CHARACTERISTIC-STEERED FLOW MATCHING FOR ORACLE-DEFINED SETS

Anonymous authors
Paper under double-blind review

ABSTRACT

Generating samples that satisfy complex physical or structural constraints remains a fundamental challenge when valid regions cannot be expressed in closed form. For instance, finding valid candidates, such as a stable crystal structure or a collision-free robot trajectory, fundamentally relies on evaluating proposals and rejecting failures against a binary verifier. This problem can be expressed as learning distributions over *oracle-defined sets*, where admissibility is accessible only through pointwise verification. We study how to adapt continuous generative flows in this regime, where certified reference samples seed a base transport, but further constraint satisfaction must be learned from oracle evaluations that provide no gradient feedback. We introduce *Characteristic-Steered Flow Matching* (CSFM), a method that converts discrete membership evaluations into directional supervision along generative paths. By pairing each inadmissible candidate with a certified feasible anchor, CSFM transports the required terminal correction backward along the generative trajectory via reverse-time linearization, supervising the velocity field to realize first-order endpoint corrections without differentiating the oracle. Across synthetic geometric sets, physics-constrained continuous control, and long-horizon trajectory planning, CSFM achieves favorable validity–coverage and downstream trade-offs relative to existing flow-correction baselines, preserving coverage and distributional fidelity as ambient dimension scales.

1 INTRODUCTION

Across scientific and engineering disciplines, generative models are increasingly deployed in domains where candidate outputs must satisfy strict operational constraints. Generated molecules must exhibit chemical stability and binding selectivity, physical structures must satisfy exact geometric tolerances, and robot trajectories must remain collision-free under complex dynamical limits (De Santi et al., 2026; Gutjahr et al., 2026; Zeni et al., 2025). In these settings, admissibility is frequently governed not by a differentiable objective, but by a binary, pointwise verification procedure or membership oracle that simply confirms whether a candidate satisfies all criteria.

Continuous flows offer simulation-free training and regular, invertible probability paths (Chen et al., 2018; Lipman et al., 2023; Liu et al., 2023), making them natural candidates for constrained generative modeling. However, adapting them to non-differentiable boundaries is non-trivial: a binary rejection confirms an invalid endpoint but provides no gradient to guide trajectory correction. Standard guidance and flow adaptation frameworks require differentiable penalties and costly solver adjoints (Chen et al., 2018; Domingo-Enrich et al., 2025)—both unavailable under a boolean oracle. Moreover, naive adaptation can collapse coverage or distort the latent-to-output map, compromising downstream tasks. This raises two fundamental questions: (1) how to extract a feasible terminal displacement from a non-differentiable oracle, and (2) how to distribute that correction along the generative trajectory?

To address both challenges, we introduce *Characteristic-Steered Flow Matching* (CSFM), circumventing the need for solver differentiation or differentiable surrogates. Our method first establishes an explicit geometric correspondence by pairing each inadmissible terminal sample with a certified feasible anchor, optionally refining the pair toward boundary crossings via chord bisection in logarithmic queries. At training time, CSFM transports this terminal displacement backward along the characteristic trajectory via a reverse-time Jacobian–vector product (JVP); regressing the velocity

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.

field toward these transported adjustments distributes the correction across the trajectory to first order within standard flow matching objectives. At inference time, these certified correspondences serve as a lightweight projection operator, mapping residual boundary violations back to certified states to guarantee admissible sample generation without distorting the global transport map.

We evaluate CSFM across synthetic geometric benchmarks, physics-constrained continuous control, and long-horizon trajectory planning. Across these tasks, CSFM retains coverage and distributional fidelity in high dimensions where baselines collapse, while requiring an order of magnitude fewer oracle evaluations. When used for action and trajectory representations, it preserves chart regularity and improves downstream performance.

Our primary contributions are:

- We derive a first-order transport of terminal corrections into velocity regression targets for flow matching through reverse-flow Jacobian–vector products.
- We construct corrections from certified anchors for both training and runtime repair, with an optional latent-distance cost that makes correspondence selection sensitive to the flow's latent coordinates.
- Across geometric generation, control, and planning, we show that oracle acceptance must be assessed alongside coverage and downstream performance. CSFM improves acceptance while retaining useful generative structure, including as ambient dimension increases.

2 Problem Formulation and Background

We investigate how to steer continuous generative flows toward elements that satisfy an oracle-defined constraint while preserving the regularity of the underlying transport. Because continuous flow architectures define diffeomorphic mappings, they inherently struggle to represent discontinuous density thresholds and sharp boundary cuts, even when enough target samples are available during training. When the geometry or topology of the admissible set is complex, enforcing exact constraint satisfaction and high distributional coverage forces the flow into extreme local expansion and shear (Cornish et al., 2020). We therefore seek to improve oracle satisfaction and coverage while avoiding unnecessary local distortion of the generative transport.

2.1 Problem Definition

Let $s \in \mathcal{S}$ denote a conditioning variable and let $\mathcal{A} \subset \mathbb{R}^d$ denote the space of elements to be generated. We assume access to a deterministic binary membership oracle $\mathcal{O}_s$ that defines the measurable admissible set $\mathcal{C}_s$:

$$\mathcal{O}_s : \mathcal{A} \to \{0, 1\}, \qquad \mathcal{C}_s := \{a \in \mathcal{A} : \mathcal{O}_s(a) = 1\}, \qquad 0 < \mathrm{Vol}(\mathcal{C}_s) < \infty. \tag{1}$$

Following De Santi et al. (2026), strong oracles characterize validity exactly, whereas weak oracles check necessary conditions whose satisfaction need not establish validity. Our primary target is the uniform law over $\mathcal{C}_s$, $p_s^\star := \mathrm{Unif}(\mathcal{C}_s)$, with density

$$p_s^\star(a) = \frac{\mathcal{O}_s(a)}{\mathrm{Vol}(\mathcal{C}_s)}, \qquad \mathrm{Vol}(\mathcal{C}_s) = \int_{\mathcal{A}} \mathcal{O}_s(a)\, da. \tag{2}$$

Membership defines the support but does not determine a probability law on it; uniformity is an explicit modeling choice.[1] We identify the source point with a latent variable $z = x_0$ drawn from a fixed reference distribution q on a latent domain $\mathcal{Z} \subset \mathbb{R}^d$. We seek a conditional deterministic generative transport

$$T_\theta : \mathcal{S} \times \mathcal{Z} \to \mathcal{A} \tag{3}$$

such that, for each condition s, its induced pushforward $p_\theta(\cdot \mid s) = \big(T_\theta(s, \cdot)\big)_* q \approx p_s^\star$.

[1] In some settings, one may instead wish to approximate a non-uniform target distribution, for instance to reflect a preference over admissible solutions. We evaluate this case in Suite C through imitation learning from certified expert demonstrations.

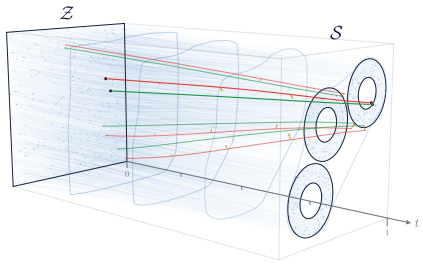


Figure 1: **Characteristic steering overview.** Latents $z \in \mathcal{Z}$ map along generative paths (blue) to candidate endpoints in target space $\mathcal{A}$. A constraint-violating sample $x^- \notin \mathcal{C}_s$ (red) is paired with a certified anchor $x^\star \in \mathcal{C}_s$ (green); the terminal error $r = x^\star - x^-$ is pulled backward along the characteristic to supply first-order velocity targets (orange).

2.2 Continuous Generative Flows

We realize the transport T_θ as the terminal map of a conditional continuous flow Φ^θ generated by a velocity field $v_\theta(x, t, s) \in \mathbb{R}^d$, and learn this field by Flow Matching (Lipman et al., 2023) along rectified paths between coupled source and target samples:

$$\dot{x}_t = v_\theta(x_t, t, s), \qquad x_0 = z \sim q, \qquad x_1 = T_\theta(s, z) = \Phi^\theta_{0\to 1}(z; s). \tag{4}$$

Specifically, let $\pi_s \in \Pi(q, p^\star_s)$ couple the source and target distributions and $(x_0, x_1) \sim \pi_s$. We use the affine path $x_t = (1-t)x_0 + tx_1$ with constant conditional velocity $u_t = x_1 - x_0$ (Albergo & Vanden-Eijnden, 2023; Liu et al., 2023), and optimize

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \mathbb{E}_{\substack{(x_0,x_1)\sim\pi_s \\ t\sim\mathcal{U}[0,1]}} \left[\|v_\theta(x_t, t, s) - (x_1 - x_0)\|_2^2 \right]. \tag{5}$$

We construct π_s between source and certified-target minibatches by entropically regularized optimal transport under squared Euclidean cost using the Sinkhorn algorithm (Cuturi, 2013; Tong et al., 2024). Assuming sufficient regularity of v_θ, the resulting ODE admits unique solutions and induces differentiable invertible flow maps $\Phi^\theta_{t_a\to t_b}$, with

$$x_{t_b} = \Phi^\theta_{t_a\to t_b}(x_{t_a}; s), \qquad \Phi^\theta_{t_b\to t_a} = \left(\Phi^\theta_{t_a\to t_b}\right)^{-1}, \qquad J^\theta_{t_a\to t_b}(x_{t_a}; s) := \frac{\partial \Phi^\theta_{t_a\to t_b}(x_{t_a}; s)}{\partial x_{t_a}}. \tag{6}$$

3 Method

Although a binary oracle provides only a membership decision, every accepted query yields an unequivocally certified point in $\mathcal{C}_s$. Connecting a rejected sample to a certified anchor forms a straight chord guaranteed to intersect the boundary $\partial\mathcal{C}_s$. We exploit this geometry by pairing rejected endpoints with feasible anchors, optionally localizing boundary crossings via chord bisection to establish tight training targets or provide certified runtime projections (Section 3.1).

3.1 Certified Correspondences

Let $\mathcal{X}^+_s \subset \mathcal{C}_s$ denote the pool of available certified anchors (Assumption 1). We define a correspondence operator $P_s : \mathcal{A} \to \mathcal{C}_s$ that acts as the identity on certified points and projects rejected endpoints $x^- \in \mathcal{X}^-_s \subset \mathcal{A} \setminus \mathcal{C}_s$ onto $\mathcal{C}_s$ by

$$x^\star := P_s(x^-) \in \arg\min_{x^+\in\mathcal{X}^+_s} \left\{ \|x^+ - x^-\|_2^2 + \kappa \, \|\bar{\Phi}_{1\to 0}(x^+; s) - \bar{\Phi}_{1\to 0}(x^-; s)\|_2^2 \right\}.^2 \tag{7}$$

The first term encourages local spatial corrections, while the second encourages neighborhood locality in the latent coordinates of the reference flow. Optionally, $x^\star$ can be refined to a certified boundary point along the chord $[x^-, x^\star]$ via oracle bisection (Appendix A.1). At inference time, composing the generator with this operator as $P_s \circ T_\theta$ guarantees oracle-accepted generation under Assumption 1.

[2]Unless stated otherwise, we set $\kappa = 0$ during training and $\kappa > 0$ at inference.

3.2 CHARACTERISTIC STEERING

Let $r := x^\star - x^-$ denote the terminal displacement from a rejected endpoint to its certified correspondence. Characteristic steering determines how this displacement is realized across the entire generation trajectory. Rather than backpropagating through a numerical ODE solver, we pose this correction as an endpoint-control problem along the reference trajectory and resolve it in closed form.

Reference reverse characteristics. Let $\bar{v}$ denote a reference velocity field with induced flow $\bar{\Phi}$. A rejected endpoint $x^- = \bar{\Phi}_{0\to 1}(z; s) \notin \mathcal{C}_s$ originates from a latent state z along the reference characteristic $x_t := \bar{\Phi}_{1\to t}(x^-; s)$ for $t \in [0, 1]$. The reverse flow sensitivity $A_t := \bar{J}_{1\to t}(x^-; s)$ transports the terminal displacement backward along this characteristic:

$$r_t := A_t\, r = \bar{J}_{1\to t}(x^-; s)\,(x^\star - x^-). \tag{8}$$

Since $A_t^{-1} = \bar{J}_{t\to 1}(x_t; s)$, it follows that $\bar{J}_{t\to 1}(x_t; s)\, r_t = r$ for all $t \in [0, 1]$. Hence, r_t represents the exact perturbation at time t that propagates forward to r at $t = 1$. Computationally, evaluating r_t along a discretized characteristic requires only a single Jacobian–vector product (JVP) with tangent r through the frozen reverse integrator.

Steering as endpoint control. Consider perturbing the reference trajectory by an additive control c_t, giving $\dot{y}_t = \bar{v}(y_t, t, s) + \alpha\, c_t$ with $y_0 = z$. To first order in α, the terminal state variation responds linearly through the forward sensitivity:

$$\delta x_1 = \int_0^1 \bar{J}_{\tau\to 1}(x_\tau; s)\, c_\tau\, d\tau. \tag{9}$$

Setting $c_\tau = A_\tau r$ yields $\bar{J}_{\tau\to 1}(x_\tau; s)\, c_\tau = r$ identically for all τ, which integrates directly to $\delta x_1 = \int_0^1 r\, d\tau = r$. Consequently, the perturbed endpoint satisfies $y_1 = x^- + \alpha r + o(\alpha)$ as $\alpha \to 0$ (Appendix A.2). More generally, any temporal weighting $c_t = w(t) A_t r$ normalized such that $\int_0^1 w(t)\, dt = 1$ attains the exact same first-order displacement r. We set $w \equiv 1$, distributing the terminal correction uniformly across integration time (see Appendix B for non-uniform allocations).

Velocity regression targets. Rather than executing an online control loop, we train the active network to internalize this steering behavior. Because x_t, r_t, and $\bar{v}$ are evaluated along the frozen reference characteristic, the steered trajectory yields a static regression target detached from parameter graphs. For a scaling coefficient $\alpha > 0$, the target velocity field along the path is defined as

$$u(t) := \bar{v}(x_t, t, s) + \alpha\, r_t. \tag{10}$$

3.3 JOINT TRAINING OBJECTIVE

We optimize θ under a composite objective balancing target generation and constraint enforcement:

$$\mathcal{L}(\theta) = \mathcal{L}_{\mathrm{FM}}(\theta) + \lambda\, \mathcal{L}_{\mathrm{CSFM}}(\theta), \qquad \lambda > 0. \tag{11}$$

The first term, $\mathcal{L}_{\mathrm{FM}}$, is the standard Flow Matching loss equation 5, evaluated on certified targets $x_1 \sim p_s^\star$ coupled to uniform latents $x_0 \sim q = \mathrm{Unif}([-1, 1]^d)$[3] via the minibatch Sinkhorn plan π_s.

The second term, $\mathcal{L}_{\mathrm{CSFM}}$, penalizes constraint violations by regressing the active field onto the steered velocity targets of rejected trajectories. For each rejected endpoint $x_i^- \notin \mathcal{C}_{s_i}$ with certified correspondence $x_i^\star = P_{s_i}(x_i^-)$, reverse characteristic $x_{i,t}$, and target velocity $u_i(t)$:

$$\mathcal{L}_{\mathrm{CSFM}}(\theta) = \mathbb{E}_{i,\, t\sim\mathcal{U}[0,1]} \left[\| v_\theta(x_{i,t}, t, s_i) - \mathrm{sg}[u_i(t)] \|_2^2 \right], \tag{12}$$

where $\mathrm{sg}[\cdot]$ denotes the stop-gradient operator. Gradients flow exclusively through the active network v_θ; the reference states $x_{i,t}$ and target velocities $u_i(t)$ remain frozen.

[3] Pairing uniform source and target distributions promotes uniform volume scaling over bounded coordinates.

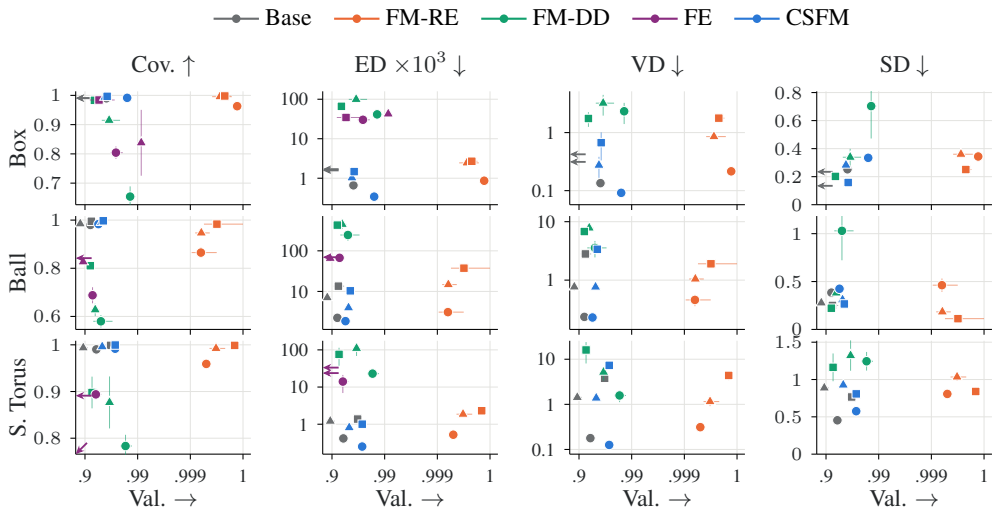


Figure 2: **Geometric benchmark** ($m = 3$). Val. versus Cov., ED ($\times 10^3$, log scale), VD, and SD for Box, Ball, and Solid Torus. Markers denote dimension: $\circ$ ($d = 2$), $\triangle$ ($d = 8$), $\square$ ($d = 16$). The Val. axis is logarithmic in $1 - \text{Val.}$; values below 0.85 are drawn as arrows ($\leftarrow$) on the frame. Values in Table 6.

3.4 Adaptation Step

CSFM adapts the velocity field v_θ while maintaining an exponential moving average (Polyak averaging; Polyak & Juditsky, 1992) to update the reference field $\bar{v}$. Each training iteration executes four steps:

1. **Generation and verification.** For conditions s_i and latents $z_i \sim q$, the reference flow generates candidate endpoints $x_i = \bar{\Phi}_{0\to 1}(z_i; s_i)$, which the oracle partitions into accepted samples $\mathcal{X}^+_{s_i}$ and rejected samples $\mathcal{X}^-_{s_i}$.
2. **Correspondence.** Each rejected point $x_i^- \in \mathcal{X}^-_{s_i}$ is mapped to a certified correspondence $x_i^\star \in \mathcal{X}^+_{s_i}$ via P_{s_i} equation 7, optionally refined by chord bisection.
3. **Characteristic steering.** A single reverse JVP traces the characteristic $x_{i,t} = \bar{\Phi}_{1\to t}(x_i^-; s_i)$ and evaluates the transported variation $r_{i,t} = A_t(x_i^\star - x_i^-)$, yielding the target velocity $u_i(t)$ equation 10.
4. **Joint regression.** Parameters θ are updated via stochastic gradient descent on $\mathcal{L}(\theta)$ equation 11.

4 Experiments

We evaluate our approach across three complementary suites, each isolating a different aspect of the proposed method, including: **Suite A (Geometric Generation):** Benchmarks our method across controlled synthetic settings spanning key difficulty axes: ambient dimension scaling, component fragmentation, and complex boundary geometry; **Suite B (Continuous Control):** Evaluates state-conditioned action representations in reinforcement learning under the action-mapping paradigm (Theile et al., 2025; Brahmanage et al., 2023; Kasaura et al., 2023), testing whether learned boundary supervision supports efficient downstream policy optimization; **Suite C (Trajectory Planning):** Evaluates full-horizon motion planning against HardFlow (Li et al., 2026) in an imitation-learning regime relying exclusively on offline demonstrations, testing whether adapted flows deliver safe, non-collapsing closed-loop control.

Evaluation criteria. To capture the trade-offs of constrained generation, we evaluate four complementary axes (formal definitions in Appendix E): (i) *validity* (oracle acceptance rate), (ii) *coverage and fidelity* (k-NN coverage (Naeem et al., 2020) and energy distance (Székely & Rizzo, 2013)

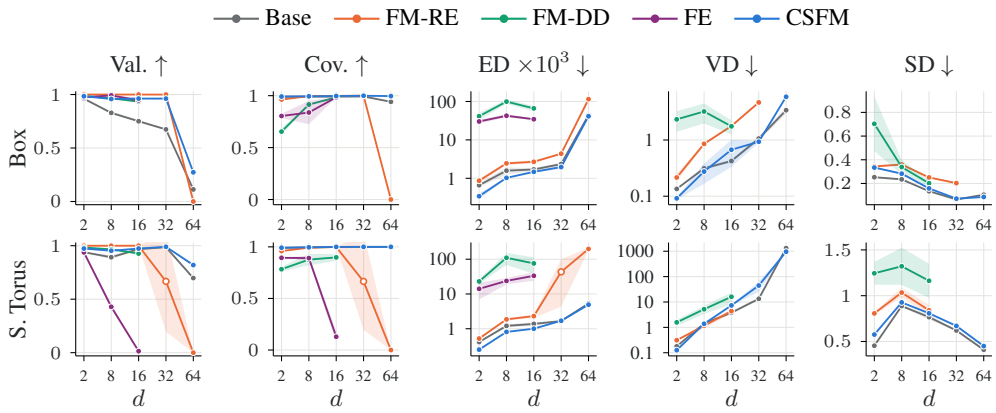


Figure 3: **Scaling with ambient dimension** ($m = 3$). Val., Cov., ED ($\times 10^3$, log scale), VD, and SD. Hollow markers indicate runs in which some seeds collapsed. Values in Table 6b.

to reference samples), (iii) *geometric regularity* (volume and shape distortion of the latent map), and (iv) *downstream utility* (RL return on Suite B; execution safety, success, and replan latency on Suite C).

4.1 Suite A: Geometric Sets

We benchmark CSFM against two alternative approaches for adapting the model distribution to the feasible set: randomized-exploration adaptation (FM-RE) and the privileged distance-to-set baseline (FM-DD, with analytic boundary access; Huan et al., 2026). We additionally compare against Flow Expander (FE; De Santi et al., 2026), which explicitly formulates adaptation as verifier-constrained entropy maximization, seeking to broaden the model distribution while retaining validity. FE therefore provides a particularly relevant comparison for whether constraint-aware adaptation preserves or expands coverage. Experiments sweep ambient dimension and component count; full tables and Gaussian-prior comparisons appear in Appendix F.1. FM-RE and FE use their native samplers: FM-RE's adapted field acts only on the ODE integrator, and FE requires a Gaussian source [4].

The cost of validity. While the base flow establishes a regular and well-distributed transport, the learned base flow leaves nonzero probability mass outside $\mathcal{C}_s$ in these experiments (Figure 2). Constrained adaptation reveals a clear trade-off: FM-RE often increases energy distance and geometric distortion while maximizing validity. In contrast, CSFM accepts a modest decrease in nominal validity while preserving broader coverage and superior distributional fidelity. Metric distortion varies across geometries; for instance, CSFM shows higher volume distortion on the $d = 16$ Solid Torus.

Scaling with dimension. As ambient dimension scales, CSFM maintains high coverage and substantially lower energy distance than FM-RE (Figure 3). This contrast is most evident on the Solid Torus: FM-RE exhibits severe inter-seed instability at $d = 32$ and completely collapses at $d = 64$, whereas CSFM keeps generating valid samples and maintains near-complete measured coverage. While CSFM scales more robustly overall, its volume distortion can exceed FM-RE's in lower dimensions where both remain valid. Increasing component count m preserves coverage across all methods; CSFM's validity tracks the base flow's slight decline while retaining the lowest energy distance, whereas FM-RE enforces strict validity throughout (Figure 6, Table 7).

Thin regions and irregular boundaries. Under the irregular boundary of the Julia set ($d = 2$), CSFM retains base coverage and validity while reducing energy distance and volume distortion; FM-RE gains higher validity at the expense of severe coverage loss (Figure 4). On the hyperplane ($d = 10$), CSFM outperforms the privileged FM-DD baseline in validity, coverage, and shape distortion.

[4] under which even an exact transport onto a uniform target has distortion, so FE is additionally compared under the same prior (Table 9)

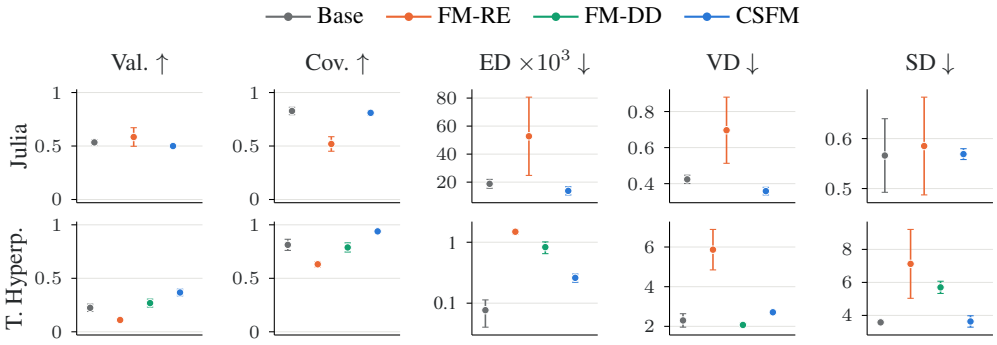


Figure 4: **Irregular and thin target regions.** Julia set ($d = 2$, top) and thickened hyperplane ($d = 10$, bottom). Points show individual seeds; bars show population SD. Values in Table 8.

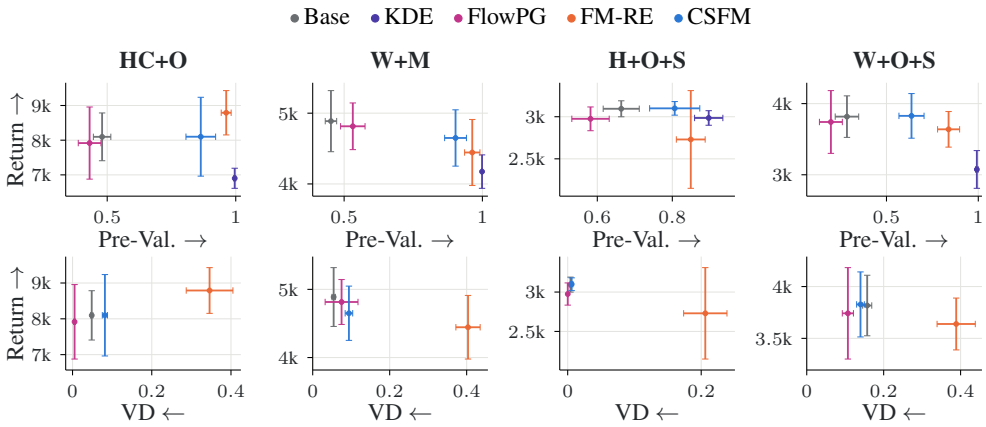


Figure 5: **Downstream control versus map quality.** Projected-policy return versus Pre-Val. (top) and VD (bottom). All policies are trained and evaluated with runtime projection; Pre-Val. is the share of their training actions accepted before repair (Table 11, projection off)(Table 10). VD was not evaluated for KDE; Returns in Table 12.

4.2 Suite B: Continuous Control

Suite B evaluates whether continuous-flow adaptation preserves downstream optimizer utility when deployed as latent action charts for reinforcement learning. Following the action-mapping paradigm (Theile et al., 2025; Kasaura et al., 2023), we benchmark four continuous-control tasks spanning isolated energetic limits (HC+O, HalfCheetah power, $d = 6$; W+M, Walker2d work, $d = 6$) and composite power-kinematic intersections (H+O+S, Hopper, $d = 3$; W+O+S, Walker2d, $d = 6$; Table 5). We freeze each generative map and train downstream TD3 policies (Fujimoto et al., 2018) on its latent coordinates, optionally applying runtime projection(Appendix D.1.3). Alongside the base flow and FM-RE, we benchmark two established standalone methods from scratch: *KDE* (Theile et al., 2024), a one-shot learner with explicit kernel regularity terms, and *FlowPG* (Brahmanage et al., 2023), parameterized by RealNVP and thus invertible by construction. Full generation diagnostics appear in Appendix F.3.

Raw validity does not dictate downstream control performance. Evaluating policies without runtime repair reveals a confound: unprojected models can achieve high nominal returns by frequently issuing inadmissible actions. The unadapted base flow, for instance, achieves competitive scores while violating constraints on over a third of its proposals (Table 12). Runtime projection brings delivered validity close to one for the flow-based maps, decoupling task success from constraint non-compliance and enabling a fair comparison of the underlying latent charts.

Task	Method	Pre-Val. (%) ↑	Safe (%) ↑	Success (%) ↑	Task metric	Latency (ms) ↓
Avoidance (steps ↓)	HardFlow	–	$99.2_{\pm 1.0}$	$99.2_{\pm 1.0}$	$50.6_{\pm 0.1}$	133
	Base	29.2	$73.6_{\pm 6.5}$	$28.0_{\pm 2.5}$	$78.3_{\pm 3.4}$	**27**
	Base + CEM	29.2	**100**	**100**	$\mathbf{50.1}_{\pm 0.1}$	81
	FM-RE	$\mathbf{76.5}_{\pm 0.0}$	$98.7_{\pm 1.9}$	$65.3_{\pm 46.2}$	$80.4_{\pm 14.7}$	30
	FM-RE + CEM	$\mathbf{76.5}_{\pm 0.0}$	**100**	$66.7_{\pm 47.1}$	$79.7_{\pm 14.4}$	80
	CSFM (ours)	$56.1_{\pm 0.7}$	$99.2_{\pm 1.0}$	$98.4_{\pm 0.8}$	$56.2_{\pm 1.7}$	31
	CSFM + CEM (ours)	$56.1_{\pm 0.7}$	**100**	**100**	$50.9_{\pm 0.6}$	81
Maze (score ↑)	HardFlow	–	**100**	$99.6_{\pm 0.8}$	$\mathbf{1.62}_{\pm 0.01}$	1,082
	Base	24.0	$28.8_{\pm 6.3}$	$97.6_{\pm 2.3}$	$1.56_{\pm 0.04}$	**242**
	Base + CEM	24.0	$99.2_{\pm 1.0}$	**100**	$1.59_{\pm 0.00}$	511
	FM-RE	$\mathbf{67.2}_{\pm 5.9}$	$57.3_{\pm 39.4}$	$32.0_{\pm 36.9}$	$0.46_{\pm 0.58}$	247
	FM-RE + CEM	$\mathbf{67.2}_{\pm 5.9}$	$65.3_{\pm 34.7}$	$37.3_{\pm 42.0}$	$0.55_{\pm 0.65}$	499
	CSFM (ours)	$46.7_{\pm 1.0}$	$98.8_{\pm 1.6}$	**100**	$1.54_{\pm 0.02}$	271
	CSFM + CEM (ours)	$46.7_{\pm 1.0}$	$96.8_{\pm 2.0}$	$98.0_{\pm 3.1}$	$1.54_{\pm 0.04}$	514

Table 1: **Trajectory planning against HardFlow.** Closed-loop performance under single-round and CEM deployment. **Pre-Val.** is validity before search and repair; **Latency** is wall-clock time per replan on one RTX 5090. Mean ± population SD across seeds.

CSFM preserves policy learning under complex constraints. On the composite boundaries of H+O+S and W+O+S, CSFM attains the highest mean return, approximately matching the unadapted base flow within seed variability and exceeding FM-RE. Crucially, while FM-RE attains higher raw validity on these tasks, it is associated with lower downstream return than both the base flow and CSFM. CSFM balances boundary steering with geometric regularity, providing a stable representation for the RL agent.

Adaptation on simpler boundaries. On simpler energetic limits (HC+O and W+M), downstream benefits are mixed. On HC+O, FM-RE has the higher mean return, whereas CSFM matches the base flow within seed variability. On W+M, the unadapted base flow has the highest mean return despite the lowest raw validity, followed by FlowPG. Although both adapted flows see lower returns than the base model on W+M, CSFM retains more of the baseline mean return than FM-RE, consistent with better-preserved map regularity.

4.3 Suite C: Trajectory Planning

Suite C assesses trajectory-level generation on two benchmark tasks from HardFlow (Li et al., 2026): receding horizon obstacle avoidance ($x \in \mathbb{R}^{16\times 6}$, replanning every eight steps) and full horizon maze navigation ($x \in \mathbb{R}^{384\times 6}$, tracked via PD control). We deliberately target the imitation learning regime in which the initial certified dataset consists exclusively of expert demonstrations rather than uniform feasible draws, comparing black-box methods against HardFlow, which solves a differentiable program at each replan. All flow-based models initialize from the released HardFlow checkpoint and are deployed under the same inference pipeline: latent cross-entropy method (CEM) (Rubinstein & Kroese, 2004) search guided by the task value function, boolean verification of candidate proposals, and chord bisection repair for rejected trajectories (Appendix F.4).

One-shot Planning. Without iterative search, CSFM reaches near-perfect safety and success ($> 98\%$), matching HardFlow's performance at far lower latency (Table 1). Conversely, FM-RE's severe task failures despite high nominal validity show that unguided adaptation easily collapses onto trivial certified paths.

Effects of boundary projection. When a candidate trajectory fails verification, chord bisection toward a certified anchor returns the certified side of a shrinking bracket near a boundary crossing rather than a point in the interior of $\mathcal{C}_s$. This placement may reduce the robustness margin, and the residual safety violations of certified maze rollouts during tracking reflect the mismatch between verification on the predicted rollout and execution. In addition, projection shifts trajectories away from the demonstration manifold toward local optima of the value model, which is consistent with the small metric difference relative to the unadapted base model (Appendix F.4).

5 Related Work

Constraint Satisfaction and Validity in Generative Flows. Constrained flow models differ primarily in the information they assume about the feasible set. When explicit geometric or differentiable constraint structure is available, flows and diffusions can be built on known manifolds (Gemici et al., 2016; De Bortoli et al., 2022), Reflected diffusion and Flow Matching enforce known domain boundaries through reflected dynamics (Lou & Ermon, 2023; Xie et al., 2024), while Mirror Flow Matching constructs flows on known convex domains through regularized mirror maps (Guan et al., 2026). Other methods incorporate distance-to-set penalties during training (FM-DD; Huan et al., 2026) or solve differentiable constrained optimization problems at inference time (HardFlow; Li et al., 2026). ECI sampling instead adapts pretrained flow-matching models at inference time without gradient-based fine-tuning by alternating extrapolation, constraint correction, and interpolation steps (Cheng et al., 2025). These approaches exploit geometric, differentiable, or correction structure beyond pointwise membership. Under verifier-only access, FM-RE (Huan et al., 2026) uses randomized exploration of the constraint set, while Flow Expander (De Santi et al., 2026) optimizes verifier-constrained entropy over stochastic processes. CSFM addresses this black-box regime by converting certified correspondences into deterministic displacement supervision and transporting those corrections along reference-flow characteristics.

Flow Sensitivities, Reward Adaptation, and Steering. Characteristic steering is related to sensitivity-based control and adaptation of continuous generative models. Classical neural-ODE adjoint methods propagate derivatives of differentiable terminal objectives through continuous dynamics (Chen et al., 2018; Li et al., 2020). More recently, Adjoint Matching formulates reward fine-tuning of flow and diffusion models as stochastic optimal control and casts the resulting control problem as regression (Domingo-Enrich et al., 2025). CSFM operates under a different supervision model: the membership oracle may supply a binary reward but no differentiable terminal objective. Instead, pairing rejected endpoints with certified feasible anchors produces explicit terminal displacement vectors, which are transported through the differential of the reference flow to construct velocity-regression targets without differentiating the oracle.

Action and Trajectory Mapping. In safe reinforcement learning and long-horizon planning, generative decoders can parameterize policies over constrained action or trajectory spaces under the *action-mapping paradigm* (Theile et al., 2025; 2024; Kasaura et al., 2023; Brahmanage et al., 2023; Janner et al., 2022) or continuous action masking (Stolz et al., 2024). Aggressively maximizing validity can introduce local distortion, poor conditioning, or reduced coverage, potentially impairing optimization in the latent coordinates. CSFM combines base Flow Matching with characteristic steering to improve validity while retaining coverage and downstream usefulness.

6 Conclusion

We introduced CSFM for adapting continuous flows to oracle-defined sets through characteristic steering and certified correspondences. The correspondence construction also supports runtime repair, with an optional latent-distance cost for selecting corrections. Across geometric generation, control, and planning, the results show that improving oracle acceptance alone is insufficient: coverage and downstream usefulness also matter. Base flows combined with search and projection are already strong, motivating adaptation that improves acceptance while retaining their useful structure. CSFM offers this balance across the tested settings, including increasing ambient dimension.

Future work should isolate the benefits of latent-aware projection and test transfer to richer oracle-defined domains, including molecular and materials generation. Adaptive exploration and growing, diverse anchor pools could reduce dependence on the initial certified dataset. A further direction is to learn generators over abstract geometric domains, extending action mapping to spaces of candidate decisions defined by membership oracles. Combining such generators with geometric observation representations (Zapata Rodriguez et al., 2026) could let policies reason about and explore these spaces directly, opening applications beyond physical action coordinates.

AI USE STATEMENT

In this work, we used generative AI tools (Claude Code, OpenAI Codex, and ChatGPT) to help implement methods, limited to scaling and efficiency work on our code, porting baselines to our framework, generating experiment, evaluation, and visualization scripts, and running experiments across machines; to provide feedback on experiment configurations and protocols, with all design decisions made by the authors. We have not used generative AI tools to develop the theoretical framework, formulate or prove mathematical claims, propose hypotheses, or design the methodology or experiments; the remaining required-disclosure tasks are not applicable to this work. Additionally, we used generative AI tools to brainstorm plot layouts, create all figures and tables, edit the text for readability (including condensing equations), and format and verify references. We have reviewed all AI-assisted work: Ported baselines were checked against their released implementations, and table values were recomputed from the raw evaluation records. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The assumptions behind characteristic steering are stated in Appendix A, with the first-variation argument in Appendix A.2 and the optional profile results in Appendix B. The benchmark environments, their oracles, and the sampling procedures for reference data are described in Appendix C; the three-stage training, checkpoint-selection, and evaluation protocol in Appendix D; the architecture, optimizer settings, and budgets of every arm in Table 2; the metric definitions and reporting conventions in Appendix E; and the Suite C deployment routine and query accounting in Appendix F.4. Baselines were ported into our framework and checked against their released implementations, and Suite C builds on HardFlow's released checkpoints and benchmark code. The training, evaluation, and figure-generation code will be released upon publication.

REFERENCES

Michael S. Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2209.15571.

Janaka Chathuranga Brahmanage, Jiajing Ling, and Akshat Kumar. FlowPG: Action-constrained policy gradient with normalizing flows. In *Advances in Neural Information Processing Systems*, volume 36, pp. 20118–20132, 2023. doi: 10.52202/075280-0882. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/3fd9fe8ec6d7238bf71784797399bb61-Abstract-Conference.html.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://arxiv.org/abs/1806.07366.

Chaoran Cheng, Boran Han, Danielle C. Maddix, Abdul Fatir Ansari, Andrew Stuart, Michael W. Mahoney, and Bernie Wang. Gradient-free generation for hard-constrained systems. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=teE4pl9ftK.

Rob Cornish, Anthony Caterini, George Deligiannidis, and Arnaud Doucet. Relaxing bijectivity constraints with continuously indexed normalising flows. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 2133–2143, 2020. URL https://proceedings.mlr.press/v119/cornish20a.html.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, volume 26, pp. 2292–2300, 2013. URL https://proceedings.neurips.cc/paper/2013/hash/af21d0c97db2e27e13572cbf59eb343d-Abstract.html.

Bernard Dacorogna and Jürgen Moser. On a partial differential equation involving the Jacobian determinant. *Annales de l'Institut Henri Poincaré C, Analyse non linéaire*, 7(1):1–26, 1990. doi: 10.

1016/S0294-1449(16)30307-9. URL https://doi.org/10.1016/S0294-1449(16)30307-9.

Valentin De Bortoli, Emile Mathieu, Michael Hutchinson, James Thornton, Yee Whye Teh, and Arnaud Doucet. Riemannian score-based generative modelling. In *Advances in Neural Information Processing Systems*, volume 35, pp. 2407–2422, 2022. URL https://arxiv.org/abs/2202.02763.

Riccardo De Santi, Kimon Protopapas, Ya-Ping Hsieh, and Andreas Krause. Verifier-constrained flow expansion for discovery beyond the data. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2602.15984.

Carles Domingo-Enrich, Michal Drozdzal, Brian Karrer, and Ricky T. Q. Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2409.08861.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4RL: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020. URL https://arxiv.org/abs/2004.07219.

Scott Fujimoto, Herke van Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1587–1596, 2018. URL https://proceedings.mlr.press/v80/fujimoto18a.html.

Mevlana C. Gemici, Danilo Jimenez Rezende, and Shakir Mohamed. Normalizing flows on Riemannian manifolds. *arXiv preprint arXiv:1611.02304*, 2016. URL https://arxiv.org/abs/1611.02304.

Yunrui Guan, Krishnakumar Balasubramanian, and Shiqian Ma. Mirror flow matching with heavy-tailed priors for generative modeling on convex domains. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=dZKl7uc0XQ.

Sven Gutjahr, Riccardo De Santi, Luca Schaufelberger, Kjell Jorner, and Andreas Krause. Constrained flow optimization via sequential fine tuning for molecular design. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://arxiv.org/abs/2605.30610.

Philip Hartman. *Ordinary Differential Equations*. Number 38 in Classics in Applied Mathematics. SIAM, 2nd edition, 2002. doi: 10.1137/1.9780898719222. URL https://doi.org/10.1137/1.9780898719222.

Zhengyan Huan, Jacob Boerma, Li-Ping Liu, and Shuchin Aeron. Constraint-aware flow matching via randomized exploration. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=OR4h9WPJhV.

Michael Janner, Yilun Du, Joshua B. Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 9902–9915, 2022. URL https://arxiv.org/abs/2205.09991.

Kazumi Kasaura, Shuwa Miura, Tadashi Kozuno, Ryo Yonetani, Kenta Hoshino, and Yohei Hosoe. Benchmarking actor-critic deep reinforcement learning algorithms for robotics control with action constraints. *IEEE Robotics and Automation Letters*, 8(8):4449–4456, 2023. doi: 10.1109/LRA.2023.3284378. URL https://arxiv.org/abs/2304.08743.

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://arxiv.org/abs/1904.06991.

Xuechen Li, Ting-Kam Leonard Wong, Ricky T. Q. Chen, and David Duvenaud. Scalable gradients for stochastic differential equations. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 3870–3882, 2020. URL https://proceedings.mlr.press/v108/li20i.html.

Zeyang Li, Kaveh Alim, and Navid Azizan. HardFlow: Hard-constrained sampling for flow-matching models via trajectory optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(10):13619–13635, 2026. doi: 10.1109/TPAMI.2026.3708824. URL https://arxiv.org/abs/2511.08425.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.02747.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2209.03003.

Aaron Lou and Stefano Ermon. Reflected diffusion models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 22675–22701, 2023. URL https://arxiv.org/abs/2304.04740.

Muhammad Ferjad Naeem, Seong Joon Oh, Youngjung Uh, Yunjey Choi, and Jaejun Yoo. Reliable fidelity and diversity metrics for generative models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7176–7185, 2020. URL https://proceedings.mlr.press/v119/naeem20a.html.

Boris T. Polyak and Anatoli B. Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992. doi: 10.1137/0330046.

Reuven Y. Rubinstein and Dirk P. Kroese. *The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning*. Springer, New York, 2004. doi: 10.1007/978-1-4757-4321-0.

Roland Stolz, Hanna Krasowski, Jakob Thumm, Michael Eichelbeck, Philipp Gassert, and Matthias Althoff. Excluding the irrelevant: Focusing reinforcement learning through continuous action masking. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-3013. URL https://arxiv.org/abs/2406.03704.

Gábor J. Székely and Maria L. Rizzo. Energy statistics: A class of statistics based on distances. *Journal of Statistical Planning and Inference*, 143(8):1249–1272, 2013. doi: 10.1016/j.jspi.2013.03.018.

Mirco Theile, Daniele Bernardini, Raphael Trumpp, Cristina Piazza, Marco Caccamo, and Alberto L. Sangiovanni-Vincentelli. Learning to generate all feasible actions. *IEEE Access*, 12:40668–40681, 2024. doi: 10.1109/ACCESS.2024.3376739. URL https://doi.org/10.1109/ACCESS.2024.3376739.

Mirco Theile, Lukas Dirnberger, Raphael Trumpp, Marco Caccamo, and Alberto L. Sangiovanni-Vincentelli. Action mapping for reinforcement learning in continuous environments with constraints. *Reinforcement Learning Journal*, 6:343–363, 2025. URL https://rlj.cs.umass.edu/2025/papers/Paper27.html.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, 2012. doi: 10.1109/IROS.2012.6386109.

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. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=CD9Snc73AW.

Jonathan Weed and Francis Bach. Sharp asymptotic and finite-sample rates of convergence of empirical measures in Wasserstein distance. *Bernoulli*, 25(4A):2620–2648, 2019. doi: 10.3150/18-BEJ1065.

Tianyu Xie, Yu Zhu, Longlin Yu, Tong Yang, Ziheng Cheng, Shiyue Zhang, Xiangyu Zhang, and Cheng Zhang. Reflected flow matching. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 54614–54634. PMLR, 2024. URL https://proceedings.mlr.press/v235/xie24k.html.

Andres R. Zapata Rodriguez, Hongpeng Cao, Raphael Trumpp, and Marco Caccamo. A hierarchical geometric observation interface for spatial planning in reinforcement learning. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=U10DFNcMrW.

Claudio Zeni, Robert Pinsler, Daniel Zügner, Andrew Fowler, Matthew Horton, Xiang Fu, Zilong Wang, Aliaksandra Shysheya, Jonathan Crabbé, Shoko Ueda, Roberto Sordillo, Lixin Sun, Jake Smith, Bichlien Nguyen, Hannes Schulz, Sarah Lewis, Chin-Wei Huang, Ziheng Lu, Yichi Zhou, Han Yang, Hongxia Hao, Jielan Li, Chunlei Yang, Wenjie Li, Ryota Tomioka, and Tian Xie. A generative model for inorganic materials design. *Nature*, 639(8055):624–632, 2025. doi: 10.1038/s41586-025-08628-5. URL https://doi.org/10.1038/s41586-025-08628-5.

A Characteristic Steering: Components and Derivations

A.1 Certified anchors and correspondence selection

Assumption 1 (Certified-anchor availability) *For every condition s, the oracle-defined feasible set*

$$\mathcal{C}_s = \{x \in \mathcal{A} : \mathcal{O}_s(x) = 1\}$$

is nonempty, consistent with Equation equation 1. During training, we assume access to a nonempty set of certified anchors

$$\varnothing \neq \mathcal{X}_s^+ \subset \mathcal{C}_s,$$

used to construct corrective correspondences; the certified reference samples among them also serve as targets of the Flow Matching term. At inference, we require only that at least one certified anchor can be recovered whenever correction is invoked. Such anchors need not be given a priori and may be obtained from memory, feasible primitives, accepted model proposals, or other oracle-certified proposals; in our experiments they are the certified reference bank and the accepted decodes of the model.

Fix a condition s and an anchor set $\mathcal{X}_s^+$ as in Assumption 1. Acceptance certifies membership, not strict interiority. A pool for another condition need not remain accepted at s; it must be re-certified. Obtaining an initial anchor can itself be expensive for sets with low acceptance probability. Likewise, accepted model samples and chord queries need not be uniform on $\mathcal{C}_s$ and cannot automatically replace the uniform target bank in the Flow-Matching term.

Optional chord refinement and runtime repair. For a rejected x and accepted anchor a, let $\ell(\tau) = x + \tau(a - x)$, $\tau \in [0, 1]$. Assume the oracle is defined on the chord; for an oracle restricted to $\mathcal{A}$, one can extend it by rejecting points outside $\mathcal{A}$. Start with a rejected–accepted bracket $[\tau_-, \tau_+]$, initially $[0, 1]$. Query its midpoint and replace the endpoint with the same label. After b queries, return $x^\star = \ell(\tau_+)$, which is still certified, with bracket length

$$\|\ell(\tau_+) - \ell(\tau_-)\|_2 = 2^{-b}(\tau_+^{(0)} - \tau_-^{(0)})\|a - x\|_2. \tag{13}$$

Thus a bracket of initial length L is reduced to tolerance $\varepsilon > 0$ using $\max\{0, \lceil \log_2(L/\varepsilon) \rceil\}$ midpoint queries, excluding anchor acquisition and endpoint certification. Every bracket contains a topological boundary point; no convexity or monotone membership along the chord is required. Multiple crossings can occur, so this procedure need not identify the first crossing or the nearest point in the set. A coarse sweep over several chords can first select a shorter bracket, but may miss narrower accepted intervals. The guarantee is for the returned accepted point, not for the full chord or for strict interiority.

Set $P_s(x) = x$ for accepted inputs. For rejected inputs, returning either an accepted anchor or the accepted end of a refined bracket guarantees membership under a deterministic oracle for the same condition. If no anchor is available, this guarantee does not apply. Even when the underlying flow is smooth and invertible, the repaired map $P_s \circ T_\theta$ need not be continuous or invertible.

A.2 Reference-flow first variation for the constant allocation

Fix s, suppress it below, and let $x_t = \bar{\Phi}_{1\to t}(x)$ be a frozen reference characteristic, with $x_0 = z$ and $x_1 = x$. Define

$$A_t = \bar{J}_{1\to t}(x), \qquad r = x^\star - x, \qquad r_t = A_t r. \tag{14}$$

The displacement r is unnormalized. Assume that $\bar{v}$ is continuous in time and continuously differentiable in space on a neighborhood supporting these flows and nearby perturbed solutions (Hartman, 2002, Ch. V). Then A_t is continuous and invertible, $A_1 = I$, and the chain rule gives

$$A_t = \bar{J}_{0\to t}(z)\bar{J}_{0\to 1}(z)^{-1}. \tag{15}$$

For a continuous prescribed control c_t, consider

$$\dot{y}_t^h = \bar{v}(y_t^h, t) + hc_t, \qquad y_0^h = z, \qquad \delta x_t = \frac{d}{dh} y_t^h \bigg|_{h=0}. \tag{16}$$

The variational equation is $\dot{\delta x}_t = D_x \bar{v}(x_t, t)\delta x_t + c_t$ with $\delta x_0 = 0$. Its transition matrix from τ to t is $A_t A_\tau^{-1}$, so

$$\delta x_t = \int_0^t A_t A_\tau^{-1} c_\tau \, d\tau. \tag{17}$$

For the constant allocation, take $c_t = A_t r$. Cancellation of the transition matrices gives

$$\delta x_t = t A_t r, \qquad y_t^\alpha = x_t + \alpha t A_t r + o(\alpha), \tag{18}$$

and, at the endpoint,

$$y_1^\alpha = x + \alpha r + o(\alpha), \qquad \alpha \to 0. \tag{19}$$

The corresponding regression target on the reference path is exactly $u(t) = \bar{v}(x_t, t) + \alpha A_t r$, as in Equation equation 10. In particular, α scales the full selected displacement, without a trajectory-dependent rescaling. Constant allocation refers to equal contributions to the first-order *terminal* response per unit time, not equal correction norms $\|\alpha A_t r\|_2$ along the path.

Scope of the statement. This is a first variation of an ideal perturbed flow as $\alpha \to 0$, not an exact finite-step endpoint identity. Under stronger smoothness bounds the remainder can be bounded quadratically, but continuous spatial differentiability alone gives the stated $o(\alpha)$. A smooth field perturbation that agrees with c_t on the reference path has the same first variation. A jointly trained neural field need not match this prescription exactly, because of regression error, finite sampling, shared parameters, and the competing Flow-Matching loss. Even the linearized point $x + \alpha r$ can remain outside the set when $0 < \alpha < 1$, and a chord through a nonconvex set need not stay inside the set. Neither the first variation nor the loss proves convergence to the uniform target or a bound on chart distortion.

Numerical realization. At sampled times t_m, one JVP through the frozen numerical reverse trajectory returns the approximate states and transported displacements. The targets are simply $u_m = \bar{v}(x_{t_m}, t_m) + \alpha r_{t_m}$, with no node-wise normalization or stiffness multiplier. Training differentiates only the active network prediction. The JVP differentiates the discrete reverse solver; it is not automatically the differential of the exact inverse of the discrete forward solver. Thus numerical integration error is separate from the continuous-flow identity above. Fixed solver-node regression is also a discrete approximation to the time expectation in Equation equation 12.

Component	Suites A and B	Suite C
Velocity network	set transformer: $d_{\text{model}} = 256$, 8 heads, 1 cross-attention layer (8 heads, expansion 4, pre-LN), PMA pooling (256, 8 heads); head width 256, depth 2, GELU	HardFlow's released 1-D convolutional U-Net
Source prior	$\mathcal{U}(-1,1)^d$ ($\mathcal{N}(0,I)$ for Table 9)	HardFlow's
ODE integration	Euler, 32 steps	Euler, 10 steps
Base training	Adam, lr 3×10^{-4}, gradient clip 1.0; 32 states $\times$ 256 targets per update; minibatch Sinkhorn coupling, $\varepsilon = 0.02$ annealed from 0.5 over the first 10,000 updates, 50 iterations; 20,000 / 40,000 / 80,000 updates for $d \leq 8$ / 16 / ≥ 32 (Julia and slab: 20,000; Suite B: 300,000); evaluated every 1,000 updates (10,000 in Suite B)	released checkpoint, no base training
Adaptation (CSFM, FM-RE, FM-DD)	5,000 updates, Adam, lr 3×10^{-4}, clip 1.0, same batch as pretraining; evaluated every 500 updates	5,000 updates (`avoiding-v0`), 2,000 (`maze2d-large-v1`); 32 states / 8 states per update
CSFM	$\alpha = 0.1$, $\lambda = 1$, $K = 8$ knots; $F = 256$ reference decodes graded per state, 32 rejected decodes corrected per state, Polyak-averaged reference field ($\tau = 0.01$); flow-matching term on the state's 256 reference samples	same, with 256 / 64 decodes, 32 / 8 corrections and 64 / 16 reference rows per state; chord-refined correspondence (16 nearest anchors, 4 grid points, 8 bisections; Appendix C.3)
FM-RE	2,560 rays per state, $\lambda = 20$, $t_0 = 0.6$, $\sigma_{\text{init}} = 0.3$, $\sigma_{\min} = 10^{-5}$, σ lr 10^{-4}	2,560 / 640 rays per state, otherwise the same
FM-DD	$\lambda = 20$, product coupling	–
Flow Expander	10 rounds $\times$ (4 expansion + 4 projection) updates, lr 10^{-4}, $\eta = 2$, $\tilde{\gamma} = 1.5$ decaying as $1/(1 + 3(k-1))$, $\lambda = 1.2$ cut at $t > 0.95$, 40 SDE steps, 8 draws per state	–
KDE	the flows' set network (backbone and head as above), applied once as $z \mapsto a$ (no ODE); 1,024 supports, $\sigma = 0.1$, $\sigma' = 2\sigma$, regularization 10^{-3}, lr 10^{-4}, 220,000 updates, best checkpoint per seed by Val.$\times$Cov.	–
FlowPG	RealNVP, 6 couplings, hidden 256, box-violation Gaussian prior $\sigma = 10^{-4}$, Adam 10^{-4}, clip 0.1, 32 rows per update, 500,000 updates	–
Evaluation	512 held-out states $\times$ 2,048 latents against 1,024 uniform references (5 repeats); Jacobians on 256 states $\times$ 128 latents; checkpoint selection by validity $\times$ coverage on the first 64 evaluation states $\times$ 256 draws (128 references)	50 episodes per seed, Appendix C.3

Table 2: **Architectures, training schedules and optimizer settings of every arm**, read from the saved run configurations.

A.3 Method Hyperparameters and Implementation Details

This section details the hyperparameter specifications, architectural dimensions and optimization parameters of every arm across Suites A, B and C; per-suite training and evaluation protocols follow in Appendix D.

Characteristic steering and correspondence parameters. In Suites A and B, unless varied in an ablation, CSFM uses the steering formulation with constant temporal allocation $c_t = A_t r$. The steering coefficient is fixed at the dimensionless value $\alpha = 0.1$, and the joint loss weight is set to $\lambda = 1.0$, balancing the base Sinkhorn OT-Flow Matching objective with the reverse-transported steering loss $\mathcal{L}_{\text{CSFM}}$:

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{FM}}(\theta) + \lambda\, \mathcal{L}_{\text{CSFM}}(\theta). \tag{20}$$

In Suites A and B, for rejected decodes $x^- = \Phi_{0\to1}(z^-; s) \notin \mathcal{C}_s$, certified anchors are selected as the nearest certified point ($x^\star = \arg\min_{a \in \mathcal{X}_s^+} \|a - x^-\|_2$) of a pool holding the state's 256 reference samples and the $F = 256$ reference decodes graded in that step. When optional chord

refinement is enabled, 16 coarse probes and $b = 8$ bisections are applied along the segment connecting x^- to $x^\star$. The reverse-transported variation $r_t = \bar{J}_{1\to t}(x^-)[x^\star - x^-]$ is evaluated via a single Jacobian-vector product (JVP) through the frozen numerical reverse ODE integrator over $N = 32$ discretization steps across $t \in [0, 1]$ ($N = 10$ in Suite C), evaluated at $K = 8$ knots, the integrator nodes $\{0, 5, 9, 14, 18, 23, 27, 32\}/32$ ($\{0, 1, 3, 4, 6, 7, 9, 10\}/10$ in Suite C), without node-wise normalization, stiffness clipping, or trust-region scaling.

Architectures and optimization parameters. Table 2 summarizes the neural network architectures, training schedules, optimizer settings, and batch configurations across all three benchmark suites.

B Optional Temporal Profiles as Minimization Problems

This section gives optional generalizations of the constant allocation used in the main text. Fix one reference characteristic, its nonzero displacement $r = x^\star - x$, and the same scalar α. Write $a(t) = \|A_t r\|_2$. Invertibility and continuity of A_t on $[0,1]$ imply $0 < \min_t a(t) \le \max_t a(t) < \infty$. For $r = 0$, the correction vanishes and no profile optimization is needed.

Consider nonnegative time profiles with unit integral and prescribe

$$\mathcal{W} = \left\{ w : [0,1] \to [0,\infty) \text{ measurable} : \int_0^1 w(t)\,dt = 1 \right\}, \qquad c_t^w = w(t) A_t r. \tag{21}$$

For continuous profiles (including all the optimizers below), Equation equation 17 gives

$$\delta x_t = A_t r \int_0^t w(\tau)\,d\tau, \qquad \delta x_1 = r. \tag{22}$$

The unit-integral constraint therefore fixes the same first-order terminal displacement αr. Choosing a cost for the time-local velocity correction defines an optimization problem for w; different costs produce different profiles. These are optima within the restricted family $w(t)A_t r$, not over arbitrary vector-valued controls or neural-network updates.

Minimum integrated squared correction. One can minimize the total squared velocity correction,

$$\min_{w \in \mathcal{W}} E(w), \qquad E(w) = \alpha^2 \int_0^1 a(t)^2 w(t)^2\,dt. \tag{23}$$

Let $Z_2 = \int_0^1 a(t)^{-2}\,dt$. Cauchy–Schwarz gives

$$1 = \left(\int_0^1 [a(t)w(t)] a(t)^{-1}\,dt \right)^2 \le \left(\int_0^1 a(t)^2 w(t)^2\,dt \right) Z_2. \tag{24}$$

Equality holds precisely when $a(t)w(t)$ is proportional to $a(t)^{-1}$ almost everywhere. Hence

$$w_E^\star(t) = \frac{a(t)^{-2}}{Z_2}, \qquad \min_{w \in \mathcal{W}} E(w) = \frac{\alpha^2}{Z_2}. \tag{25}$$

Minimum peak correction. Alternatively, minimize the largest correction norm,

$$\min_{w \in \mathcal{W}} P(w), \qquad P(w) = \alpha \operatorname*{ess\,sup}_{t \in [0,1]} a(t)w(t). \tag{26}$$

Let $Z_1 = \int_0^1 a(t)^{-1}\,dt$ and $K = \operatorname{ess\,sup}_t a(t)w(t)$. Since $w(t) \le K/a(t)$ almost everywhere, normalization implies $1 \le KZ_1$. The lower bound is attained by

$$w_P^\star(t) = \frac{a(t)^{-1}}{Z_1}, \qquad \min_{w \in \mathcal{W}} P(w) = \frac{\alpha}{Z_1}. \tag{27}$$

This profile equalizes the velocity-correction norm over time. It differs from the energy minimizer, which allocates more terminal response to times with smaller $a(t)$.

Relation to the constant allocation. The main-text choice $w(t) = 1$ satisfies the terminal-response constraint and requires no trajectory normalization. It minimizes $\int_0^1 w(t)^2\,dt$ under that constraint, since $\int_0^1 w(t)^2\,dt \ge (\int_0^1 w(t)\,dt)^2 = 1$. This is a cost on the allocation itself; it is not generally the physical correction energy or peak norm. Constant allocation solves Equations equation 23 and equation 26 only when $a(t)$ is constant almost everywhere. Thus the optional optimal profiles do not imply an energy- or peak-minimality claim for the simplified method.

Discrete counterparts. For nodes t_m and positive quadrature weights β_m summing to one, impose $\sum_m \beta_m w_m = 1$. Replacing the integrals by these weighted sums gives

$$w^\star_{E,m} = \frac{a(t_m)^{-2}}{\sum_j \beta_j a(t_j)^{-2}}, \qquad w^\star_{P,m} = \frac{a(t_m)^{-1}}{\sum_j \beta_j a(t_j)^{-1}}. \tag{28}$$

These minimize the sampled squared energy and sampled peak norm, respectively, at fixed α. Uniform weights $\beta_m = 1/M$ give arithmetic-mean normalization; a nonuniform time grid generally calls for time quadrature weights. Neither sampled normalization establishes exact continuous-time normalization. Multiplying α by either normalization factor would change the prescribed terminal displacement and is not part of the simplified method.

C Experimental Environments

C.1 Geometric Benchmark

The geometric benchmark varies primitive geometry, disjunctive complexity, and ambient dimension. Increasing d increases the ambient dimension while preserving the defining rule of each primitive: boxes acquire additional coordinate-wise bounds, balls acquire additional radial coordinates, and solid tori acquire additional orthogonal cross-sectional coordinates. Disjunctive complexity is controlled by assembling m non-overlapping components translated by centers $\{c_j\}_{j=1}^m \subset \mathbb{R}^d$. The conditioning state is $s = (\{c_j\}_{j=1}^m, \Theta_g)$, where Θ_g contains primitive-specific scale parameters. The binary membership oracle is

$$\mathcal{O}_s(x) = \mathbb{I}\left[\min_{j\in\{1,\dots,m\}} G_{g,d}(x - c_j) \le 0\right]. \tag{29}$$

Every component lies inside $\mathcal{A} = [-1, 1]^d$ by construction, so membership implies $x \in \mathcal{A}$; the slab and Julia oracles test $x \in \mathcal{A}$ explicitly. For these analytic primitives, training and reference samples are drawn from $\mathrm{Unif}(\mathcal{C}_s)$ by selecting components in proportion to their volume and sampling uniformly within the selected component.

Primitive	**Constraint Function** $G_{g,d}(x)$	**Parameters** Θ_g	**Geometry**
Box	$\max_{i=1,\dots,d}(\lvert x_i\rvert - a_i)$	$a \in \mathbb{R}^d_{>0}$	Hyperrectangle / orthotope
Ball	$\lVert x\rVert_2 - r$	$r > 0$	Solid d-ball
Solid Torus	$(\sqrt{x_1^2 + x_2^2} - R)^2 + \sum_{i=3}^d x_i^2 - r^2$	$R > r > 0$	Solid hypertorus

Table 3: **Synthetic geometric primitives.** Points satisfying $G_{g,d}(x) \le 0$ define the unshifted component.

Boxes are sampled coordinatewise, while balls use the standard radial transform $\rho = ru^{1/d}$ with a uniform direction. Solid tori are sampled analogously in tube coordinates, with rejection correction for the toroidal volume factor $R + u$, yielding exact volume-uniform draws.

Cell parameters. Every cell uses $\mathcal{A} = [-1, 1]^d$ and a fixed m; all scale parameters are drawn i.i.d. per state and component. Box half-widths $a_i \sim \mathcal{U}[0.28, 0.34]$ with centres in $[-0.6, 0.6]^d$; ball radii $r \sim \mathcal{U}[0.1, 0.2]$ with centres in $[-0.7, 0.7]^d$; torus core radii $R \sim \mathcal{U}[0.26, 0.34]$ with tube radius $r = 0.1$ and centres in $[-0.5, 0.5]^d$. Centres are relaxed for 40 steps until every pair is at least $\mathrm{ext}_i + \mathrm{ext}_j + 0.03$ apart, where ext is the box half-diagonal, the ball radius or $R + r$. The conditioning state lists each component's descriptor: $(\mathrm{lo}, \mathrm{hi})$ for a box, (c, r) for a ball and (c, R, r) for a torus.

Julia Set. The Julia benchmark evaluates generalization across irregular oracle-defined boundaries generated by the quadratic dynamics

$$z_{k+1} = z_k^2 + c. \tag{30}$$

A scene is specified by the complex parameter c, and actions are mapped to the plane by the fixed scaling $\psi(a) = 1.5\,a$, so a state is a scene. Given an action $a \in [-1, 1]^2$, we set $z_0 = \psi(a)$ and define the membership oracle

$$\mathcal{O}_s(a) = \mathbb{I}\left[\max_{0\le k\le 60} |z_k| \le 2\right], \qquad z_{k+1} = z_k^2 + c_s. \tag{31}$$

We use eight Julia scenes, split into four training, two validation, and two held-out test parameters: $c = -1$, $0.2823 + 0.5301i$, $0.3795 + 0.3349i$, -1.3107 (training); $-0.1226 + 0.7449i$, $-0.1134 + 0.8606i$ (validation); $-0.5043 + 0.5628i$, $-1.1380 + 0.2403i$ (test). Each scene is one state, so the Julia test set has two states. The selected scenes were screened for slow-escape artifacts: across all scenes, the acceptance fraction under the 60-step oracle differs from the corresponding 5000-step estimate by at most 2.1×10^{-3} of the sampling domain. Reference samples are generated offline by uniform box rejection, yielding uniform samples over $\mathcal{C}_s$.

Thickened Hyperplane Benchmark. The thickened hyperplane benchmark evaluates generation on a positive-volume neighborhood of a hyperplane, rather than on a zero-volume manifold. In $d = 10$, the oracle-defined set is

$$\mathcal{C}_s = \left\{ x \in [-1, 1]^{10} : |w^\top x - b| \leq \epsilon \right\}, \tag{32}$$

with $\epsilon = 5 \times 10^{-4}$. The implementation uses a fixed unit normal $w = (1, \ldots, 1)/\sqrt{10}$ and $b = 0$, so this stress test does not vary the condition. Uniform reference proposals draw $d - 1$ coordinates and a transverse offset uniformly, solve for the remaining coordinate, and reject points outside the box. The coordinate transformation has constant Jacobian, so accepted proposals are uniform on the slab.

Study	**Geometry**	**Dim.** d	**OR Comps.** m	**Parameters**
Geometry	Box, Ball, Solid Torus	$\{2, 8, 16\}$	3	Ranges in text
2D Hard Shapes	Julia	2	—	Fixed
Thickened Hyperplane	Slab	10	1	Fixed ($\epsilon = 5 \times 10^{-4}$)
Dimension Scaling	Box, Solid Torus	$\{16, 32, 64\}$	3	Ranges in text
OR Scaling	Solid Torus	16	$\{3, 4, 8, 16, 32\}$	Ranges in text

Table 4: **Geometric benchmark sweeps.** Controlled variation of geometry, ambient dimension, and disjunctive complexity.

C.2 Robotics Benchmark

The robotics benchmark evaluates conditional transport under physics-derived MuJoCo (Todorov et al., 2012) boundaries over the compact output domain $\mathcal{A} = [-1, 1]^d$. Here, the conditioning variable $s = (\theta, \omega) \in \mathcal{S}$ collects joint positions θ and angular velocities $\omega = \dot{\theta}$. Because boundary limits depend directly on kinematic and energetic quantities in s, the oracle-defined set $\mathcal{C}_s = \{x \in \mathcal{A} : \mathcal{O}_s(x) = 1\}$ reconfigures across state transitions (Table 5). For each banked state, certified reference actions are obtained by uniform proposals from $\mathcal{A}$ followed by hard-oracle rejection, so accepted actions are distributed as $\mathrm{Unif}(\mathcal{C}_s)$ conditional on that state.

Env.	d	**Membership Oracle** $\mathcal{O}_s(x)$	**Boundary Character**
HC+O	6	$\mathbb{I}\left[x \in \mathcal{A} \wedge \sum_{i=1}^{6} \lvert\omega_i x_i\rvert \leq 20\right]$	Symmetric weighted-ℓ_1 (power)
W+M	6	$\mathbb{I}\left[x \in \mathcal{A} \wedge \sum_{i=1}^{6} \max(\bar{\omega}_i x_i, 0) \leq 10\right]$	Asymmetric piecewise-linear (work)
H+O+S	3	$\mathbb{I}\left[x \in \mathcal{A} \wedge \sum_{i=1}^{3} \lvert\bar{\omega}_i x_i\rvert \leq 10 \wedge \sum_{i=1}^{3} x_i^2 \sin^2\theta_i \leq 0.1\right]$	Weighted-$\ell_1 \cap$ ellipsoid
W+O+S	6	$\mathbb{I}\left[x \in \mathcal{A} \wedge \sum_{i=1}^{6} \lvert\bar{\omega}_i x_i\rvert \leq 10 \wedge \sum_{i=1}^{6} x_i^2 \sin^2\theta_i \leq 0.1\right]$	Weighted-$\ell_1 \cap$ ellipsoid

Table 5: **Continuous-control benchmarks.** Physics-derived membership oracles over the action domain $\mathcal{A} = [-1, 1]^d$.

Kinematic and Composite Boundaries. The suite spans two standard benchmarks. The oracles read the joint angles θ and velocities ω from the observation; for the Hopper and Walker2d tasks $\bar{\omega}_i = \mathrm{clip}(\omega_i, -10, 10)$ is the velocity as gym clips it in the observation, while HC+O reads ω unclipped. From Brahmanage et al. (2023), HC+O (HalfCheetah) and W+M (Walker2d) enforce isolated energetic boundaries in $d = 6$: HC+O restricts total power expenditure, rescaling the coordinate intercepts with joint velocities, while W+M penalizes solely positive work, generating an asymmetric polytope. From Kasaura et al. (2023), H+O+S (Hopper, $d = 3$) and W+O+S (Walker2d, $d = 6$) enforce composite boundaries formed by the intersection of a non-smooth weighted-ℓ_1 power limit with a smooth, configuration-dependent ellipsoidal bound, enabling a direct 3D $\rightarrow$ 6D comparison under the same constraint family.

C.3 Trajectory Benchmark

The trajectory planning benchmark evaluates conditional generative models over multi-step trajectory spaces $x \in \mathbb{R}^{H \times (d_s + d_a)}$ under obstacle and boundary constraints, adopting the experimental

suites, simulators, and evaluation loops of HardFlow (Li et al., 2026). We consider two distinct navigation regimes: an obstacle-avoidance domain requiring receding-horizon replanning, and a long-horizon maze navigation domain executed under low-level proportional-derivative (PD) tracking.

A plan stacks H actions and states, $x = (u_{1:H}, s_{1:H})$ with $d_a = 2$ and $d_s = 4$, and the oracle grades its predicted execution from the current state s,

$$\mathcal{O}_s(x) = \mathbb{I}\big[\rho\big(p_h(x; s)\big) \geq \delta \text{ for all } h = 1, \ldots, H-1\big], \tag{33}$$

where p_h is the planar position reached at step h under the task's execution model and ρ is the task-specific clearance or level-set score for the oracle obstacle set: a plan is admissible only if every one of its predicted positions $h = 1, \ldots, H-1$ is ($h = 0$ is the current state). When the current clearance d_0 is already below δ, the margin required at step h is $\min\{\delta, \max(d_0, 0.002) + 0.003\, h\}$, so a plan that starts inside the margin must recover it. Boundedness is not a rejection criterion here: actions are clamped to the action box before the predicted rollout. The execution model, obstacle set, and margin δ of each task are given below.

Obstacle Avoidance (`avoiding-v0`). In `avoiding-v0`, a planar agent navigates toward a goal region past six fixed pillars, a circular obstacle, and boundary lines that HardFlow adds to the scene. At each planning cycle, the model predicts an open-loop plan of horizon $H = 16$. The agent executes the first 8 steps before replanning in a receding-horizon fashion. The oracle grades the plan's predicted execution: its actions, clamped to the action box, are rolled out under the benchmark's fitted linear dynamics, and every resulting position must keep the safety margin $\delta = 0.02$ from the obstacle set. Actual execution safety is evaluated at margin 0; any collision immediately terminates the episode, so episode length reflects both efficiency and survival. Trajectories are ranked using the benchmark's native `distance` task value model with weights $0.1/10$ balancing progress against control effort.

Maze Navigation (`maze2d-large-v1`). In `maze2d-large-v1`, an agent navigates a simple 2D maze with wall obstacles from a fixed initial position. The model plans a single global trajectory over horizon $H = 384$, which is subsequently tracked by an internal PD controller for 800 simulation steps. The constraint set consists of the ellipse and the superellipse (exponent 4) that HardFlow adds to the maze; the maze walls belong to the simulator, not to the oracle. The oracle grades the predicted PD-tracked execution of the planned states under the benchmark's fitted linear dynamics (gains 5 and 1, actions clipped), which must keep the level-set margin $\delta = 0.4$ from both regions at every step; task value weights are set to $2.0/0.1$. We count an episode as successful if its normalized D4RL (Fu et al., 2020) score exceeds 0.5; the threshold is ours, and the scores are bimodal, so no episode lies near it.

D Experimental Setup

D.1 Training and Evaluation Procedure

Across all benchmark suites, experiments follow a structured three-stage lineage: base pretraining (Stage 1), constraint adaptation and steering (Stage 2), and downstream task evaluation (Stage 3). Flow adaptation methods—FM-RE, Flow Expander(FE), and our CSFM model—share the identical seed-specific base flow whenever their respective methods permit. Standalone baselines (KDE mappings, FlowPG) are trained independently from scratch; KDE uses the flows' set network as a one-pass generator, while FlowPG keeps its native RealNVP parameterization. We enforce architectural equality within the FM-derived family and KDE, while FlowPG retains its native architecture; all methods are matched on training data and evaluation states, and the query budgets of every arm are stated below.

D.1.1 Stage 1: Base Flow-Matching Pretraining Across Suites

In Stage 1, each environment and seed trains a conditional optimal-transport flow-matching (OT-FM) base model using minibatch Sinkhorn coupling. While all suites share the flow-matching objective, the model architectures, training schedules, and sources of target supervision reflect the distinct characteristics of each suite:

- **Suite A (Synthetic Geometric Domains):** Models are trained from scratch using the Adam optimizer (lr $= 3 \times 10^{-4}$) on minibatches of 32 states $\times$ 256 target points, with the source uniform on $[-1, 1]^d$. Training schedules scale with dimension: 20,000 updates for $d \leq 8$, 40,000 for $d = 16$, and 80,000 for $d \geq 32$ (20,000 for the Julia set and the $d = 10$ slab); the Gaussian-prior comparison of Table 9 uses 20,000-update bases at every d. Target points are sampled online via exact volume-weighted coordinate transformations (Appendix C.1), granting direct procedural access to $\mathrm{Unif}(\mathcal{C}_s)$. For the Julia set, points are collected by box rejection sampling, and scenes 0–3 train, 4–5 validate and 6–7 test; the thickened hyperplane stress test uses a fixed condition.
- **Suite B (Physics-Derived Continuous-Control Domains):** Models are trained from scratch with Adam (lr $= 3 \times 10^{-4}$) for 300,000 updates on the action domain $\mathcal{A} = [-1, 1]^d$. Because continuous-control kinematics and power boundaries lack closed-form volume samplers, targets are acquired offline into a fixed training bank of 20,000 states $\times$ 256 uniform feasible actions, generated by uniform proposals on $\mathcal{A}$ accepted by the hard oracle.
- **Suite C (Multi-Step Trajectory Planning Domains):** Rather than training from scratch, models adopt the 1D convolutional U-Net architecture of HardFlow (Li et al., 2026) and warm-start from HardFlow's released base checkpoint, which is fitted to the benchmark's demonstration trajectories ($H = 16$ plans in `avoiding-v0` and $H = 384$ plans in `maze2d-large-v1`); the model generates full kinematic plans conditioned on the agent's state.

Stage 1 Checkpoint Selection. Because base generative fidelity and coverage do not evolve monotonically throughout training, checkpoints are validated periodically (every 1,000 updates in Suite A, every 10,000 in Suite B) on the first 64 of the 512 evaluation states, with fresh latents (256 draws) and 128 references; these states are disjoint from training but not from the final evaluation set, and the rule and the states are identical for every arm (Julia selects on its two validation scenes). We select and freeze the base checkpoint maximizing the joint selection score $S_{\text{select}} = \text{Validity} \times \text{Coverage}$; Base rows report this selected starting checkpoint. In Suite C the released checkpoint is used as is.

D.1.2 Stage 2: Constraint Adaptation and Steering Across Suites

Starting from the identical seed-matched Stage 1 base checkpoint, each constraint-adaptation method applies its respective objective: FM-RE applies its randomized-exploration update, FM-DD its differentiable distance-to-set penalty ($\lambda = 20$, product coupling), Flow Expander expands the push-forward measure, and CSFM applies our characteristic steering loss. In Suites A and B, FM-RE, FM-DD and CSFM adapt for 5,000 updates using the same optimizer, learning rates, and batch configurations as pretraining, with CSFM setting the steering strength to $\alpha = 0.1$, $\lambda = 1$ and the constant allocation $w(t) = 1$ with direct nearest-anchor correspondences. Flow Expander runs its released schedule of 10 rounds of 4 expansion and 4 projection updates (80 updates at learning rate 10^{-4}, the G-FE variant without a pull toward the pre-trained flow). Its projection step is the zeroth-order variant on the binary oracle (a score-function update with a standardized advantage on the validity bits of its own stochastic rollouts, the route the authors prescribe for non-differentiable verifiers), not the smoothed differentiable verifier of their experiments, and its expansion step reads only the score of its own iterate off its velocity field, so FE receives the same membership information as the other arms. In Suite C, CSFM and FM-RE adapt HardFlow's checkpoint for 5,000 updates on `avoiding-v0` and 2,000 updates on `maze2d-large-v1`; CSFM issues 81,952 and 5,128 oracle queries per update (decodes, chord refinement and demonstration grading) and FM-RE's ray counts are chosen so that it issues 81,920 and 5,120, a matched budget, with exact forward-mode JVPs, Polyak updates of the reference field and chord-refined correspondences; the Suite C protocol and provenance are detailed in Appendix F.4.

Stage 2 Checkpoint Selection. During adaptation, candidate checkpoints are evaluated every 500 updates on the same 64 states using the identical selection metric S_{select}, and every arm reports the checkpoint with the highest score for downstream evaluation in Stage 3. Suite C has no uniform validation bank: CSFM and FM-RE both deploy the checkpoint with the highest probe validity (Appendix F.4).

Query accounting. For the main direct-anchor CSFM configuration in Suites A and B, grading 32×256 reference decodes costs $8{,}192$ membership evaluations per update, or $40{,}960{,}000$ over $5{,}000$ updates. FM-RE uses $32 \times 2{,}560 = 81{,}920$ evaluations per update, or $409{,}600{,}000$ over the same number of updates. These are adaptation counts, excluding acquisition of certified banks, validation, and deployment. Chord refinement adds 24 queries per corrected endpoint. Consequently, matching update counts does not match oracle budgets, and the chord ablation changes query cost as well as correspondence choice.

D.1.3 Stage 3: Downstream Evaluation and Per-Suite Protocols

In Stage 3, we evaluate the frozen adapted generators. We assess both intrinsic generative quality across all suites and downstream task performance according to each suite's operational domain:

- **Generative Quality Evaluation:** For Suites A and B (Tables 6 to 9 and 10), models decode $N = 2{,}048$ latent endpoints over 512 unseen test states (Julia: its two test scenes) against $M = 1{,}024$ certified reference samples (5 repeats, which resample the references only; Suite A references are drawn with replacement from the volume-uniform sampler, about 6% repeated rows, Suite B and Julia banks without); Jacobian diagnostics use 256 states $\times$ 128 latents. FM-RE samples through its two-stage sampler (base field for $t < 0.6$, adapted field thereafter); FE is evaluated under a Gaussian prior both as a deterministic flow and with its 40-step stochastic sampler. For Suite C, generative quality is tracked over fixed probe sets (64 fixed states $\times$ 64 draws on `avoiding-v0`, and 32 fixed states $\times$ 16 draws on `maze2d-large-v1`, both with a fixed seed shared by every arm).
- **Suite B (Continuous-Control Reinforcement Learning):** downstream performance is evaluated by deploying frozen action charts within the *action-mapping* RL paradigm:
  - *Latent-Critic Action-Mapping Paradigm:* Following Theile et al. (2025) and Kasaura et al. (2023), the actor $\pi_\theta(s)$ and twin critics $Q_\phi(s, z)$ operate in the unconstrained latent action space $z \in [-1, 1]^d$: exploration noise is added to z, the replay buffer stores z, and an environment wrapper decodes $x = T(s, z)$ and, when enabled, repairs a rejected action by chord bisection: 32 fresh prior latents of the same state are decoded and graded, 4 certified decodes are drawn at random as anchors, each chord is probed at 16 points and the chord with the nearest crossing is bisected 8 times (72 queries per repaired action); if none of the 32 decodes is certified, the raw action is executed. Separate policies are trained and evaluated with repair off and on.
  - *TD3 Policy Optimization:* The policy is optimized via Twin Delayed DDPG (TD3) (Fujimoto et al., 2018). Actor and twin critics are ReLU MLPs with hidden widths $(400, 300)$, trained with Adam (10^{-3}, batch size 100 on HC+O, W+M, W+O+S; 3×10^{-4}, batch size 256 on H+O+S), discount factor $\gamma = 0.99$, Polyak rate $\tau = 0.005$, policy delay 2, target noise $\epsilon \sim \mathrm{clip}(\mathcal{N}(0, 0.2^2), -0.5, 0.5)$, and exploration noise $\sigma = 0.1$.
  - *Training and Evaluation Schedule:* Policies train for 10^6 environment steps across $N_{\mathrm{envs}} = 64$ parallel environments with update-to-data ratio 1 after a $10{,}000$-step warmup. Every $5{,}000$ environment steps the current actor is scored on 5 deterministic validation episodes drawn from a fixed stream that is disjoint from training; the checkpoint with the highest validation return is kept. The return reported in Table 12 is that checkpoint's score on 50 fresh deterministic test episodes from a third stream, which is never used for selection. Action validity and repair displacement (Table 11) are aggregated during training and therefore describe a different state distribution and period than the final return.
- **Suite C (Closed-Loop Trajectory Planning):** frozen flows are deployed through the benchmark's own evaluation loop under a single-round policy (value selection among 16 repaired decodes) and a CEM policy (3 rounds of $1{,}024$ or 256 latents scored by the value model, 16 graded proposals, chord repair, and a speed-primitive fallback in `avoiding-v0`), for 50 episodes per seed. Safety is the benchmark's margin-0 check on the executed trajectory; latency is the mean wall-clock time per replan, each method alone on one RTX 5090 (Appendix F.4). Budgets, checkpoint rules and the deployment routine are detailed in Appendix F.4.

E EVALUATION METRICS

A learned chart over an oracle-defined set can fail in several distinct ways: it can generate invalid points, cover only a limited portion of the feasible region, skew its induced density, or exhibit excessive local geometric distortion. Because individual metrics can be misleading in isolation, we report complementary diagnostics of both the induced distribution and the local geometry of the learned chart. All metrics are evaluated conditionally within fixed states $s \sim p(s)$ using N generated endpoints $\hat{x}_i = T_\theta(s, z_i)$ with $z_i \sim q$, and averaged across test states.

Validity. The single-decode feasibility rate measures adherence to the oracle:

$$\mathrm{Val} = \mathbb{E}_{z\sim q}\left[\mathcal{O}_s(T_\theta(s,z))\right]. \tag{34}$$

Support and Coverage. Given M certified target reference samples $\{x_j^\star\}_{j=1}^M \sim \mathrm{Unif}(\mathcal{C}_s)$, coverage (Naeem et al., 2020; Kynkäänniemi et al., 2019) measures the fraction of reference points whose k-nearest-neighbor sphere contains at least one certified generated endpoint:

$$\mathrm{Cov} = \frac{1}{M}\sum_{j=1}^{M} \mathbb{I}\left[\exists\, i \in \{1,\dots,N\} \text{ s.t. } \mathcal{O}_s(\hat{x}_i) = 1 \ \wedge\ \|\hat{x}_i - x_j^\star\|_2 \le R_k(x_j^\star)\right], \tag{35}$$

where $R_k(x_j^\star)$ is the Euclidean distance from $x_j^\star$ to its k-th nearest neighbor in $\{x_l^\star\}_{l\neq j}$; we use $k = 5$.

Energy Distance. Given generated samples $\{\hat{x}_i\}_{i=1}^N$ and reference samples $\{x_j^\star\}_{j=1}^M$, we estimate the energy distance (Székely & Rizzo, 2013), normalized by the mean pairwise distance of the reference cloud, as

$$\mathcal{E} = \frac{1}{\bar{D}_\star}\left[\frac{2}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}\|\hat{x}_i - x_j^\star\|_2 - \frac{1}{N(N-1)}\sum_{i\neq i'}\|\hat{x}_i - \hat{x}_{i'}\|_2 - \bar{D}_\star\right],$$
$$\bar{D}_\star = \frac{1}{M(M-1)}\sum_{j\neq j'}\|x_j^\star - x_{j'}^\star\|_2. \tag{36}$$

All generated endpoints enter, rejected ones included, and the tables report $\mathcal{E} \times 10^3$. We use this metric as a complementary global distributional diagnostic, particularly in dimensions where local k-nearest-neighbor coverage becomes less informative (Weed & Bach, 2019).

Differential Geometry and Chart Regularity. Let $J^\theta_{0\to1}(z;s) \in \mathbb{R}^{d\times d}$ have singular values $\sigma_1(z) \ge \cdots \ge \sigma_d(z) > 0$. We evaluate coordinate distortion:

- **Volume Distortion** (VD)**:** Spatial variance of logarithmic Jacobian determinants,

$$\mathrm{VD} = \mathrm{Var}_{z\sim q}\left(\log|\det J^\theta_{0\to1}(z;s)|\right). \tag{37}$$

  Thus $\mathrm{VD} = 0$ indicates constant local volume scaling across latent space. For a uniform source and uniform target, the ideal constant is the global volume ratio $|\det J^\theta_{0\to1}(z;s)| = \frac{\mathrm{Vol}(\mathcal{C}_s)}{\mathrm{Vol}(\mathcal{Z})}$ whenever an exact injective transport exists (Dacorogna & Moser, 1990). For a Gaussian source, an exact transport onto a uniform target instead has $\log|\det J| = \log q(z) + \mathrm{const}$ and hence $\mathrm{VD} = \mathrm{Var}(\|z\|_2^2/2) = d/2$, so VD is comparable only within one source prior.

- **Shape Distortion** (SD)**:** Local metric shear, measured as the mean variance of log-singular values:

$$\mathrm{SD} = \mathbb{E}_{z\sim q}\left[\frac{1}{d}\sum_{k=1}^{d}\left(\log\sigma_k(z) - \bar{\mu}_\sigma(z)\right)^2\right], \quad \bar{\mu}_\sigma(z) = \frac{1}{d}\sum_{k=1}^{d}\log\sigma_k(z). \tag{38}$$

  $\mathrm{SD} = 0$ indicates local conformality (pure rotation and isotropic scaling).

Reporting conventions. Unless stated otherwise, entries report mean $\pm$ population standard deviation across seeds, which describes seed variability rather than a confidence interval. Suite A uses 3 seeds and Suite B uses 5; Suite C uses 5 seeds for CSFM, Base, and HardFlow and 3 for FM-RE. **Val.** and **Cov.** denote validity and coverage; **ED** is normalized energy distance, reported as $\mathrm{ED} \times 10^3$; **VD** and **SD** denote volume and shape distortion. A dash denotes an unreported metric. Distortion is omitted when any seed has validity below 0.1; † marks cells containing seeds on both sides of this threshold. Jacobian metrics are not reported for KDE or Flow Expander's stochastic sampler. Values rounded to 1.000 do not imply an exact guarantee.

F Supplementary Material: Full Experimental Results and Ablations

F.1 Suite A: Geometric Benchmark, Stress Tests, and Latent Preimages

This section collects the full quantitative results, dimension scaling, disjunctive complexity ladders, stress tests, and latent preimage visualizations for the synthetic geometric benchmarks of Suite A. Tables 6, 6b and 7 report the complete metric evaluations (validity, coverage, energy distance, volume distortion, and shape distortion) under the uniform source prior, while Table 9 reports the corresponding evaluations under a Gaussian source prior (enabling comparison with Flow Expander).

		Box					Ball					Solid Torus				
Method	d	**Val.** $\uparrow$	**Cov.** $\uparrow$	**ED** $\times 10^3 \downarrow$	**VD** $\downarrow$	**SD** $\downarrow$	**Val.** $\uparrow$	**Cov.** $\uparrow$	**ED** $\times 10^3 \downarrow$	**VD** $\downarrow$	**SD** $\downarrow$	**Val.** $\uparrow$	**Cov.** $\uparrow$	**ED** $\times 10^3 \downarrow$	**VD** $\downarrow$	**SD** $\downarrow$
Base	2	$0.961_{\pm.002}$	$0.990_{\pm.000}$	$0.661_{\pm.072}$	$0.135_{\pm.003}$	$0.253_{\pm.005}$	$0.921_{\pm.010}$	$0.980_{\pm.006}$	$2.24_{\pm.34}$	$0.235_{\pm.007}$	$0.384_{\pm.012}$	$0.939_{\pm.003}$	$0.990_{\pm.000}$	$0.415_{\pm.013}$	$0.178_{\pm.008}$	$0.454_{\pm.003}$
FM-RE	2	$1.000_{\pm.000}$	$0.963_{\pm.008}$	$0.864_{\pm.103}$	$0.215_{\pm.034}$	$0.344_{\pm.010}$	$0.999_{\pm.000}$	$0.865_{\pm.021}$	$3.09_{\pm.55}$	$0.458_{\pm.100}$	$0.462_{\pm.070}$	$1.000_{\pm.000}$	$0.959_{\pm.005}$	$0.522_{\pm.031}$	$0.312_{\pm.018}$	$0.807_{\pm.017}$
FM-DD	2	$0.986_{\pm.003}$	$0.654_{\pm.035}$	$41.4_{\pm9.2}$	$2.33_{\pm.93}$	$0.703_{\pm.231}$	$0.950_{\pm.020}$	$0.580_{\pm.026}$	$245_{\pm70}$	$3.56_{\pm1.12}$	$1.03_{\pm.31}$	$0.983_{\pm.004}$	$0.783_{\pm.024}$	$23.0_{\pm3.3}$	$1.57_{\pm.45}$	$1.24_{\pm.12}$
CSFM	2	$0.984_{\pm.001}$	$0.992_{\pm.000}$	$0.342_{\pm.062}$	$0.092_{\pm.004}$	$0.334_{\pm.005}$	$0.945_{\pm.004}$	$0.983_{\pm.003}$	$1.85_{\pm.39}$	$0.229_{\pm.005}$	$0.425_{\pm.018}$	$0.973_{\pm.001}$	$0.992_{\pm.001}$	$0.251_{\pm.018}$	$0.126_{\pm.006}$	$0.576_{\pm.010}$
Base	8	$0.829_{\pm.013}$	$0.990_{\pm.000}$	$1.58_{\pm.34}$	$0.313_{\pm.019}$	$0.235_{\pm.009}$	$0.877_{\pm.013}$	$0.985_{\pm.002}$	$7.03_{\pm.82}$	$0.765_{\pm.096}$	$0.277_{\pm.007}$	$0.892_{\pm.019}$	$0.994_{\pm.001}$	$1.21_{\pm.12}$	$1.43_{\pm.12}$	$0.887_{\pm.010}$
FM-RE	8	$1.000_{\pm.000}$	$0.996_{\pm.000}$	$2.45_{\pm.23}$	$0.853_{\pm.045}$	$0.360_{\pm.010}$	$0.999_{\pm.000}$	$0.947_{\pm.012}$	$14.7_{\pm1.3}$	$1.05_{\pm.06}$	$0.183_{\pm.011}$	$1.000_{\pm.000}$	$0.992_{\pm.001}$	$1.86_{\pm.15}$	$1.15_{\pm.24}$	$1.03_{\pm.05}$
FM-DD	8	$0.965_{\pm.013}$	$0.915_{\pm.001}$	$99.2_{\pm12.6}$	$3.21_{\pm1.25}$	$0.339_{\pm.060}$	$0.936_{\pm.010}$	$0.628_{\pm.026}$	$449_{\pm100}$	$7.81_{\pm.72}$	$0.381_{\pm.028}$	$0.966_{\pm.004}$	$0.877_{\pm.056}$	$110_{\pm41}$	$5.17_{\pm1.80}$	$1.32_{\pm.20}$
CSFM	8	$0.958_{\pm.005}$	$0.994_{\pm.000}$	$1.03_{\pm.04}$	$0.274_{\pm.108}$	$0.283_{\pm.011}$	$0.951_{\pm.003}$	$0.990_{\pm.000}$	$3.97_{\pm.11}$	$0.762_{\pm.019}$	$0.310_{\pm.021}$	$0.953_{\pm.001}$	$0.996_{\pm.000}$	$0.810_{\pm.030}$	$1.38_{\pm.02}$	$0.927_{\pm.004}$
Base	16	$0.750_{\pm.011}$	$0.991_{\pm.000}$	$1.70_{\pm.13}$	$0.423_{\pm.026}$	$0.135_{\pm.002}$	$0.924_{\pm.017}$	$0.995_{\pm.000}$	$13.5_{\pm.8}$	$2.81_{\pm.20}$	$0.250_{\pm.013}$	$0.967_{\pm.004}$	$0.999_{\pm.000}$	$1.38_{\pm.17}$	$3.78_{\pm.42}$	$0.766_{\pm.027}$
FM-RE	16	$1.000_{\pm.000}$	$0.998_{\pm.000}$	$2.69_{\pm.24}$	$1.77_{\pm.07}$	$0.251_{\pm.005}$	$1.000_{\pm.000}$	$0.983_{\pm.004}$	$37.4_{\pm4.1}$	$1.90_{\pm.12}$	$0.112_{\pm.007}$	$1.000_{\pm.000}$	$0.999_{\pm.000}$	$2.31_{\pm.26}$	$4.38_{\pm.93}$	$0.840_{\pm.057}$
FM-DD	16	$0.934_{\pm.008}$	$0.983_{\pm.005}$	$66.1_{\pm11.1}$	$1.75_{\pm.50}$	$0.202_{\pm.025}$	$0.921_{\pm.009}$	$0.811_{\pm.017}$	$430_{\pm53}$	$6.79_{\pm.74}$	$0.222_{\pm.025}$	$0.927_{\pm.012}$	$0.898_{\pm.034}$	$75.7_{\pm38.4}$	$16.0_{\pm8.0}$	$1.16_{\pm.19}$
CSFM	16	$0.962_{\pm.004}$	$0.997_{\pm.000}$	$1.47_{\pm.10}$	$0.671_{\pm.339}$	$0.158_{\pm.004}$	$0.955_{\pm.004}$	$0.997_{\pm.000}$	$10.4_{\pm1.2}$	$3.37_{\pm.46}$	$0.266_{\pm.021}$	$0.973_{\pm.003}$	$1.000_{\pm.000}$	$1.00_{\pm.04}$	$7.32_{\pm.41}$	$0.808_{\pm.005}$
Scaling with ambient dimension (Table 5b): Box and Solid Torus, $m = 3$; the $d = 16$ rows above continue here, Ball is not run beyond $d = 16$																
Base	32	$0.674_{\pm.007}$	$0.992_{\pm.000}$	$2.35_{\pm.25}$	$1.06_{\pm.10}$	$0.065_{\pm.001}$	–	–	–	–	–	$0.986_{\pm.000}$	$1.000_{\pm.000}$	$1.67_{\pm.03}$	$13.3_{\pm1.6}$	$0.620_{\pm.009}$
FM-RE	32	$0.999_{\pm.000}$	$1.000_{\pm.000}$	$4.38_{\pm.20}$	$4.66_{\pm.26}$	$0.203_{\pm.010}$	–	–	–	–	–	$0.667_{\pm.471}^{\dagger}$	$0.665_{\pm.470}$	$43.2_{\pm56.1}$	–	–
CSFM	32	$0.962_{\pm.010}$	$0.999_{\pm.000}$	$1.96_{\pm.18}$	$0.931_{\pm.035}$	$0.071_{\pm.002}$	–	–	–	–	–	$0.990_{\pm.000}$	$1.000_{\pm.000}$	$1.70_{\pm.10}$	$44.5_{\pm19.3}$	$0.670_{\pm.012}$
Base	64	$0.113_{\pm.000}$	$0.941_{\pm.002}$	$44.0_{\pm2.3}$	$3.39_{\pm.46}$	$0.105_{\pm.001}$	–	–	–	–	–	$0.698_{\pm.006}$	$1.000_{\pm.000}$	$5.34_{\pm.12}$	$1324_{\pm60}$	$0.410_{\pm.011}$
FM-RE	64	$0.000_{\pm.000}$	$0.000_{\pm.000}$	$115_{\pm20}$	–	–	–	–	–	–	–	$0.000_{\pm.000}$	$0.000_{\pm.000}$	$198_{\pm18}$	–	–
CSFM	64	$0.273_{\pm.010}$	$0.996_{\pm.001}$	$41.3_{\pm1.8}$	$5.87_{\pm.08}$	$0.086_{\pm.001}$	–	–	–	–	–	$0.819_{\pm.019}$	$1.000_{\pm.000}$	$4.92_{\pm.35}$	$962_{\pm38}$	$0.450_{\pm.014}$

Table 6: **Geometric benchmark under a uniform source prior.** Box, Ball, and Solid Torus at $d \in \{2, 8, 16\}$, followed below the heavy rule (Table 6b) by the Box and Solid Torus dimension ladder to $d = 64$; $m = 3$ throughout. FM-RE uses its prescribed two-stage sampler here and in every later table. Metrics and reporting conventions are defined in Appendix E.

Method	m	**Val.** $\uparrow$	**Cov.** $\uparrow$	**ED** $\times 10^3 \downarrow$	**VD** $\downarrow$	**SD** $\downarrow$
Base	3	$0.967_{\pm.004}$	$0.999_{\pm.000}$	$1.38_{\pm.17}$	$3.78_{\pm.42}$	$0.766_{\pm.027}$
FM-RE	3	$1.000_{\pm.000}$	$0.999_{\pm.000}$	$2.31_{\pm.26}$	$4.38_{\pm.93}$	$0.840_{\pm.057}$
CSFM	3	$0.973_{\pm.003}$	$1.000_{\pm.000}$	$1.00_{\pm.04}$	$7.32_{\pm.41}$	$0.808_{\pm.005}$
Base	4	$0.957_{\pm.011}$	$0.999_{\pm.000}$	$1.13_{\pm.09}$	$4.43_{\pm.50}$	$0.784_{\pm.024}$
FM-RE	4	$1.000_{\pm.000}$	$0.999_{\pm.001}$	$1.82_{\pm.26}$	$5.14_{\pm.16}$	$0.849_{\pm.056}$
CSFM	4	$0.966_{\pm.003}$	$1.000_{\pm.000}$	$0.861_{\pm.097}$	$7.98_{\pm.56}$	$0.816_{\pm.010}$
Base	8	$0.938_{\pm.010}$	$0.999_{\pm.000}$	$0.768_{\pm.035}$	$6.20_{\pm.09}$	$0.866_{\pm.015}$
FM-RE	8	$1.000_{\pm.000}$	$1.000_{\pm.000}$	$1.04_{\pm.06}$	$5.38_{\pm.39}$	$0.909_{\pm.057}$
CSFM	8	$0.962_{\pm.003}$	$1.000_{\pm.000}$	$0.527_{\pm.034}$	$8.43_{\pm.18}$	$0.875_{\pm.022}$
Base	16	$0.933_{\pm.008}$	$0.998_{\pm.000}$	$0.633_{\pm.030}$	$5.31_{\pm.84}$	$0.887_{\pm.057}$
FM-RE	16	$1.000_{\pm.000}$	$1.000_{\pm.000}$	$0.826_{\pm.067}$	$6.79_{\pm.84}$	$1.00_{\pm.08}$
CSFM	16	$0.962_{\pm.007}$	$0.999_{\pm.000}$	$0.409_{\pm.011}$	$7.57_{\pm1.02}$	$0.892_{\pm.030}$
Base	32	$0.903_{\pm.022}$	$0.997_{\pm.000}$	$0.474_{\pm.058}$	$5.85_{\pm1.16}$	$0.894_{\pm.010}$
FM-RE	32	$1.000_{\pm.000}$	$0.999_{\pm.000}$	$0.604_{\pm.055}$	$8.90_{\pm1.78}$	$0.999_{\pm.074}$
CSFM	32	$0.956_{\pm.012}$	$0.998_{\pm.000}$	$0.354_{\pm.018}$	$8.99_{\pm.61}$	$0.937_{\pm.005}$

Table 7: **Scaling with disjunctive complexity.** Solid Torus at $d = 16$ with $m \in \{3, 4, 8, 16, 32\}$ disjoint components of fixed geometry.

Scaling with the number of components. Figure 6 varies disjunctive complexity on the Solid Torus at $d = 16$. FM-RE and CSFM retain nearly complete coverage as the number of components grows. FM-RE retains higher validity, while CSFM has lower energy distance and shape distortion; CSFM does not consistently improve volume distortion (Table 7).

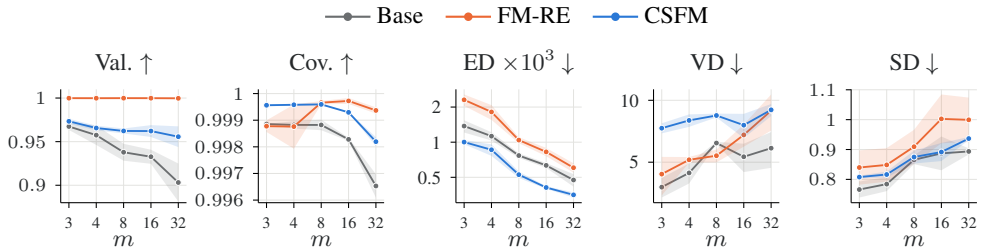


Figure 6: **Scaling with the number of components** (Solid Torus, $d = 16$). Val., Cov., ED ($\times 10^3$, log scale), VD, and SD as m increases. Values in Table 7.

	Julia ($d = 2$)					Thickened Hyperplane ($d = 10$)				
Method	**Val.** ↑	**Cov.** ↑	**ED** $\times 10^3$ ↓	**VD** ↓	**SD** ↓	**Val.** ↑	**Cov.** ↑	**ED** $\times 10^3$ ↓	**VD** ↓	**SD** ↓
Base	$0.534_{\pm.024}$	$0.828_{\pm.034}$	$18.8_{\pm3.2}$	$0.424_{\pm.023}$	$0.566_{\pm.074}$	$0.224_{\pm.034}$	$0.813_{\pm.052}$	$0.077_{\pm.036}$	$2.30_{\pm.34}$	$3.58_{\pm.12}$
FM-RE	$0.584_{\pm.087}$	$0.519_{\pm.068}$	$52.7_{\pm27.9}$	$0.696_{\pm.183}$	$0.585_{\pm.098}$	$0.110_{\pm.005}$	$0.631_{\pm.026}$	$1.49_{\pm.14}$	$5.87_{\pm1.02}$	$7.12_{\pm2.09}$
FM-DD	–	–	–	–	–	$0.268_{\pm.037}$	$0.788_{\pm.043}$	$0.831_{\pm.182}$	$2.07_{\pm.09}$	$5.70_{\pm.37}$
CSFM	$0.500_{\pm.012}$	$0.810_{\pm.020}$	$13.8_{\pm2.9}$	$0.359_{\pm.021}$	$0.569_{\pm.011}$	$0.367_{\pm.031}$	$0.939_{\pm.010}$	$0.261_{\pm.040}$	$2.71_{\pm.06}$	$3.63_{\pm.35}$

Adaptation oracle queries: FM-RE 4.096×10^8 and CSFM 4.096×10^7 on both; FM-DD reads the analytic geometry, issues none, and is not run on the Julia set.

Table 8: **Julia set and thin target region.** Julia ($d = 2$) and a thickened hyperplane ($d = 10$). FM-DD is privileged on the latter through analytic geometric access.

		Box					Ball					Solid Torus				
Method	d	**Val.** ↑	**Cov.** ↑	**ED** $\times 10^3$ ↓	**VD** ↓	**SD** ↓	**Val.** ↑	**Cov.** ↑	**ED** $\times 10^3$ ↓	**VD** ↓	**SD** ↓	**Val.** ↑	**Cov.** ↑	**ED** $\times 10^3$ ↓	**VD** ↓	**SD** ↓
Base	2	$0.967_{\pm.005}$	$0.989_{\pm.001}$	$0.654_{\pm.028}$	$0.953_{\pm.024}$	$0.315_{\pm.006}$	$0.935_{\pm.002}$	$0.975_{\pm.007}$	$2.87_{\pm.58}$	$0.962_{\pm.033}$	$0.369_{\pm.009}$	$0.953_{\pm.004}$	$0.990_{\pm.001}$	$0.454_{\pm.119}$	$1.09_{\pm.04}$	$0.526_{\pm.017}$
FE	2	$0.899_{\pm.020}$	$0.774_{\pm.034}$	$14.2_{\pm1.9}$	$0.405_{\pm.029}$	$0.401_{\pm.017}$	$0.903_{\pm.032}$	$0.408_{\pm.063}$	$45.0_{\pm5.1}$	$0.960_{\pm.141}$	$0.765_{\pm.089}$	$0.816_{\pm.089}$	$0.520_{\pm.015}$	$42.4_{\pm11.2}$	$2.42_{\pm.35}$	$2.61_{\pm.28}$
FE (SDE)	2	$0.974_{\pm.007}$	$0.804_{\pm.021}$	$30.2_{\pm3.2}$	–	–	$0.928_{\pm.005}$	$0.688_{\pm.033}$	$67.2_{\pm12.3}$	–	–	$0.938_{\pm.013}$	$0.894_{\pm.003}$	$14.0_{\pm7.1}$	–	–
FM-RE	2	$1.000_{\pm.000}$	$0.976_{\pm.005}$	$0.837_{\pm.136}$	$1.73_{\pm.20}$	$0.546_{\pm.073}$	$0.999_{\pm.001}$	$0.859_{\pm.009}$	$3.65_{\pm.73}$	$2.45_{\pm.18}$	$0.722_{\pm.054}$	$0.999_{\pm.000}$	$0.974_{\pm.004}$	$0.533_{\pm.141}$	$1.74_{\pm.07}$	$1.04_{\pm.04}$
CSFM	2	$0.983_{\pm.001}$	$0.989_{\pm.003}$	$0.613_{\pm.097}$	$1.05_{\pm.04}$	$0.397_{\pm.024}$	$0.955_{\pm.003}$	$0.983_{\pm.003}$	$2.18_{\pm.25}$	$0.989_{\pm.017}$	$0.400_{\pm.011}$	$0.975_{\pm.001}$	$0.991_{\pm.001}$	$0.294_{\pm.013}$	$1.08_{\pm.03}$	$0.634_{\pm.006}$
Base	8	$0.891_{\pm.004}$	$0.992_{\pm.000}$	$2.52_{\pm1.86}$	$2.62_{\pm.09}$	$0.388_{\pm.010}$	$0.908_{\pm.017}$	$0.984_{\pm.002}$	$7.69_{\pm.79}$	$1.44_{\pm.06}$	$0.105_{\pm.004}$	$0.932_{\pm.013}$	$0.995_{\pm.000}$	$1.33_{\pm.08}$	$2.21_{\pm.04}$	$0.755_{\pm.021}$
FE	8	$0.970_{\pm.006}$	$0.925_{\pm.053}$	$16.3_{\pm6.9}$	$4.31_{\pm.94}$	$0.587_{\pm.048}$	$0.929_{\pm.042}$	$0.683_{\pm.107}$	$40.7_{\pm10.3}$	$5.19_{\pm1.65}$	$0.546_{\pm.023}$	$0.521_{\pm.078}$	$0.828_{\pm.051}$	$10.1_{\pm1.5}$	$14.4_{\pm2.8}$	$1.39_{\pm.06}$
FE (SDE)	8	$0.991_{\pm.001}$	$0.838_{\pm.113}$	$42.5_{\pm4.2}$	–	–	$0.892_{\pm.011}$	$0.827_{\pm.010}$	$65.4_{\pm9.4}$	–	–	$0.429_{\pm.024}$	$0.891_{\pm.023}$	$23.8_{\pm4.9}$	–	–
FM-RE	8	$0.999_{\pm.000}$	$0.994_{\pm.000}$	$3.08_{\pm2.23}$	$6.86_{\pm.37}$	$0.839_{\pm.048}$	$0.999_{\pm.000}$	$0.952_{\pm.013}$	$13.7_{\pm1.6}$	$4.43_{\pm.41}$	$0.239_{\pm.055}$	$0.999_{\pm.000}$	$0.990_{\pm.004}$	$1.95_{\pm.20}$	$4.09_{\pm.21}$	$1.05_{\pm.03}$
CSFM	8	$0.977_{\pm.003}$	$0.994_{\pm.000}$	$1.45_{\pm.47}$	$3.23_{\pm.24}$	$0.487_{\pm.029}$	$0.962_{\pm.004}$	$0.991_{\pm.000}$	$4.83_{\pm.46}$	$1.58_{\pm.06}$	$0.129_{\pm.006}$	$0.969_{\pm.003}$	$0.997_{\pm.000}$	$1.07_{\pm.05}$	$1.88_{\pm.01}$	$0.780_{\pm.007}$
Base	16	$0.727_{\pm.039}$	$0.988_{\pm.002}$	$9.34_{\pm.48}$	$4.90_{\pm.18}$	$0.321_{\pm.014}$	$0.846_{\pm.024}$	$0.987_{\pm.003}$	$19.5_{\pm.5}$	$1.25_{\pm.14}$	$0.039_{\pm.001}$	$0.916_{\pm.020}$	$0.998_{\pm.000}$	$2.30_{\pm.11}$	$4.13_{\pm.78}$	$0.575_{\pm.017}$
FE	16	$0.890_{\pm.032}$	$0.985_{\pm.002}$	$20.6_{\pm2.0}$	$6.73_{\pm.70}$	$0.481_{\pm.003}$	$0.805_{\pm.058}$	$0.866_{\pm.038}$	$65.2_{\pm11.5}$	$15.3_{\pm1.5}$	$0.564_{\pm.016}$	$0.058_{\pm.026}$	$0.246_{\pm.080}$	$24.3_{\pm6.5}$	–	–
FE (SDE)	16	$0.946_{\pm.027}$	$0.984_{\pm.002}$	$34.5_{\pm1.8}$	–	–	$0.534_{\pm.049}$	$0.842_{\pm.047}$	$69.6_{\pm9.3}$	–	–	$0.014_{\pm.006}$	$0.129_{\pm.055}$	$33.4_{\pm9.3}$	–	–
FM-RE	16	$1.000_{\pm.000}$	$0.997_{\pm.000}$	$10.3_{\pm.3}$	$16.6_{\pm1.3}$	$0.937_{\pm.070}$	$0.998_{\pm.001}$	$0.905_{\pm.014}$	$64.9_{\pm2.6}$	$4.35_{\pm.43}$	$0.093_{\pm.007}$	$1.000_{\pm.000}$	$0.994_{\pm.001}$	$4.05_{\pm.36}$	$10.9_{\pm1.6}$	$1.04_{\pm.02}$
CSFM	16	$0.950_{\pm.002}$	$0.996_{\pm.000}$	$8.25_{\pm1.84}$	$5.13_{\pm.03}$	$0.368_{\pm.006}$	$0.921_{\pm.002}$	$0.993_{\pm.002}$	$20.0_{\pm2.6}$	$1.69_{\pm.21}$	$0.049_{\pm.005}$	$0.962_{\pm.009}$	$0.999_{\pm.000}$	$2.17_{\pm.13}$	$6.58_{\pm.40}$	$0.638_{\pm.018}$

Table 9: **Gaussian source prior.** Geometric benchmark with all methods trained from $\mathcal{N}(0, I)$; all rows use 20,000-update bases. FE denotes deterministic Flow Expander sampling and FE (SDE) its stochastic sampler.

F.2 Preimages

Figure 7 shows where each model sends its latent space. Each method is shown by its best of three seeds by validity $\times$ coverage of its reported record, at the selected checkpoint used for the corresponding reported record; the panel title names the seed. All panels of one figure decode the same state, drawn once with a fixed key; the Julia figure instead shows every held-out scene, one per row. The *left* panel is a two-dimensional slice through the latent box: the (z_0, z_1) plane at $d = 2$, and above it the plane $z = (\xi_0 h_0 + \xi_1 h_1)/2$ spanned by two rows of the Sylvester–Hadamard matrix, which moves every coordinate and reaches the corners of the box. Each latent is coloured by the component its decode lands in, and *grey marks the error preimage*: latents whose decode the oracle rejects. A light/dark checkerboard over the slice is carried along by the map, so the *right* panel, the decodes of the same slice in the first two target coordinates, shows how the latent grid is stretched and sheared in the displayed coordinates. The fraction of the slice and of fresh prior draws that decode outside the set is printed in each left panel. Above $d = 2$ the right panel is a projection: in the two ladders the outlines are each component's cross-section through its own center, and in the single-cell figures the tinted cloud is a uniform sample of each component, projected.

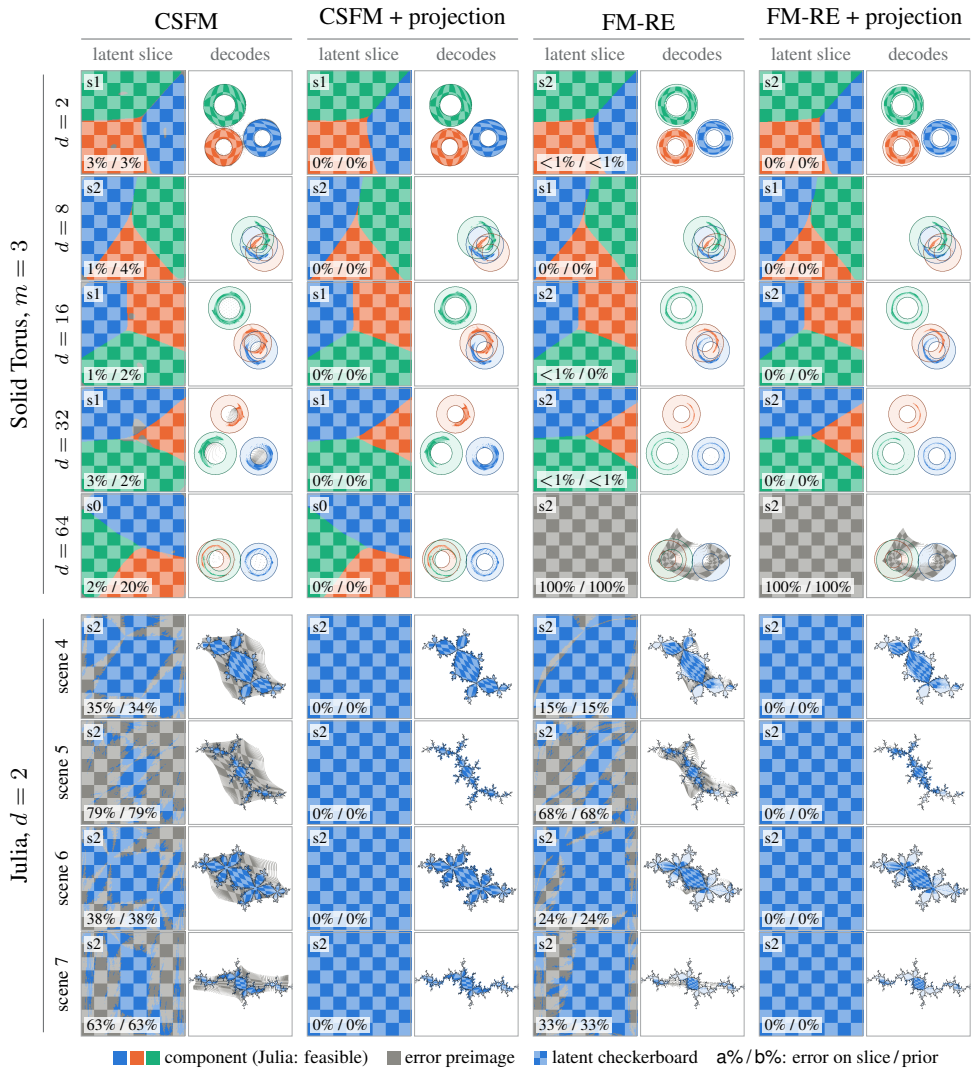


Figure 7: **Latent preimages along the dimension ladder (top) and on the Julia set (bottom).** Top: Solid Torus, $m = 3$, $d = 2$ to 64; at $d = 64$ every FM-RE seed has zero validity and its whole slice is error preimage. Bottom: one held-out Julia scene per row, validation scenes 4 and 5 (checkpoint selection) and test scenes 6 and 7 (Table 8); training used scenes 0 to 3. The second and fourth columns show CSFM and FM-RE after certified runtime projection with the deployed repair rule: each rejected decode is chorded toward its 4 nearest certified decodes among 32 prior draws of the same state, and the crossing that minimizes target plus latent distance ($\kappa = 1$) is bisected. This removes the error preimage in every displayed slice except FM-RE at $d = 64$, where none of the 32 decodes is certified and the projector has no anchor to move toward. Panels name the seed.

F.3 Suite B: State-Conditioned Robotics Generation and Downstream RL

Table 10 below details the generation quality across all methods on the four state-conditioned environments of Suite B (HC+O, W+M, H+O+S, W+O+S), and Table 12 reports the downstream reinforcement learning return with and without projection. Table 11 below provides the action validity rates during policy training and mean projection displacement under the state-conditioned membership oracle.

	HC+O					W+M				
Method	**Val.** ↑	**Cov.** ↑	**ED** $\times 10^3$ ↓	**VD** ↓	**SD** ↓	**Val.** ↑	**Cov.** ↑	**ED** $\times 10^3$ ↓	**VD** ↓	**SD** ↓
Base	$0.987_{\pm.001}$	$0.996_{\pm.000}$	$0.257_{\pm.071}$	$0.048_{\pm.003}$	$0.095_{\pm.003}$	$0.983_{\pm.001}$	$0.996_{\pm.000}$	$0.489_{\pm.079}$	$0.054_{\pm.005}$	$0.107_{\pm.003}$
FM-RE	$1.000_{\pm.000}$	$0.996_{\pm.000}$	$0.326_{\pm.063}$	$0.346_{\pm.059}$	$0.180_{\pm.018}$	$1.000_{\pm.000}$	$0.996_{\pm.000}$	$0.555_{\pm.056}$	$0.404_{\pm.032}$	$0.188_{\pm.008}$
FlowPG	$0.797_{\pm.007}$	$0.984_{\pm.001}$	$2.04_{\pm.25}$	$0.005_{\pm.002}$	$0.004_{\pm.001}$	$0.775_{\pm.011}$	$0.988_{\pm.001}$	$13.3_{\pm2.9}$	$0.075_{\pm.043}$	$0.045_{\pm.003}$
KDE	$1.000_{\pm.000}$	$0.996_{\pm.001}$	$5.86_{\pm.72}$	–	–	$1.000_{\pm.000}$	$0.996_{\pm.000}$	$6.09_{\pm.33}$	–	–
CSFM	$0.999_{\pm.001}$	$0.997_{\pm.000}$	$0.413_{\pm.082}$	$0.082_{\pm.007}$	$0.094_{\pm.005}$	$0.999_{\pm.000}$	$0.996_{\pm.000}$	$0.602_{\pm.070}$	$0.094_{\pm.010}$	$0.093_{\pm.003}$

	H+O+S					W+O+S				
Method	**Val.** ↑	**Cov.** ↑	**ED** $\times 10^3$ ↓	**VD** ↓	**SD** ↓	**Val.** ↑	**Cov.** ↑	**ED** $\times 10^3$ ↓	**VD** ↓	**SD** ↓
Base	$0.999_{\pm.000}$	$0.996_{\pm.000}$	$0.045_{\pm.014}$	$0.004_{\pm.001}$	$0.008_{\pm.001}$	$0.953_{\pm.002}$	$0.995_{\pm.000}$	$0.594_{\pm.036}$	$0.157_{\pm.012}$	$0.492_{\pm.017}$
FM-RE	$1.000_{\pm.000}$	$0.996_{\pm.000}$	$0.069_{\pm.017}$	$0.206_{\pm.032}$	$0.055_{\pm.007}$	$1.000_{\pm.000}$	$0.996_{\pm.000}$	$1.44_{\pm.40}$	$0.389_{\pm.050}$	$0.657_{\pm.034}$
FlowPG	$0.995_{\pm.001}$	$0.996_{\pm.000}$	$0.103_{\pm.023}$	$0.000_{\pm.000}$	$0.004_{\pm.000}$	$0.378_{\pm.012}$	$0.887_{\pm.007}$	$11.3_{\pm.8}$	$0.107_{\pm.014}$	$0.132_{\pm.012}$
KDE	$1.000_{\pm.000}$	$0.992_{\pm.000}$	$0.404_{\pm.036}$	–	–	$1.000_{\pm.000}$	$0.995_{\pm.000}$	$4.52_{\pm.19}$	–	–
CSFM	$1.000_{\pm.000}$	$0.996_{\pm.000}$	$0.062_{\pm.023}$	$0.006_{\pm.002}$	$0.008_{\pm.001}$	$0.994_{\pm.001}$	$0.996_{\pm.000}$	$0.773_{\pm.105}$	$0.140_{\pm.010}$	$0.497_{\pm.011}$

Table 10: **Suite B generation quality.** Val., Cov., ED, VD, and SD on the four state-conditioned continuous-control environments, evaluated on a fixed bank of unseen test states.

		HC+O		W+M		H+O+S		W+O+S	
Method	**Repair**	**Val.** ↑	**Proj. Dist.** ↓	**Val.** ↑	**Proj. Dist.** ↓	**Val.** ↑	**Proj. Dist.** ↓	**Val.** ↑	**Proj. Dist.** ↓
Base	Before	$0.480_{\pm.034}$	–	$0.452_{\pm.021}$	–	$0.664_{\pm.048}$	–	$0.285_{\pm.064}$	–
	After	$1.000_{\pm.000}$	$0.100_{\pm.011}$	$1.000_{\pm.000}$	$0.128_{\pm.027}$	$0.993_{\pm.012}$	$0.208_{\pm.048}$	$1.000_{\pm.000}$	$0.165_{\pm.032}$
FlowPG	Before	$0.432_{\pm.044}$	–	$0.531_{\pm.044}$	–	$0.582_{\pm.050}$	–	$0.197_{\pm.062}$	–
	After	$1.000_{\pm.000}$	$0.407_{\pm.051}$	$1.000_{\pm.000}$	$0.279_{\pm.085}$	$0.999_{\pm.001}$	$0.318_{\pm.036}$	$1.000_{\pm.000}$	$0.260_{\pm.023}$
KDE	Before	$0.996_{\pm.004}$	–	$0.999_{\pm.001}$	–	$0.897_{\pm.038}$	–	$0.992_{\pm.007}$	–
	After	$1.000_{\pm.000}$	$0.045_{\pm.008}$	$1.000_{\pm.000}$	$0.058_{\pm.005}$	$1.000_{\pm.000}$	$0.126_{\pm.026}$	$1.000_{\pm.000}$	$0.100_{\pm.031}$
FM-RE	Before	$0.963_{\pm.019}$	–	$0.963_{\pm.028}$	–	$0.849_{\pm.038}$	–	$0.838_{\pm.060}$	–
	After	$1.000_{\pm.000}$	$0.041_{\pm.008}$	$1.000_{\pm.000}$	$0.093_{\pm.038}$	$0.999_{\pm.001}$	$0.205_{\pm.017}$	$1.000_{\pm.000}$	$0.121_{\pm.027}$
CSFM	Before	$0.864_{\pm.058}$	–	$0.903_{\pm.040}$	–	$0.806_{\pm.067}$	–	$0.636_{\pm.068}$	–
	After	$1.000_{\pm.000}$	$0.071_{\pm.016}$	$1.000_{\pm.000}$	$0.108_{\pm.021}$	$0.999_{\pm.001}$	$0.238_{\pm.057}$	$0.998_{\pm.003}$	$0.147_{\pm.027}$

Table 11: **Action validity during policy training.** For the policies trained with runtime projection, Val. of the decoded actions before repair and of the delivered actions after repair, with the mean repair displacement (**Proj. Dist.**); the unprojected policies' Val. is in Table 12. Mean ± population SD over 5 seeds.

	HC+O				W+M				H+O+S				W+O+S			
	Raw		Proj.		Raw		Proj.		Raw		Proj.		Raw		Proj.	
Method	**Return**	**Val.**	**Return**	**Val.**	**Return**	**Val.**	**Return**	**Val.**	**Return**	**Val.**	**Return**	**Val.**	**Return**	**Val.**	**Return**	**Val.**
Base	$8634_{\pm716}$	$0.384_{\pm.015}$	$8097_{\pm689}$	$1.000_{\pm.000}$	$4274_{\pm301}$	$0.471_{\pm.061}$	$4888_{\pm432}$	$1.000_{\pm.000}$	$2729_{\pm797}$	$0.645_{\pm.026}$	$3094_{\pm95}$	$0.993_{\pm.012}$	$3865_{\pm229}$	$0.275_{\pm.033}$	$3817_{\pm292}$	$1.000_{\pm.000}$
FlowPG	$9213_{\pm656}$	$0.150_{\pm.070}$	$7917_{\pm1040}$	$1.000_{\pm.000}$	$4572_{\pm348}$	$0.548_{\pm.062}$	$4815_{\pm330}$	$1.000_{\pm.000}$	$3221_{\pm162}$	$0.601_{\pm.019}$	$2975_{\pm141}$	$0.999_{\pm.001}$	$3757_{\pm393}$	$0.120_{\pm.050}$	$3742_{\pm441}$	$1.000_{\pm.000}$
KDE	$6366_{\pm484}$	$0.994_{\pm.008}$	$6901_{\pm290}$	$1.000_{\pm.000}$	$4262_{\pm360}$	$0.999_{\pm.001}$	$4174_{\pm236}$	$1.000_{\pm.000}$	$2913_{\pm149}$	$0.875_{\pm.041}$	$2985_{\pm87}$	$1.000_{\pm.000}$	$3209_{\pm184}$	$0.988_{\pm.009}$	$3075_{\pm265}$	$1.000_{\pm.000}$
FM-RE	$7536_{\pm1259}$	$0.985_{\pm.005}$	$8791_{\pm639}$	$1.000_{\pm.000}$	$4509_{\pm225}$	$0.971_{\pm.018}$	$4444_{\pm466}$	$1.000_{\pm.000}$	$3071_{\pm115}$	$0.847_{\pm.027}$	$2730_{\pm581}$	$0.999_{\pm.001}$	$3686_{\pm71}$	$0.854_{\pm.037}$	$3639_{\pm250}$	$1.000_{\pm.000}$
CSFM	$8243_{\pm1255}$	$0.894_{\pm.044}$	$8100_{\pm1137}$	$1.000_{\pm.000}$	$4794_{\pm484}$	$0.869_{\pm.064}$	$4650_{\pm398}$	$1.000_{\pm.000}$	$3178_{\pm148}$	$0.793_{\pm.060}$	$3100_{\pm81}$	$0.999_{\pm.001}$	$3625_{\pm227}$	$0.644_{\pm.086}$	$3828_{\pm314}$	$0.998_{\pm.003}$

Table 12: **Downstream RL performance.** Return and delivered-action Val. for policies trained and evaluated with projection off (**Raw**) or on (**Proj.**). Returns are evaluated on 50 held-out test episodes using checkpoints selected on a separate validation stream; Val. is aggregated during training. Mean ± population SD over 5 seeds.

F.4 SUITE C: TRAJECTORY-LEVEL CERTIFICATION AND PLANNING BENCHMARK

Training protocol. All adapted flows start from HardFlow's released demonstration-fitted checkpoint. Avoidance adaptation uses $5{,}000$ updates with 32 states, 256 reference decodes, 32 corrections, and 64 flow-matching pairs per state. Maze adaptation uses $2{,}000$ updates with 8 states, 64 reference decodes, 8 corrections, and 16 flow-matching pairs per state. The steering runs use frozen reverse-flow JVPs and Polyak updates of the reference field. FM-RE uses $2{,}560$ rays per state in avoidance and 640 in maze. The saved seed-0 CSFM logs record $81{,}952$ membership evaluations per avoidance update and $5{,}128$ per maze update, including demonstration grading. These differ from direct-anchor adaptation costs in Suites A and B. FM-RE's ray counts, $2{,}560 = 256 + 32 \times 72$ and $640 = 64 + 8 \times 72$, match this budget by construction.

Training correspondences. The deployed CSFM runs train the constant-allocation objective of Equation equation 10 with $\alpha = 0.1$ and $\lambda = 1$; their logs record $\|c_t\| = \alpha\|r_t\|$ at every update. Unlike Suites A and B, the training correspondence is chord-refined: each rejected plan is paired with its 16 nearest certified anchors, every chord is graded on 4 points and the chord with the closest crossing is bisected 8 times, and the target is that certified crossing, which accounts for the 72 queries per rejected plan in the budgets above.

Probe and checkpoints. Probe validity uses 64 fixed states with 64 draws in avoidance and 32 fixed states with 16 draws in maze, with a shared probe seed. Every adapted flow, CSFM and FM-RE alike, deploys the checkpoint with the highest probe validity.

Deployment. The single-round policy selects by task value among 16 decoded proposals after repair. The CEM policy runs 3 rounds with $1{,}024$ samples per round in avoidance and 256 in maze. It ranks proposals by the task value model, refits a diagonal Gaussian to the top 10% with smoothing 0.7 and standard-deviation floor 0.05, and retains the best 16 proposals. These are graded, and every rejected proposal is repaired by one chord toward the nearest certified proposal of the same replan, 16 probes and 8 bisections on that chord. Avoidance additionally checks 17 speed primitives and a standstill fallback. The fixed-shape routine processes all 16 proposals, giving $16 + 16 \times 24 = 400$ queries per replan, or 418 with the avoidance fallback checks. The single-round policy executes the zero action, ungraded, when no proposal is certified, hence 400. This is the reported implementation cost, not a lower bound on repair cost. HardFlow instead evaluates constraint values and Jacobians inside an interior-point solver.

Execution and timing. Each seed has 50 evaluation episodes: 5 seeds for CSFM, the base flow and HardFlow, and 3 for FM-RE. The base flow uses one released checkpoint. Safety and success describe executed episodes; probe validity describes decoded plans before search and repair. Latency is measured separately over 10 episodes at seed 0, with each method alone on one RTX 5090, so it is not an average over the outcome-evaluation seeds.

Table 13 repeats the main-text comparison with oracle query counts and task metrics at higher precision. Benchmark and deployment settings are given in Appendix C.3. Figure 8 shows the executed trajectories behind the CEM rows, overlaid on the MuJoCo scenes.

Table 14 traces the pre-projection validity of Table 1 and the repair statistics quoted in Section 4.3. *Valid* is the share of the fixed probe set's decodes that the oracle certifies (Appendix D.1.3), evaluated at the deployed checkpoint; *travel* is the mean straight-line displacement between the first and last planned positions, in the benchmark's units, and is reported because a stationary plan can be accepted, so a flow can raise validity by moving less. The executed-plan columns classify every plan the policy executed: certified as *decoded*, certified after chord *repair*, or executed with *none* certified (the brake in `avoiding-v0`, the raw plan in `maze2d-large-v1`).

Task	Method	Safe (%) ↑	Success (%) ↑	Task metric	Oracle / replan	Latency / replan (ms) ↓
Avoidance steps ↓	HardFlow	$99.2_{\pm 1.0}$	$99.2_{\pm 1.0}$	$50.62_{\pm .14}$	–	133
	Base	$73.6_{\pm 6.5}$	$28.0_{\pm 2.5}$	$78.33_{\pm 3.45}$	400	27
	+ CEM	100	100	$50.14_{\pm .06}$	418	81
	FM-RE	$98.7_{\pm 1.9}$	$65.3_{\pm 46.2}$	$80.43_{\pm 14.69}$	400	30
	+ CEM	100	$66.7_{\pm 47.1}$	$79.67_{\pm 14.44}$	418	80
	CSFM	$99.2_{\pm 1.0}$	$98.4_{\pm .8}$	$56.25_{\pm 1.70}$	400	31
	+ CEM	100	100	$50.92_{\pm .59}$	418	81
Maze score ↑	HardFlow	100	$99.6_{\pm .8}$	$1.617_{\pm .013}$	–	1082
	Base	$28.8_{\pm 6.3}$	$97.6_{\pm 2.3}$	$1.556_{\pm .040}$	400	242
	+ CEM	$99.2_{\pm 1.0}$	100	$1.591_{\pm .003}$	400	511
	FM-RE	$57.3_{\pm 39.4}$	$32.0_{\pm 36.9}$	$0.462_{\pm .575}$	400	247
	+ CEM	$65.3_{\pm 34.7}$	$37.3_{\pm 42.0}$	$0.545_{\pm .650}$	400	499
	CSFM	$98.8_{\pm 1.6}$	100	$1.544_{\pm .018}$	400	271
	+ CEM	$96.8_{\pm 2.0}$	$98.0_{\pm 3.1}$	$1.538_{\pm .043}$	400	514

Table 13: **Suite C outcomes.** Single-round and CEM deployments; all flow-based methods include runtime repair. Oracle cost and latency are reported per replan.

		Probe		Executed plans, single-round (%)			Executed plans, CEM (%)		
Task	Flow	Val. (%)	Travel	decoded	repaired	none	decoded	repaired	none
Avoidance	Base	29.2	0.177	29.3	22.3	48.4	87.1	12.9	0
	FM-RE	$76.5_{\pm .0}$	0.151	95.2	4.5	0.3	98.8	1.2	0
	CSFM	$56.1_{\pm .7}$	0.170	87.5	9.3	3.3	94.1	5.9	0
Maze	Base	24.0	4.15	0	24.8	75.2	0	100	0
	FM-RE	$67.2_{\pm 5.9}$	4.39	55.3	19.3	25.3	60	30	10
	CSFM	$46.7_{\pm 1.0}$	4.05	8	92	0	31.2	68.8	0

Table 14: **Suite C diagnostics.** Probe Val. and trajectory travel, followed by the fraction of executed plans accepted directly, repaired, or executed without certification; shares pool all replans of a row.

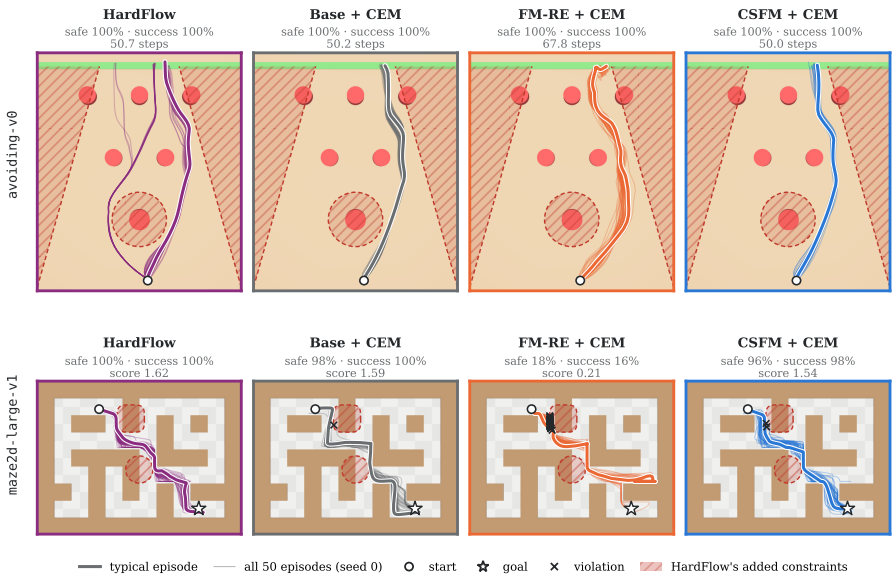


Figure 8: **Executed Suite C trajectories.** Avoidance (top) and maze (bottom) for HardFlow and the CEM-deployed flows. Each panel shows the 50 episodes of seed 0 (thin), one representative safe trajectory (bold), and all safety violations (×); hatched regions denote benchmark constraints. Numbers above the panels describe the drawn episodes, not seed means.

F.5 METHOD ABLATIONS

We conduct component and parameter ablations to isolate the contributions of the steering loss, the correspondence, and the correction magnitude α. Table 15 summarizes them on four cells (Ball $d = 16$, Solid Torus $d = 16$, Box $d = 32$, and Solid Torus $d = 32$), comparing CSFM against adaptation without steering ($\lambda = 0$), against two other correspondence rules, and against larger α. *Chord refinement* aims the displacement at the certified chord crossing (16 probes, 8 bisections; Appendix A.1) instead of the anchor, and *latent-distance anchors* are selected by target plus latent distance (Equation equation 7, $\kappa = 1$), with latents supplied by the reference reverse flow. The $\alpha = 0.5$ and $\alpha = 1$ rows prescribe a larger fraction of the displacement than CSFM's $\alpha = 0.1$; on Solid Torus at $d = 32$ the larger corrections raise the volume distortion from 44.5 to 245 and 324. Two further readings: at $d = 16$ the $\lambda = 0$ rung, five thousand further Flow-Matching updates, already reaches 0.951 and 0.970 validity against CSFM's 0.955 and 0.973, so the steering term is decisive where the base leaves the set often, as on Box at $d = 32$ (0.670 against 0.962); and chord refinement lowers Box $d = 32$ validity to 0.775, since a target at the crossing prescribes a shorter displacement, which is why Suites A and B use direct anchors. Suite C uses refined targets (Appendix F.4); that choice was not ablated there.

		Box					**Ball**					**Solid Torus**				
Variant	d	**Val.** ↑	**Cov.** ↑	**ED** $\times 10^3$ ↓	**VD** ↓	**SD** ↓	**Val.** ↑	**Cov.** ↑	**ED** $\times 10^3$ ↓	**VD** ↓	**SD** ↓	**Val.** ↑	**Cov.** ↑	**ED** $\times 10^3$ ↓	**VD** ↓	**SD** ↓
Base	16	–	–	–	–	–	$0.924_{\pm.017}$	$0.995_{\pm.000}$	$13.5_{\pm.8}$	$2.81_{\pm.20}$	$0.250_{\pm.013}$	$0.967_{\pm.004}$	$0.999_{\pm.000}$	$1.38_{\pm.17}$	$3.78_{\pm.42}$	$0.766_{\pm.027}$
FM-RE	16	–	–	–	–	–	$1.000_{\pm.000}$	$0.983_{\pm.004}$	$37.4_{\pm4.1}$	$1.90_{\pm.12}$	$0.112_{\pm.007}$	$1.000_{\pm.000}$	$0.999_{\pm.000}$	$2.31_{\pm.26}$	$4.38_{\pm.93}$	$0.840_{\pm.057}$
No correction ($\lambda = 0$)	16	–	–	–	–	–	$0.951_{\pm.001}$	$0.996_{\pm.000}$	$13.2_{\pm1.0}$	$2.61_{\pm.36}$	$0.248_{\pm.006}$	$0.970_{\pm.001}$	$0.999_{\pm.000}$	$1.25_{\pm.03}$	$3.51_{\pm.30}$	$0.777_{\pm.018}$
Chord refinement	16	–	–	–	–	–	$0.945_{\pm.005}$	$0.997_{\pm.000}$	$8.97_{\pm1.42}$	$2.82_{\pm.16}$	$0.265_{\pm.007}$	$0.976_{\pm.003}$	$0.999_{\pm.000}$	$1.04_{\pm.08}$	$4.22_{\pm.39}$	$0.790_{\pm.008}$
Latent-distance anchors	16	–	–	–	–	–	$0.960_{\pm.005}$	$0.997_{\pm.000}$	$10.9_{\pm1.0}$	$2.97_{\pm.30}$	$0.244_{\pm.003}$	$0.954_{\pm.002}$	$0.999_{\pm.000}$	$1.10_{\pm.06}$	$4.91_{\pm.54}$	$0.786_{\pm.014}$
$\alpha = 0.5$	16	–	–	–	–	–	$0.957_{\pm.015}$	$0.988_{\pm.002}$	$28.9_{\pm2.1}$	$9.58_{\pm2.16}$	$0.272_{\pm.016}$	$0.980_{\pm.003}$	$0.998_{\pm.001}$	$3.36_{\pm.46}$	$19.8_{\pm2.7}$	$0.900_{\pm.020}$
$\alpha = 1$	16	–	–	–	–	–	$0.953_{\pm.012}$	$0.914_{\pm.021}$	$62.9_{\pm10.2}$	$15.2_{\pm5.3}$	$0.280_{\pm.024}$	$0.974_{\pm.005}$	$0.989_{\pm.005}$	$6.66_{\pm3.19}$	$37.2_{\pm8.3}$	$1.09_{\pm.03}$
CSFM	16	–	–	–	–	–	$0.955_{\pm.004}$	$0.997_{\pm.000}$	$10.4_{\pm1.2}$	$3.37_{\pm.46}$	$0.266_{\pm.021}$	$0.973_{\pm.003}$	$1.000_{\pm.000}$	$1.00_{\pm.04}$	$7.32_{\pm.41}$	$0.808_{\pm.005}$
Base	32	$0.674_{\pm.007}$	$0.992_{\pm.000}$	$2.35_{\pm.25}$	$1.06_{\pm.10}$	$0.065_{\pm.001}$	–	–	–	–	–	$0.986_{\pm.000}$	$1.000_{\pm.000}$	$1.67_{\pm.03}$	$13.3_{\pm1.6}$	$0.620_{\pm.009}$
FM-RE	32	$0.999_{\pm.000}$	$1.000_{\pm.000}$	$4.38_{\pm.20}$	$4.66_{\pm.26}$	$0.203_{\pm.010}$	–	–	–	–	–	$0.667_{\pm.471}$ †	$0.665_{\pm.470}$	$43.2_{\pm56.1}$	–	–
No correction ($\lambda = 0$)	32	$0.670_{\pm.007}$	$0.993_{\pm.000}$	$2.00_{\pm.26}$	$1.04_{\pm.08}$	$0.065_{\pm.001}$	–	–	–	–	–	$0.986_{\pm.003}$	$1.000_{\pm.000}$	$1.84_{\pm.10}$	$12.8_{\pm1.0}$	$0.622_{\pm.007}$
Chord refinement	32	$0.775_{\pm.008}$	$0.995_{\pm.000}$	$1.08_{\pm.10}$	$0.953_{\pm.068}$	$0.066_{\pm.001}$	–	–	–	–	–	$0.987_{\pm.001}$	$1.000_{\pm.000}$	$1.64_{\pm.10}$	$15.8_{\pm1.9}$	$0.637_{\pm.006}$
Latent-distance anchors	32	$0.956_{\pm.001}$	$0.999_{\pm.000}$	$1.95_{\pm.34}$	$0.878_{\pm.049}$	$0.068_{\pm.003}$	–	–	–	–	–	$0.981_{\pm.002}$	$1.000_{\pm.000}$	$1.62_{\pm.07}$	$17.9_{\pm1.5}$	$0.631_{\pm.011}$
$\alpha = 0.5$	32	$0.967_{\pm.024}$	$1.000_{\pm.000}$	$12.8_{\pm2.7}$	$6.99_{\pm2.75}$	$0.117_{\pm.011}$	–	–	–	–	–	$0.984_{\pm.002}$	$1.000_{\pm.000}$	$2.35_{\pm.22}$	$245_{\pm14}$	$0.766_{\pm.010}$
$\alpha = 1$	32	$0.979_{\pm.004}$	$1.000_{\pm.000}$	$33.3_{\pm3.9}$	$35.6_{\pm4.7}$	$0.249_{\pm.028}$	–	–	–	–	–	$0.983_{\pm.003}$	$0.999_{\pm.000}$	$3.58_{\pm.08}$	$324_{\pm32}$	$0.953_{\pm.022}$
CSFM	32	$0.962_{\pm.010}$	$0.999_{\pm.000}$	$1.96_{\pm.18}$	$0.931_{\pm.035}$	$0.071_{\pm.002}$	–	–	–	–	–	$0.990_{\pm.000}$	$1.000_{\pm.000}$	$1.70_{\pm.10}$	$44.5_{\pm19.3}$	$0.670_{\pm.012}$

Table 15: **CSFM ablations.** Component and steering-strength ablations on Ball and Solid Torus at $d = 16$, and Box and Solid Torus at $d = 32$ ($m = 3$). Each row changes one CSFM component; the default uses direct correspondences and $\alpha = 0.1$. Base and FM-RE rows as in Table 6.