
	epoch 100	0.9982	7.1%

Across all six audit points, the practical and reference directions agree to cosine 0.9982–0.9995, with relative error 3.8–7.1%. The deviation grows mildly over training while remaining small, consistent with the saturated-but-not-exact regime of the depth sweep (Table 8): the $k = 5$ solve already sits on the flat part of the depth–residual curve, and the residual offset to the near-undamped reference drifts slowly rather than accumulating into a distinct direction. The fixed-depth realization therefore tracks the reference direction to within a few percent throughout training, with deviations arising from both finite damping and the truncated solve; the trajectory-level counterpart is given in Appendix C.

D.3 Damping

Damping regularizes the tangent system

$$\left(J^\top M J + \lambda I\right)\xi_i = r_i$$

and therefore controls how closely the numerical realization follows the undamped tangent projection. The damping is applied at λ times the curvature scale of the data gradient (Appendix G), so the swept values below are proportionality coefficients rather than absolute regularization strengths. Table 10 reports the complete sweep at the default depth $k = 5$, with all other settings held at the reference configuration of Table 14, so the sweep isolates the damping coefficient.

Table 10: Damping sweep on Darcy with CG depth $k = 5$. Values are mean $\pm$ standard deviation across training seeds, under the frozen evaluation protocol of Appendix E.5.

damping	$R\downarrow$	$W_2\downarrow$
$\lambda = 10^{-5}$	5.72 ± 1.07	0.016 ± 0.003
$\lambda = 10^{-4}$	8.47 ± 3.57	0.015 ± 0.002
$\lambda = 10^{-3}$ (default)	6.93 ± 1.24	0.014 ± 0.002
$\lambda = 10^{-2}$	10.67 ± 3.44	0.015 ± 0.002
$\lambda = 10^{-1}$	20.90 ± 2.96	0.018 ± 0.003

The dependence is not monotone. Low-to-moderate damping gives the strongest operating points, while $\lambda = 10^{-1}$ substantially weakens both metrics. The smallest damping gives the lowest residual in this sweep, whereas the default $\lambda = 10^{-3}$ gives the lowest reported W_2. We therefore use $\lambda = 10^{-3}$ as a stable default rather than selecting damping solely by the physical residual.

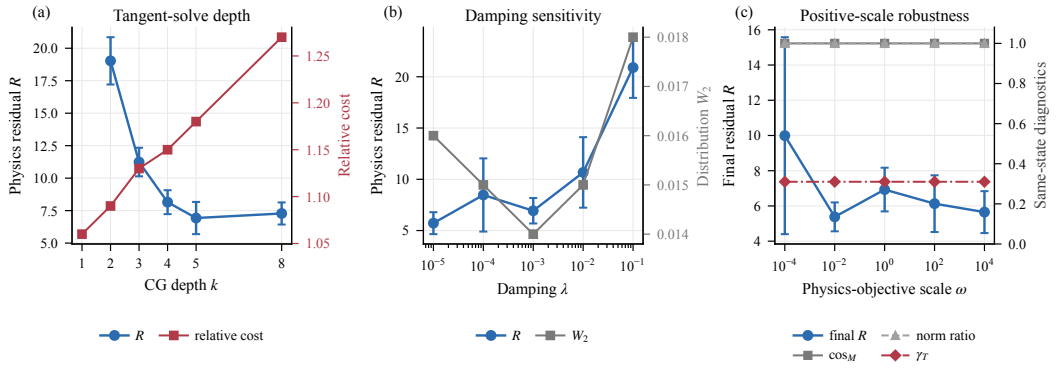


Figure 12: Numerical robustness of the practical IGFM realization on Darcy: (a) the residual improves with CG depth up to the default $k = 5$ and saturates while runtime keeps growing; (b) low-to-moderate damping gives similar operating points while strong damping degrades the physical objective; (c) the same-state intrinsic field is invariant to positive objective rescaling, with finite-step endpoints in the same broad regime over the full rescaling range of $\omega \in [10^{-4}, 10^{4}]$.

D.4 Objective-Scale Invariance

Sensitivity to heterogeneous loss scales is a well-known issue in physics-informed learning, motivating adaptive weighting and scale-aware optimization methods (van der Meer et al., 2022; Xiang et al., 2022; Yao et al., 2023). The ideal field is invariant to independent positive rescaling of either objective because IGFM constructs the Pareto direction from the normalized realizable gradients. We test this prediction by replacing $\mathcal{L}_P$ with $\omega\mathcal{L}_P$ for

$$\omega \in \{10^{-4}, 10^{-2}, 1, 10^{2}, 10^{4}\}.$$

The audit is performed in two stages. First, at a fixed model state we recompute the intrinsic direction without changing the parameters. Second, we train the full fixed-depth realization under the same scale factors, so that the two stages separate same-state invariance from finite-step endpoint behavior.

Table 11: Objective-scale audit on Darcy. The fixed-state columns compare the ideal intrinsic field after replacing $\mathcal{L}_P$ by $\omega\mathcal{L}_P$; the final column reports the endpoint residual of the corresponding end-to-end training run, all under the frozen evaluation protocol of Appendix E.5.

ω	$\cos_M(d^\star(\omega), d^\star(1))$	γ_T	$\|d^\star(\omega)\|_M / \|d^\star(1)\|_M$	final $R\downarrow$
10^{-4}	1.0000	0.3102	1.0000	9.99 ± 5.59
10^{-2}	1.0000	0.3102	1.0000	5.38 ± 0.82
1	1.0000	0.3102	1.0000	6.93 ± 1.24
10^{2}	1.0000	0.3102	1.0000	6.13 ± 1.61
10^{4}	1.0000	0.3102	1.0000	5.65 ± 1.19

At the fixed state, all three geometric quantities are unchanged across the full eight-order-of-magnitude rescaling range, directly matching the scale-invariance property of the intrinsic field (Appendix A.4). The end-to-end endpoints are not identical, as expected for a damped, finite-step realization with optimizer state and truncated tangent solves, but all five runs remain in the same broad performance regime. The distinction is important: exact scale invariance is a property of the ideal same-state field, whereas endpoint robustness additionally depends on numerical realization and training dynamics. Figure 12 summarizes the three numerical-robustness sweeps of this appendix, on tangent-solve depth, damping, and objective rescaling, respectively.

D.5 Observation-Metric Sensitivity

The invariance guarantee of Section 3 holds for a fixed observation metric M; this audit probes how the operating point responds to the choice of M itself, complementing the objective-scale study above, which varies the objectives rather than the metric. We retrain the complete Darcy pipeline under two physically motivated alternatives to the default $M = I$: a variance-weighted metric

$\mathrm{diag}(\sigma_{\mathrm{grid}}^{-2})$, where σ_{grid}^2 is the across-sample variance of each spatial grid point on the training set, and a Sobolev-type smoothed metric $M = I - 10^{-3}\nabla^2$, which penalizes rough observation directions. For each setting, Table 12 reports the endpoint residual and distribution distance over five training seeds, together with the cosine between the intrinsic direction computed under that metric and under the default, evaluated at the frozen state of the default run so that the comparison isolates the effect of the metric from training-path differences between the retrained models.

Table 12: Observation-metric sensitivity on Darcy Flow. Endpoint metrics are mean $\pm$ standard deviation over five training seeds; the direction cosine against the default ($M = I$) direction is evaluated at the frozen state of the default run, so that the cosine reflects the metric change alone.

Observation metric	$R \downarrow$	$W_2 \downarrow$	$\cos(d_M^\star, d_I^\star) \uparrow$
Default grid ($M = I$)	6.93 ± 1.24	0.014 ± 0.002	1.0000
Variance-weighted ($\mathrm{diag}(\sigma_{\mathrm{grid}}^{-2})$)	7.02 ± 1.28	0.015 ± 0.002	0.978 ± 0.005
Sobolev H^1 ($M = I - 10^{-3}\nabla^2$)	6.88 ± 1.21	0.015 ± 0.002	0.965 ± 0.008

Both alternatives leave the operating point statistically unchanged: R stays within 6.88–7.02 and W_2 within 0.014–0.015, well inside the seed-level dispersion of the default configuration, while the same-state directions agree with the default to cosine 0.965–0.978. Within the class of physically reasonable metrics tested, IGFM is therefore robust to the choice of M; the invariance statement itself remains conditional on the fixed metric M, as discussed in the Limitations of Section 5.

D.6 Learning-Rate Sensitivity

Finally, we vary the AdamW learning rate around the default value while keeping the remaining IGFM configuration fixed. The available sweep records the final Darcy residual.

Table 13: Learning-rate sensitivity of the default IGFM realization on Darcy. Values are mean $\pm$ standard deviation across training seeds under the protocol of the main experiments.

learning rate	$R \downarrow$
1.5×10^{-6}	10.33 ± 1.35
3×10^{-5} (default)	6.93 ± 1.24
6×10^{-5}	6.90 ± 0.67

The two larger learning rates give nearly identical residuals, whereas the substantially smaller step size converges to a weaker endpoint under the same training budget. This check is included as an optimization sensitivity analysis; the geometry-specific conclusions are supported by the tangent-solve, damping, and objective-scale experiments above and summarized jointly in Figure 12.

E Task Definitions and Evaluation Protocols

This appendix specifies the three tasks used in the main benchmark and the evaluation protocol shared across methods; the visual overview of the three reference fields is given in Figure 3 of the main text. Here we give the task definitions and frozen evaluation rules needed to interpret the reported metrics; implementation details are collected in Appendix G, and the long-horizon extension protocol appears at the end of this appendix, following the same frozen rules.

E.1 Observation Maps

In all three tasks, the observation $\mathcal{O}_B$ of Section 3 collects the network quantities on which the two objectives depend: the flow-model velocity output on the current flow-matching minibatch, and the endpoint field reached from it by the unrolled integration. The data objective is the flow-matching regression loss on the velocity output, and the physics objective is a task functional evaluated on the unrolled endpoint. The task-specific components are: for Darcy Flow, per-channel de-normalization with the fixed training-set pressure and permeability statistics followed by the discretized Darcy residual; for Kolmogorov Flow, the squared divergence residual of the unrolled endpoint field,

computed by spectral differentiation; and for Topology Optimization, the displacement-to-density mapping of the endpoint field followed by the self-adjoint equilibrium and volume residual of Appendix E.4. In the notation of Eq. equation 3, all tasks use the linear time weight $w(t) = t$ ($\beta = 1$), applied identically to every physics-constrained method compared. This is the weighting used by the training code of the PBFM release (Baldan et al., 2026) on which the shared stack is built; its written Algorithm 1 instead states the residual term as $\|tR\|^2$ (an effective t^2 weight), while the released code, like all methods compared here, applies the linear t to the reduced functional Φ_P. Φ_P is the grid-mean absolute Darcy residual, the grid-mean squared divergence residual of the Kolmogorov field, and the log-compliance with volume term of Appendix E.4, respectively. All normalizations use fixed dataset-level constants computed once on the training set, so the observation map and the identity output metric are fixed throughout training, as the invariance analysis of Section 3 assumes.

E.2 Darcy Flow

Darcy Flow is the primary PDE benchmark and the source of the trained model used in the reparameterization study. The task generates pressure fields governed by an elliptic Darcy system (Evans, 2022). Following the PBFM evaluation setup, physical consistency is measured by the discretized governing-equation residual R, while distribution fidelity is measured by the 2-Wasserstein distance W_2 (Villani et al., 2009) between generated and reference samples, both computed by the frozen evaluator of Appendix E.5 rather than by training-time instrumentation, keeping evaluation decoupled from training diagnostics, as the protocol of Section E.5 requires for every reported number.

All compared methods use the same held-out physical conditions and the same frozen evaluator. The comparison therefore changes the training geometry while holding the task definition, residual computation, sampling protocol, and distribution metric fixed, so that differences are attributable to training rather than evaluation, as required for the controlled comparison of Table 3.

E.3 Kolmogorov Flow

Kolmogorov Flow extends the evaluation from an elliptic PDE to nonlinear fluid dynamics. The benchmark targets stationary Navier–Stokes turbulence, with the underlying flow fields evaluated under the same held-out conditions for all methods. Spectral discretizations of this class of flows follow standard treatments such as Karniadakis & Sherwin (2005). The divergence residual is evaluated on the unrolled endpoint field as specified in Appendix E.1, the same operator used by the training objective, which keeps the reported residual aligned with the optimized quantity.

We report two complementary quantities. The primary metric is the endpoint divergence residual R_{div}, which measures the incompressibility violation of the generated state. The secondary metric is the spectrum error, measuring spectral agreement with the reference solution over the resolved wavenumber range of the spectral discretization, so the metric covers the scales the model represents.

E.4 Structural Topology Optimization

Following standard density-based topology optimization (Bendsøe et al., 2004), the task replaces a residual-based PDE objective with a downstream engineering objective. Let ρ denote the generated element-density field. The displacement u_{disp} is obtained from

$$K(\rho)u_{\mathrm{disp}} = f_{\mathrm{load}}, \tag{60}$$

where $K(\rho)$ is the density-dependent stiffness matrix and f_{load} is the applied load. The compliance is

$$C(\rho) = f_{\mathrm{load}}^{\top} u_{\mathrm{disp}}, \tag{61}$$

and the physical functional is

$$\Phi_P(\rho) = \log(C(\rho) + \varepsilon_C) + w_V \left(\bar{\rho} - v_f\right)^2, \tag{62}$$

where $\bar{\rho}$ is the mean element density, v_f the prescribed volume fraction, and $w_V = 200$ the volume weight, frozen from a pre-registered gradient-parity rule—the value at which the volume-gradient magnitude at a 5% volume violation equals that of the log-compliance gradient—before any training outcome was observed. Its self-adjoint sensitivity is

$$\frac{\partial C}{\partial \rho_e} = -u_{\mathrm{disp}}^{\top} \frac{\partial K(\rho)}{\partial \rho_e} u_{\mathrm{disp}}, \tag{63}$$

where ρ_e is the density of element e. Batched GPU conjugate-gradient solves provide the elasticity solution and the sensitivity used by the training objective, so that the physics gradient and the reported compliance derive from the same equilibrium state, and no separate surrogate is introduced for either; the equilibrium solve is shared by training and evaluation through the same batched solver.

Evaluation is performed under held-out loads and boundary conditions. We report compliance error E_C as the primary engineering metric, volume violation ΔV as the constraint metric, and topology FD as the distribution-level structural metric, so that engineering quality, constraint satisfaction, and distribution fidelity are each covered, with compliance as the primary quantity.

E.5 Frozen Evaluation Protocol

All quantitative comparisons use frozen checkpoints and fixed task-specific evaluators. Within each task, the same evaluation stream is applied to every method. EMA weights are used for evaluation, and generated samples are obtained with a 20-step deterministic Euler sampler under a sampling seed fixed across all methods, so sampling variability is shared by every comparison.

Aggregate entries in the main paper are reported as mean $\pm$ standard deviation across completed training seeds: five seeds for every method on all three tasks, with all methods within a task sharing the same data split and evaluator. Per-run bootstrap intervals, when used, are computed with 2000 bootstrap samples and serve only as diagnostic uncertainty estimates; they are kept separate from the seed-level variation quantified by the standard deviations reported in the main tables.

The evaluation protocol is deliberately decoupled from training-time diagnostics. Reparameterization statistics and solver-state quantities are analyzed in Appendices B and C; architecture, hardware, timing, and experiment provenance are reported in Appendix G. The long-horizon extension protocol of Appendix E.6 follows the same frozen rules, and its results are consistent with the base-budget comparisons, so the main-table ordering is not an artifact of the stopping point.

E.6 Long-Horizon Extension Protocol

To probe late-stage optimization dynamics, the training schedules of Darcy Flow and Kolmogorov Flow are extended by $1.5\times$, from N_{base} to $1.5\,N_{\text{base}}$ iterations, with the optimizer, learning-rate schedule, batch size, and all remaining hyperparameters held fixed; only the training budget changes. Checkpoints along the extended horizon are evaluated at matched epochs under the frozen protocol of Appendix E.5. On both tasks, the IGFM residual trajectories (R on Darcy Flow, R_{div} on Kolmogorov Flow) decrease monotonically over the full extended horizon, and during the final 30% phase ($1.0\times$–$1.5\times$) the residual continues to decrease smoothly without saturation or backsliding, indicating that the main-table comparisons are not artifacts of stopping training at the base budget.

F Additional Results on the Main Tasks

This appendix provides field-level evidence for the main tasks. The goal is to complement the aggregate metrics in Table 3 with spatial structure and per-sample distributions, together with an ablation on the physical-objective design of the topology task, so that the aggregate numbers can be read against visible field quality, rather than in isolation from the spatial structure.

F.1 Darcy Flow

We first examine whether the reduction in the aggregate Darcy residual is visible at the level of individual fields. Figure 13 shows a representative pressure field together with its pointwise residual map. The same evaluation protocol is used for all methods, and the comparison is made at a deterministic representative sample rather than by manually selecting separate examples for each method; the full-stream residual distributions appear in Figure 14 below on the same frozen evaluation stream, so the single-field and distribution views are directly comparable.

The single-field comparison is consistent with the population-level statistics. Figure 14 reports the empirical distribution of per-sample residuals over the frozen 1024-sample stream. The median residual decreases from 30.5 for FM-OT and 15.2 for PBFM to 4.9 for IGFM.

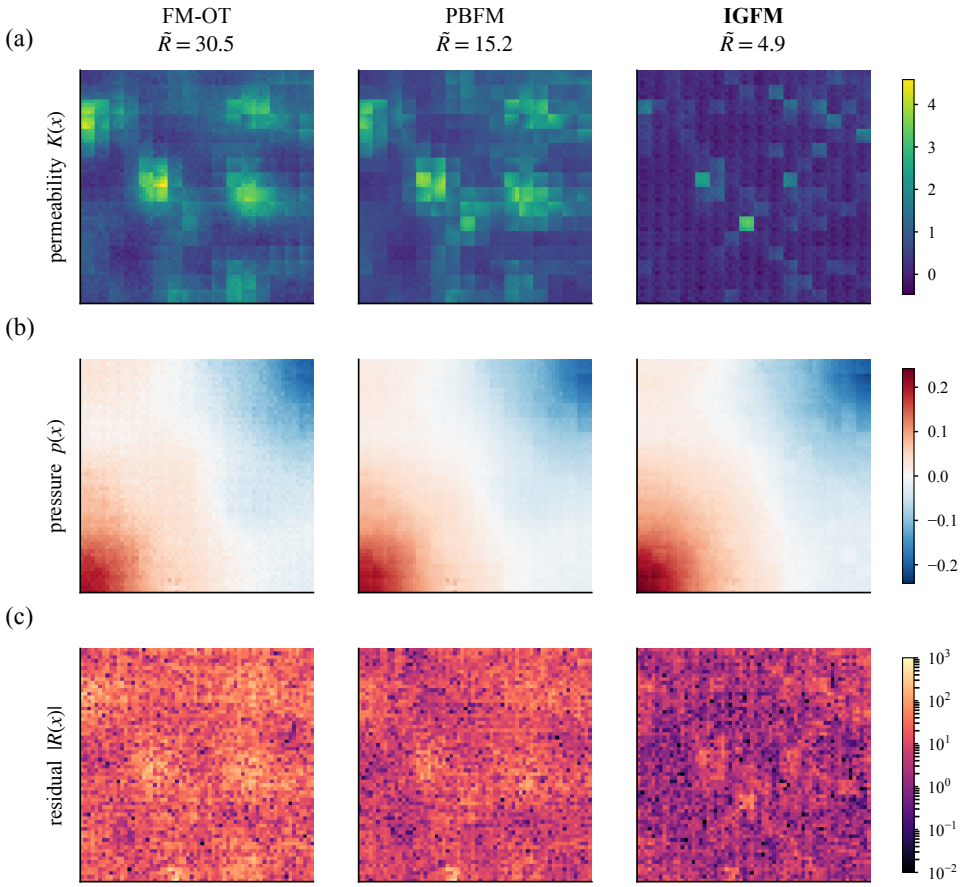


Figure 13: Representative Darcy fields under the frozen evaluator. The same median-residual sample is shown for all methods. IGFM preserves the pressure-field structure while reducing the pointwise residual; the peak $|\mathcal{R}(x)|$ decreases from $538/481$ for the baselines to 91 for IGFM.

These field-level views complement the W_2 comparison in the main text; the distribution metric is computed separately by the frozen evaluator. Figure 15 isolates the effect of physics supervision from sample diversity by pairing samples across methods under a fixed noise stream: each row is generated from the same input noise, so column-wise differences within a row are attributable to the training objective rather than to sampling variability, which the shared color scale makes directly visible, with no per-method scaling of the color range, so intensity differences are comparable across methods and rows of the grid, in keeping with the purpose of the fixed noise stream.

F.2 Kolmogorov Flow

For Kolmogorov Flow, the main benchmark shows that the divergence-residual reduction is accompanied by a nearly unchanged spectrum error. Figure 16 shows the solution domain for a held-out condition selected by the median IGFM residual over the 16 evaluated conditions: the ground truth, generated means, sampling variability, and squared-error fields. The sampling spread of the intrinsic prediction is of the same order as the ground-truth variability across the held-out family.

Figure 17 shows the qualitative comparison referenced in the main text: generated velocity fields at one held-out condition, together with the mean-velocity profiles at three heights. The profiles show where the generated family agrees with, or departs from, the reference distribution over the spatial domain, supporting the interpretation of the aggregate metrics in Table 3: the reduction in the physical divergence residual does not require a visible collapse of the held-out flow-field structure.

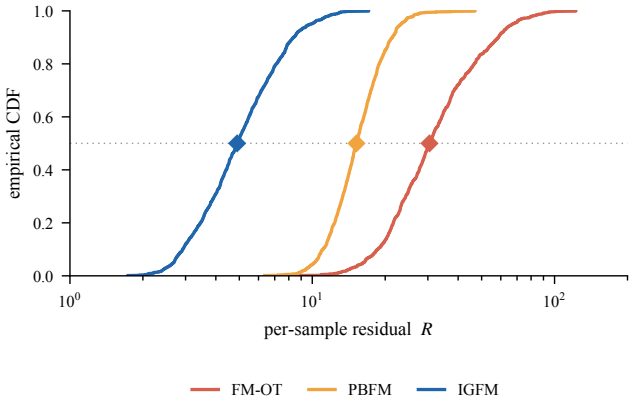


Figure 14: Empirical distribution of per-sample Darcy residuals over the frozen 1024-sample evaluation stream; markers denote the median residual of each method, showing that the reduction is a distribution-level shift rather than the effect of a few favorable samples; the entire empirical distribution shifts over the full frozen stream, with the medians moving consistently.

Figure 18 completes the Kolmogorov evidence with per-condition generated fields against the ground-truth mean, sampled one per held-out condition under the same deterministic selection principle as the Darcy grids above, so neither task involves per-method example selection.

F.3 Structural Topology Optimization

The main comparison uses the task-aligned compliance objective of Appendix E.4. Figure 19 shows the full held-out bridge-type comparison referenced in the main text, together with the equilibrium residual of each design; the ablation below examines the alternative residual objective.

An earlier formulation instead optimized the equilibrium residual R_{eq} while evaluating compliance. At epoch 100 it yields $R_{\text{eq}} = 0.820 \pm 0.085$, 0.231 ± 0.025, 0.245 ± 0.028, and 0.715 ± 0.065 for FM-OT, PBFM-1, PBFM-4, and IGFM, with compliance errors 1.11, 0.68, 2.25, and 1.18, respectively. The disagreement between the residual ranking and the compliance ranking is precisely the objective–utility mismatch that motivated the task-aligned formulation; we retain this experiment as an ablation on physical-objective design rather than a core method comparison.

Figure 20 shows the qualitative behavior of this formulation: physics supervision concentrates the residual on the structure boundary, but the residual ranking does not transfer to compliance.

G Implementation, Computational Cost, and Reproducibility

This appendix collects implementation details needed to reproduce the experiments. Architecture and training settings are reported first; computational-efficiency details are kept together in a single subsection so that engineering optimizations do not interrupt the main scientific narrative.

G.1 Model Architecture and Training Configuration

The backbone is a Diffusion Transformer (DiT) (Peebles & Xie, 2023) operating on 64×64 two-channel fields, implemented in PyTorch (Paszke et al., 2019) with adaLN-Zero conditioning (Peebles & Xie, 2023), SiLU activations (Elfwing et al., 2018), and full softmax attention (Vaswani et al., 2017). The reference Darcy model contains 9.7M parameters. Table 14 consolidates the architecture and default optimization settings. Unless an ablation states otherwise, the Darcy experiments use this configuration; Kolmogorov and Topology Optimization use their task-specific data and evaluators while retaining the same comparison principle described in Appendix E, so that the cross-task conclusions are not artifacts of a single task pipeline or of task-specific tuning.

All experiments were conducted on a single NVIDIA H100 80GB GPU with a global batch size of 64 under single-process data parallelism, without gradient accumulation. The software environ-

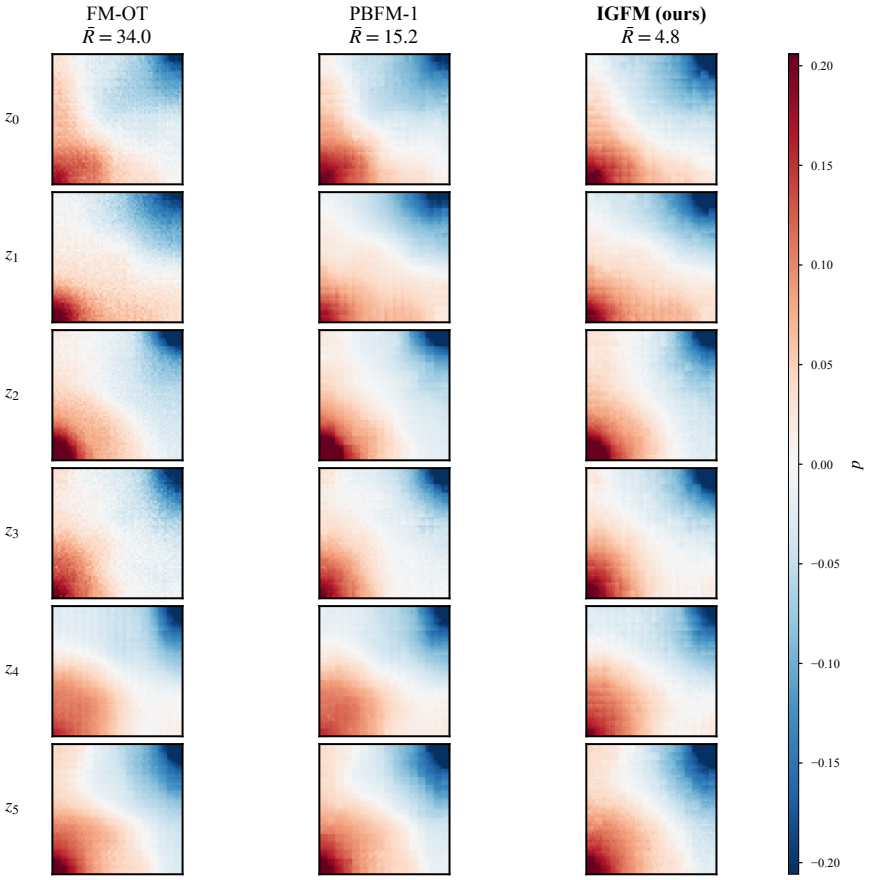


Figure 15: Generated Darcy pressure fields under a fixed noise stream (rows z_0–z_5, the first six of the 24-sample frozen stream; same row = same noise) for FM-OT, PBFM-1, and IGFM. Shared color scale; column headers report the mean per-sample residual over the full stream. The distribution distance W_2 is quantified separately in Table 1, which places it alongside the residual side of the same trade-off, for the same methods and evaluation stream, so that both sides of the fidelity–residual comparison come from one protocol and a single frozen evaluation stream.

Table 14: Reference Darcy architecture and default training configuration.

architecture		training	
input resolution	64×64	optimizer	AdamW
patch size	4	learning rate	3×10^{-5}
channels	$2/2$	global batch size	64
hidden size	256	training horizon	100 epochs
depth / heads	$8/8$	EMA decay	0.999
		CG depth / damping	$5/10^{-3}$
		functional step rule	projected step-length rule
		physics time weight	$w(t) = t$ (β=1)

ment uses Ubuntu 22.04, CUDA 12.8, PyTorch 2.11, and Python 3.10, identical across all compared methods, and the same environment runs every method and ablation, including the reparameterization and solver-dynamics audits of the preceding appendices and the timing audit of Section G.2.

Composite Direction, Step-Length Rule, and Optimizer Interface. The per-objective solves of Eq. equation 8 already realize parameter-space lifts of the projected representatives, so the composite direction is assembled directly from them: with $A := J^\top M J + \lambda I$ and $s_i^2 := \xi_i^\top A \xi_i$, the

implementation combines the two solves in the unit-A-norm basis,

$$\xi = \tfrac{1}{2}(\xi_D/s_D + \xi_P/s_P)\,, \tag{64}$$

where the minimum-norm simplex weights reduce to the bisector for two objectives, so no separate least-squares solve is performed. Since $s_i^2 = \|J\xi_i\|_M^2 + \lambda\|\xi_i\|_2^2$, the unit-$A$-norm basis coincides with the unit-M-norm basis of the output images as $\lambda \to 0$, where Eq. equation 64 is exactly the lift of the normalized bisector of the projected representatives; at the finite damping used in training it is a damped-metric approximation of that bisector, and the direction-fidelity audit of Appendix D uses a near-undamped (λ=10^{-6}) reference of the same construction. The implementation works on the ascent side: the output image $J\xi$ follows the bisector of the projected representatives in this damped normalization, the ascent counterpart of the ideal direction $d^\star$ of Eq. equation 7. Writing $\hat{\xi} := \xi/\|\xi\|_2$ for the unit-Euclidean-norm direction, the projected step-length rule sets, at optimization step j,

$$\ell_j = \max\Big\{\tfrac{1}{2}\,(r_D + r_P)^\top\,\hat{\xi},\ 0\Big\}\,, \tag{65}$$

so the step carries the average first-order decrease of the two objectives and is clipped at zero rather than reversed when they oppose the update. The scaled direction $\ell_j\hat{\xi}$ is stored as the parameter gradient, and the AdamW step subtracts it, with optimizer state carried across steps as in standard training, so the realization is the one analyzed in Appendix C. The damping of Eq. equation 8 is applied on the curvature scale of the data gradient, $\lambda = \lambda_{\mathrm{rel}}\,(r_D^\top J^\top M J\, r_D)/(r_D^\top r_D)$ with $\lambda_{\mathrm{rel}} = 10^{-3}$, a scale-invariant choice under rescaling of the objectives, as audited in Appendix D.4.

G.2 Computational Cost

PBFM Overhead as Originally Reported. The original PBFM report attributes its training overhead to two sources (Baldan et al., 2026): incorporating the residual loss with ConFIG requires an additional backward pass, and unrolling the denoising steps amplifies it further. In their Darcy measurements (A100, batch size 64), one unrolled step costs $1.56\times$ unconstrained flow matching in wall-clock time (6.69×10^{-2} vs 4.28×10^{-2} s/iteration), and four unrolled steps reach $2.38\times$ PBFM-1 in time and $3.5\times$ in peak memory (44.15 GB vs 12.62 GB). These overheads are restricted to training; inference speed is unaffected, since the sampler is identical for all methods.

Our Reproduced Baselines. Our PBFM baselines use the shared DiT backbone and training stack of Appendix G.1: PBFM-1 runs at $1.00\times$ FM-OT in wall-clock time, and PBFM-4 at $1.58\times$ with $1.31\times$ peak memory (Table 15). The two columns of Table 15 are reported for background and are not directly comparable: the official values were measured per iteration on A100 hardware with the original training stack, whereas ours were measured on H100 with our reproduction, in which the residual-loss backward pass is computed through the shared computation graph alongside the flow-matching backward. Our reproduced PBFM-4 follows the original curriculum, raising the unrolling depth from one step to two, three, and four at epochs 26, 51, and 76 (quartiles of the 100-epoch budget), so per-epoch averages for PBFM-4 aggregate over the depths visited in the measurement window; the last-50-epoch timing window (epochs 51–100) therefore averages depths three and four, the final quarter of training being at the full depth. All IGFM overheads below are normalized by our own PBFM-1 per-epoch wall-clock measurement on the same single-GPU configuration, so the IGFM comparison is unaffected by baseline-implementation differences.

IGFM Runtime and Memory. The tangent projection is the dominant additional cost of IGFM: each update evaluates chained Jacobian-vector and vector-Jacobian products through the backbone.

Table 15: PBFM training overhead, *official* (values from the PBFM report, A100, batch size 64, Darcy) versus *reimpl.* (our baselines on the shared DiT stack of Appendix G.1, H100). The two columns differ in hardware and training stack, and are therefore not directly comparable.

	official (A100)	reimpl. (H100)
PBFM-1 wall-clock vs FM-OT	$1.56\times$	$1.00\times$
PBFM-4 wall-clock vs PBFM-1	$2.38\times$	$1.58\times$
PBFM-4 peak memory vs PBFM-1	$3.5\times$	$1.31\times$

Table 16: Runtime and memory of the $k = 5$ realization and its compiled variants on a single H100 GPU. Timing runs use the same model and batch configuration (64), report last-50-epoch means, and are normalized by PBFM-1 (17.20 s/epoch). Kernel fusion and TorchCompile are mathematically exact, so the check residual is unchanged across all four IGFM rows, as the identical entries confirm.

configuration	s/epoch	latency	memory (GB)	check $R\downarrow$
PBFM-1	17.20	1.00×	22.4	18.42 ± 1.20
IGFM (k=5, uncompiled)	20.30	1.18×	28.6	6.93 ± 1.24
+ custom fused kernel	18.92	1.10×	28.0	6.93 ± 1.24
+ TorchCompile	18.06	1.05×	24.1	6.93 ± 1.24
+ fused kernel + TorchCompile	17.54	1.02×	23.5	6.93 ± 1.24

Table 16 reports a timing audit under the same single-GPU configuration (batch size 64; last-50-epoch means; PBFM-1 = 17.20 s/epoch). The uncompiled $k = 5$ realization costs 1.18× PBFM-1 in wall-clock time and 27.7% more peak memory (28.6 GB vs 22.4 GB). Two engineering steps reduce this. TorchCompile (PyTorch 2.0+, max-autotune) compiles the JVP/VJP chains and eliminates the memory-bandwidth bottleneck of the chained products, reaching 1.05× wall-clock and 24.1 GB. A custom fused kernel reduces memory traffic further; combined with TorchCompile the cost is 1.02× with 23.5 GB peak memory, within 4.9% of the PBFM-1 footprint. All experiments in the main text use this fused-and-compiled configuration. Both steps are mathematically exact reorganizations of the same computation graph, so the trained model, its five-seed results, and the convergence curves are unchanged ($R = 6.93 \pm 1.24$). Wall-clock cost grows with the tangent-solve depth: 1.06×, 1.09×, 1.13×, 1.15×, 1.18×, 1.27× at $k = 1, 2, 3, 4, 5, 8$, matching the depth sweep of Appendix D, where the residual improvement saturates at the same default depth.

The adaptive tangent solve discussed in Appendix C is not used as the default implementation. In the recorded timing audit it reduces the mean number of CG iterations but incurs additional stopping-rule overhead; its role in the paper is to analyze the difference between local solve quality and long-horizon trajectory quality, with its stopping-rule overhead measured in the audit above.

G.3 Evaluation and Statistical Protocol

All quantitative comparisons use the frozen evaluation and statistical protocol specified in Appendix E.5, including seed counts, bootstrap diagnostics, and the frozen evaluator configuration.

G.4 Experiment Provenance

Each reported result is associated with a frozen checkpoint, task-specific evaluator, data split, sampling configuration, and training seed. Reparameterization audits additionally record the parameter transformation and function-identity check; solver-dynamics experiments record functional-travel and realization diagnostics; computational-cost measurements are run without the instrumentation used by those audits, so that instrumentation overhead does not contaminate the reported wall-clock timings or the peak-memory measurements of the cost audit in Appendix G.2.

H Additional Related Work

This appendix collects the broader literature coverage omitted from Section 2 to keep the main-text narrative focused on the line of work closest to IGFM: the problem formulation, the classical Pareto construction it inherits, and the realizable function-space geometry it changes.

H.1 Physics-Informed PDE Learning

Physics-informed neural networks incorporate governing equations through residual losses (McClenny & Braga-Neto, 2023), and the optimization difficulty of competing loss terms has been analyzed through convergence analyses and neural-tangent-kernel perspectives (Shin et al., 2020; van der Meer et al., 2022), with scale-aware optimization as a mitigation (Yao et al., 2023). A broad family of deep-learning methods approximates parameterized physical systems directly, from graph-based and neural-solver discretizations to reusable surrogates (Gao et al., 2022; Han et al.,

2018; Beck et al., 2023; Cuomo et al., 2022; Brunton & Kutz, 2024; Fresca et al., 2021; Chen et al., 2021; Wu et al., 2024). These works optimize a single physical objective or a weighted sum, and do not treat distribution matching as a simultaneous objective, which is precisely the setting that our two-objective formulation addresses and that the experiments of the main text evaluate.

H.2 Generative Modeling for Physical Systems

Generative formulations extend to probabilistic surrogates and score- and diffusion-based methods for flow reconstruction, inverse problems, PDE-conditioned generation, physical design, turbulent-field recovery, and distributional fluid-state modeling (Rixner & Koutsourelakis, 2021; Holzschuh et al., 2023; Jacobsen et al., 2025; Huang et al., 2024a; Zhang & Zou, 2025; Yuan et al., 2025; Nie et al., 2021; Li et al., 2023; Sardar et al., 2024; Liu & Thuerey, 2024; Valencia et al., 2025; Guastoni & Vinuesa, 2025; Buehler, 2022; Düreth et al., 2023; Vlassis & Sun, 2023; Mazé & Ahmed, 2023), with diffusion-based surrogates for turbulent flow and weather prediction as representative applications (Lienen et al., 2024; Wang et al., 2023; Giannone et al., 2023; Bastek & Kochmann, 2023). Constrained synthesis has also been pursued within diffusion models through feasibility-preserving gauge maps for constrained generation (Li et al., 2026), grey-box dynamics matching with physics-informed latent structure (Singh et al., 2026), repeated constraint-guided updates to the sampling trajectory (Huang et al., 2024b), and dual training for constrained distribution optimization (Khalafi et al., 2024). Flow-matching models have further been used for physics-guided generative design through gradient-based optimization in source space (Yang et al., 2026), a complementary way to inject downstream physical objectives into generation. These methods differ in where physics enters the pipeline; none of them defines multi-objective conflict over the realizable function-space tangent, which is the construction that IGFM introduces and validates across the three tasks.

H.3 Multi-Task and Multi-Objective Optimization

Beyond the methods discussed in Section 2, impartial multi-task learning balances losses through reweighting (Liu et al., 2021b), and loss balancing within physics-informed learning has also been formulated explicitly as a multi-objective problem (de Wolff et al., 2022). Related work studies gradient homogenization and adaptive weighting (Javaloy & Valera, 2022; Liu et al., 2023), Jacobian-based descent formulations (Quinton & Rey, 2024), and resolution of inconsistent task signals by dropping gradient components according to sign agreement (Chen et al., 2020). All of these operate on parameter gradients in Euclidean coordinates; IGFM is complementary to them in the same way it is complementary to ConFIG and PBFM: it changes the space in which conflict is measured.

H.4 Function-Space and Invariant Training Objectives

Beyond the update rules discussed in Section 2, function-space perspectives have been used to measure learning dynamics and training progress through induced output changes rather than parameter displacement (Scannell et al., 2024; Milsom et al., 2025). These lines of work share our motivation of optimizing what the network computes rather than how it is parameterized, but they address single-objective supervision or diagnosis. The combination studied here—defining a multi-objective Pareto conflict over the realizable function-space tangent of a parameterized model, with chart and scale invariance of the resulting field—is, to the best of our knowledge, new.

The geometric optimizers are the closest relatives and merit an explicit contrast. Natural-gradient methods endow the parameter space with the Fisher information of the model distribution, obtaining a single-objective geometry whose pullback form is invariant under reparameterizations of the model (Amari, 1998; Martens & Grosse, 2015); Jacobian descent generalizes gradient descent to vector-valued objectives, but its aggregators combine the per-objective parameter gradients—the rows of the parameter Jacobian—so the resulting update remains a chart-dependent linear combination of parameter gradients, with the Gramian serving only as an efficiency device (Quinton & Rey, 2024). IGFM differs from both along the two axes that matter here: it is multi-objective, and it defines the geometry of objective interaction directly over the realizable changes of the model output for a fixed observation map and metric, so the representation of each objective is chart-invariant by construction rather than through a metric induced on the parameters themselves.

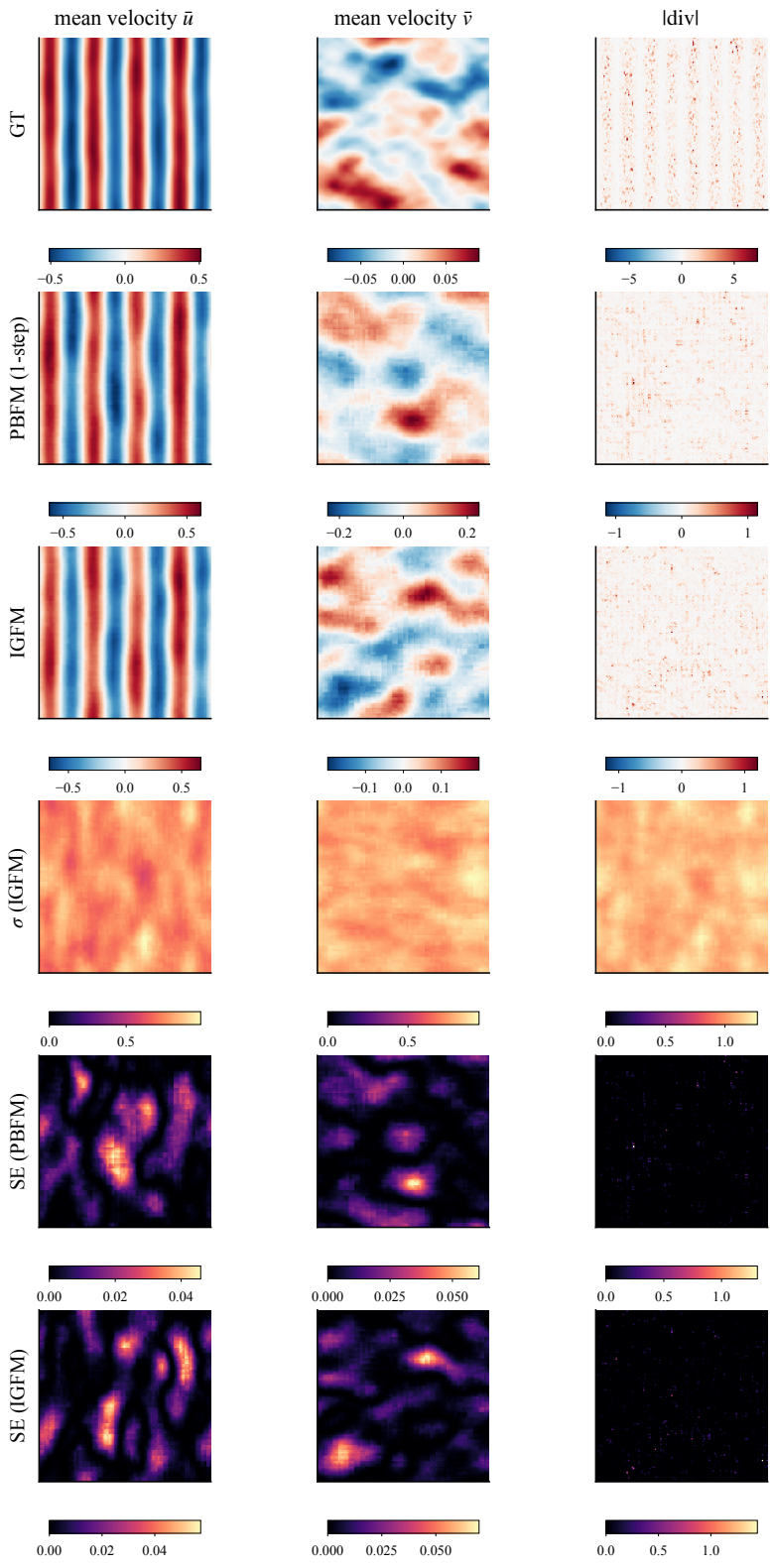


Figure 16: Kolmogorov solution-domain comparison for the held-out condition with median IGFM residual. Rows: reference field, generated means, sampling variability, and squared-error maps for the mean velocity components and divergence, under the frozen evaluation protocol.

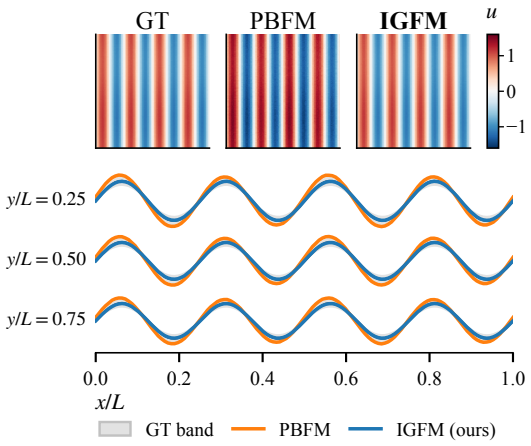


Figure 17: Kolmogorov qualitative comparison. Generated velocity fields at one held-out condition, with mean-velocity profiles at three heights; the band is the across-condition ground-truth range, and a generated profile within the band agrees with the reference variability.

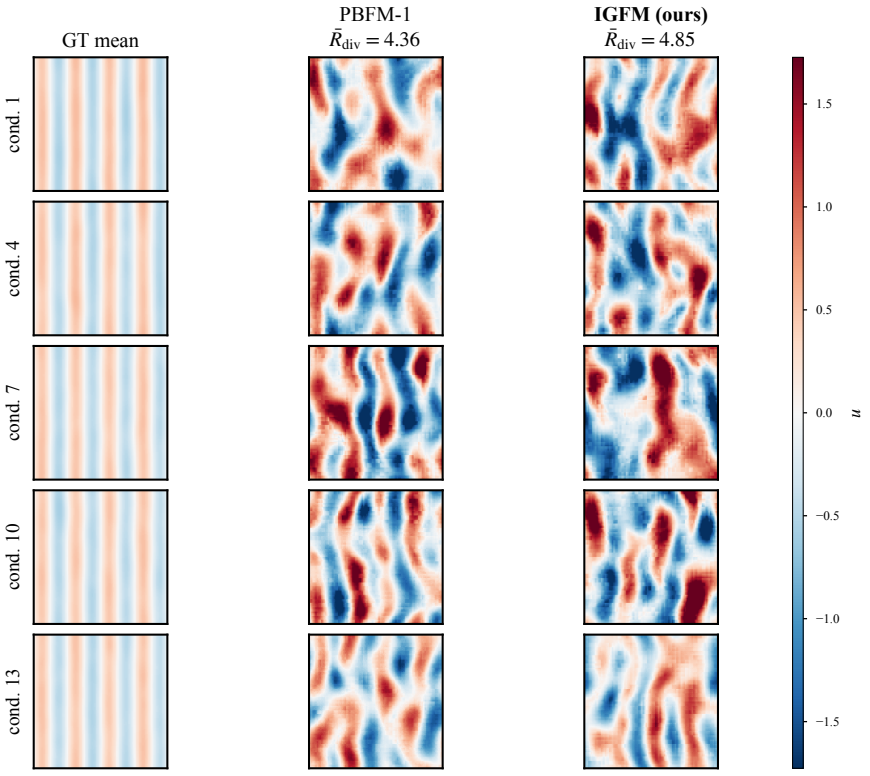


Figure 18: Generated Kolmogorov u-fields for held-out flow conditions (rows: conditions 1/4/7/10/13 of the 16 in the dump, a deterministic choice; one generated sample per condition) against the ground-truth mean field. Shared color scale; column headers report the mean single-sample divergence residual $|\mathrm{div}|$ over the 16 conditions. The instantaneous fields of the two methods are visually comparable, consistent with these near-tied single-sample statistics.

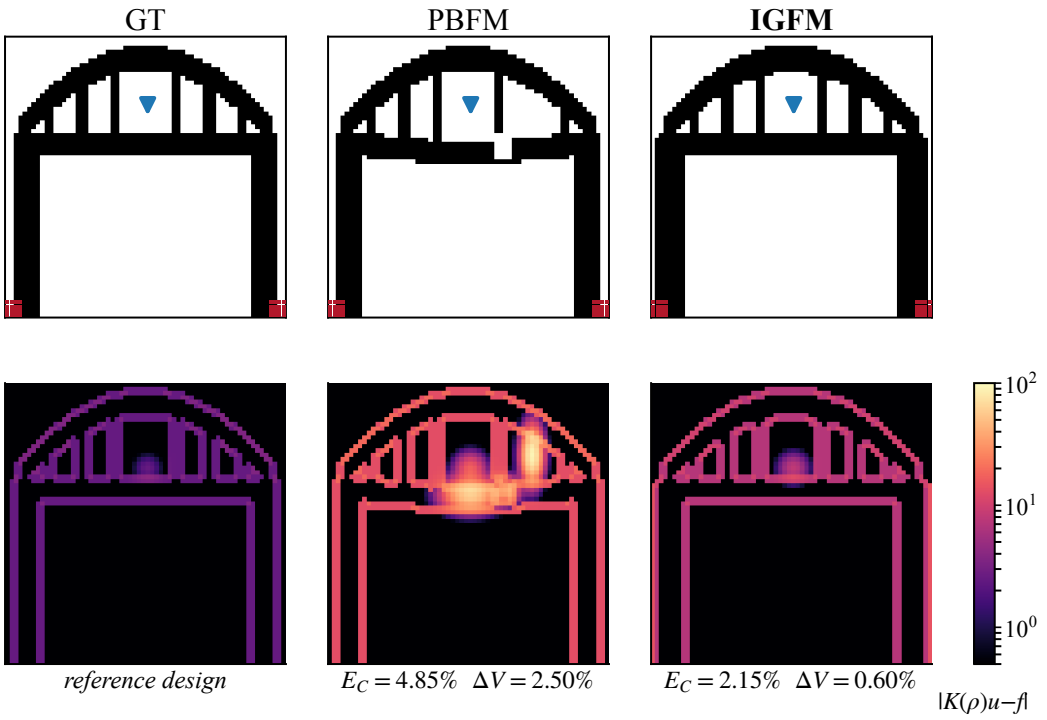


Figure 19: IGFM predicts cleaner held-out designs under a bridge-type load case. Top: predicted topologies for a tied-arch bridge under a midspan deck load (red: supports; blue arrow: load). Bottom: equilibrium residual $|K(\rho)u - f|$ of each design (shared log colorbar), annotated with compliance error E_C and volume violation ΔV. PBFM yields a sagging deck, a missing hanger, and a spurious hole; IGFM stays close to the reference design; the shown load case is among the held-out conditions, so the designs test transfer rather than memorized cases.

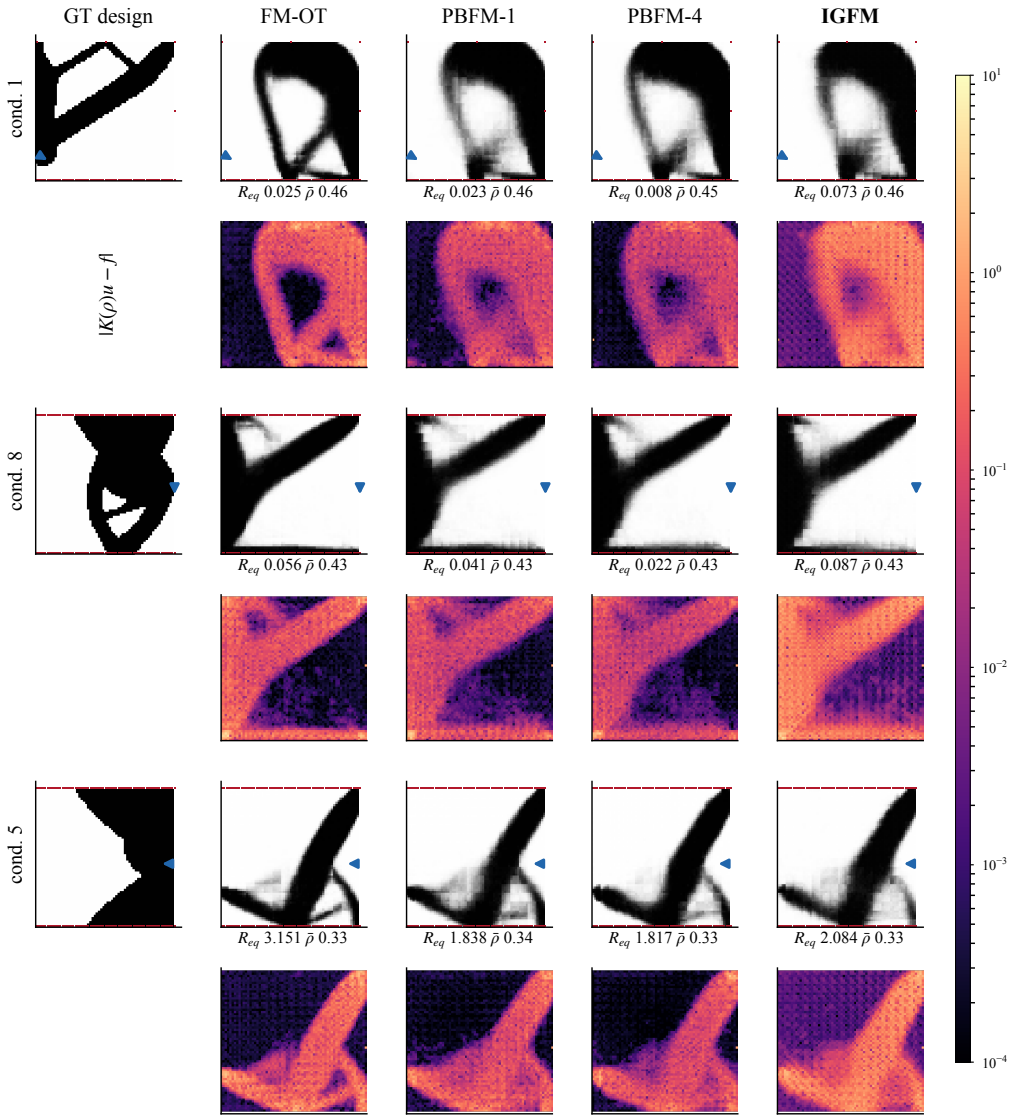


Figure 20: Generated topology designs under the original equilibrium-residual objective (rows: held-out conditions at the median, 75th percentile, and worst R_{eq} under PBFM-4, a deterministic choice; columns: ground-truth design, then design and pointwise residual $|K(\rho)u - f|$ per method; epoch 100, per-condition mean over 128 samples, frozen sampling protocol). Labels under each design: equilibrium residual R_{eq} and volume fraction $\bar{\rho}$ of the binarized mean design. The residual maps concentrate on the structure boundary, yet the residual ranking does not transfer to compliance, which motivated the task-aligned objective used for the main-text comparison of Figure 19.