SOLVING INVERSE PROBLEMS OF DISTRIBUTIONS WITH WASSERSTEIN FLOW MAPS

Anonymous authors
Paper under double-blind review

ABSTRACT

Many scientific datasets consist of distributions rather than individual points, such as point clouds, cell populations or spatial tissue neighborhoods. Recovering populations from partial or aggregate observations, and steering their generation toward desired properties, call for generative priors that can adapt to different measurement operators and guidance objectives. Wasserstein flow matching provides such priors, but obtaining terminal populations requires numerical integration, making repeated endpoint prediction for inference-time guidance computationally expensive. We introduce Wasserstein flow maps, a framework for learning finite-time transport maps between distribution-valued states, enabling direct prediction of terminal populations. We formulate these maps on Wasserstein space and derive semigroup, Eulerian, and Lagrangian characterizations. The resulting maps support deterministic reward lookahead and flow map trajectory tilting, allowing one pretrained prior per domain to address multiple inverse problems and guidance objectives without retraining the generator. Across ShapeNet, single-cell populations, and spatial transcriptomic niches, Wasserstein flow maps improve generation metrics over Wasserstein flow matching baselines using 8 rather than 100 function evaluations, and outperform inverse problems methods on inpainting, super-resolution, deconvolution, and guided generation. On 100 held-out patients, lookahead guidance reduces reconstruction MMD by 35% for deconvolution from mean single-cell RNAseq embeddings and masked MMD by 37% for cell-type inpainting, relative to task-specific fine-tuned Wasserstein flow matching. Code is available at anonymous repository.

1 INTRODUCTION

Recovering an entire population from a partial or aggregate measurement is a recurring scientific inverse problem. A tissue measurement may reveal only a small subset of cells (Liu et al., 2026), an aggregate profile may conceal cellular heterogeneity, and a partial point cloud may leave much of a shape unobserved (Chang et al., 2015). In each case, the goal is to reconstruct a population whose structure is plausible and whose measurements agree with the observations. Many populations can explain the same measurement, making reconstruction depend on a prior that captures plausible population structure. A reusable generative prior should thus support these different observation operators without being retrained for each.

Generative models over distributions provide a natural foundation: each sample is an unordered population $\mu \in \mathcal{P}_2(\mathcal{M})$, and the learned prior $\mathbb{P} \in \mathcal{P}_2(\mathcal{P}_2(\mathcal{M}))$ captures variation across populations. Wasserstein Flow Matching (WFM) learns such priors by evolving populations through a learned velocity field (Haviv et al., 2025; Piening et al., 2026). Using these priors for inverse problems requires steering generation toward agreement with the observations. This creates a computational bottleneck: the guidance objective is evaluated on the completed population, but obtaining it from a velocity model either requires numerical integration at every guidance step or relies on conditional mean of the endpoint which blurs the population over all compatible completions (Kim et al., 2025).

We introduce *Wasserstein flow maps*, which learn finite-time transport between population-valued states and directly predict terminal populations. These predictions allow an observation-dependent reward to be evaluated and differentiated from intermediate states without repeatedly integrating to the endpoint (Boffi et al., 2025; Davis et al., 2026; Woo et al., 2026). Changing the observation operator

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.

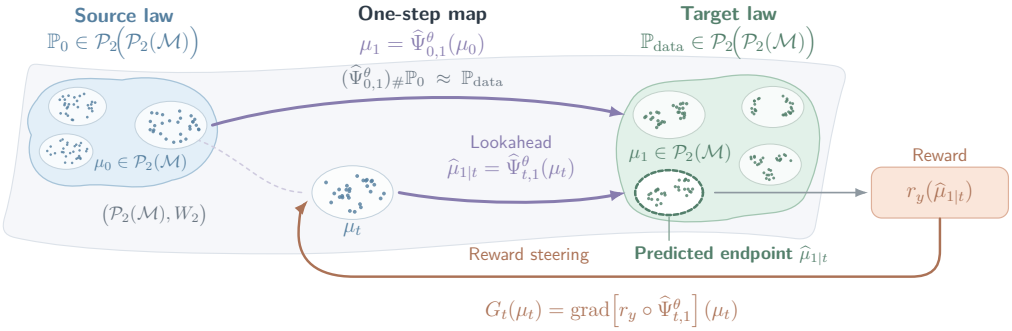


Figure 1: **Wasserstein flow maps for generation and inverse problems.** We model *distributions of distributions*: each sample is a population $\mu \in \mathcal{P}_2(\mathcal{M})$, while the source and target laws belong to $\mathcal{P}_2(\mathcal{P}_2(\mathcal{M}))$. We lift the flow-map paradigm to Wasserstein space by learning a finite-time population transport map $\widehat{\Psi}^{\theta}_{s,t} : \mathcal{P}_2(\mathcal{M}) \to \mathcal{P}_2(\mathcal{M})$. We train this map on source–data population paths using conditional velocity supervision and consistency objectives derived from the semigroup, Eulerian, or Lagrangian flow identities. The map enables one-step generation and direct prediction of terminal populations from intermediate states. Evaluating rewards on these predictions and differentiating through the map facilitates efficient reward tilting and inverse-problem solving with a frozen prior.

changes the guidance objective while keeping the pretrained prior fixed. Our framework connects flow maps on Wasserstein space to tractable learning objectives for finite, unordered populations of Euclidean or manifold-valued points. We derive semigroup, Eulerian, and Lagrangian characterizations and adapt deterministic reward lookahead and flow map trajectory tilting (Skreta et al., 2025; Sabour et al., 2025) that leverage these maps. The resulting framework supports population reconstruction with arbitrary observation operators and generation toward desired population properties.

Across ShapeNet, single-cell populations, and spatial tissue niches, Wasserstein flow maps improve on WFM across all four reported generation metrics using 8 rather than 100 function evaluations. The same pretrained priors support inpainting, super-resolution, deconvolution, and guided generation. On 100 held-out patients, lookahead guidance reconstructs cell populations from mean embeddings with 35% lower MMD than task-specific fine-tuned WFM, using 16 rather than 100 function evaluations. The same frozen prior reduces masked MMD by 37% for cell-type inpainting.

Contributions.

- **Wasserstein flow maps.** We formulate finite-time transport maps on $\mathcal{P}_2(\mathcal{M})$ and derive semigroup, Eulerian, and Lagrangian characterizations. We connect these characterizations to tractable training objectives for empirical populations of Euclidean or manifold-valued points. The result is a population prior on $\mathcal{P}_2(\mathcal{M})$ we can sample from in only a few function evaluations (Sec. 3).
- **Inverse problems and inference-time guidance for populations.** We formulate population-valued inverse problems as reward tilts of a generative prior. We adapt two inference-time methods to distribution-valued states: deterministic reward lookahead and a Riemannian Feynman-Kac corrector. We derive the corresponding Riemannian weight dynamics and identify the approximation introduced when the learned map is not exact (Sec. 4).
- **Evidence across geometries and observation types.** Across ShapeNet, single-cell data, and spatial transcriptomics, we demonstrate efficient generation and reuse of one frozen prior per domain across multiple inverse problems, improving upon existing approaches (Sec. 5).

2 Related Work

Generative modeling over distributions. Wasserstein Flow Matching (WFM) (Haviv et al., 2025) learns generative flows over families of distributions using optimal transport to construct conditional paths. RWEFM (Haviv et al., 2026) extends this construction to populations of manifold-valued points. Generalized WFM (Piening et al., 2026) couples transport between populations with transport between their constituent points. Meta Flow Matching (Atanackovic et al., 2025) learns population-dependent dynamics that generalize across initial populations. These approaches represent population evolution through instantaneous velocity fields. We learn finite-time solution operators for population-valued dynamics, providing direct terminal predictions for generation and inference-time guidance.

Flow maps and geometric extensions. Flow Map Matching (Boffi et al., 2025) provides a framework for learning two-time solution operators for few-step generation. Generalised Flow Maps (Davis et al., 2026) and Riemannian MeanFlow (Woo et al., 2026) extend flow-map learning to Riemannian manifolds. In particular, Riemannian MeanFlow develops semigroup, Eulerian, and Lagrangian characterizations of manifold average velocities and corresponding training objectives. Our finite-particle objectives build on these characterizations through permutation-equivariant maps on a product manifold. We connect this particle formulation to population-dependent flows on Wasserstein space, establishing how distribution-level identities yield tractable objectives on empirical populations.

Inverse problems over populations. Generative priors are widely used for image inverse problems such as inpainting, super-resolution, and deblurring (Lugmayr et al., 2022; Wang et al., 2023; Kawar et al., 2022; Chung et al., 2023). Inverse problems with distribution-valued unknowns have received less attention. Recent work has investigated inverse problem over distributions through variational formulations and Wasserstein gradient flows (Li et al., 2024; Sanchez Espinosa et al., 2025). Unlike our approach, these methods optimize the unknown distribution directly against observations, without relying on a pretrained generative prior over populations. Other works have explored 3D point cloud reconstruction from partial observations (Yu et al., 2021; Lyu et al., 2022). Point Set Diffusion (Lüdke et al., 2025) supports conditional generation of point sets from masked observations but do not generalize to arbitrary inverse problems, unlike our approach.

Inference-time guidance and correction. Training-free guidance incorporates differentiable rewards and data-consistency terms into pretrained generators (Bansal et al., 2024; Chung et al., 2023), with flow-based inverse solvers including FlowDPS (Kim et al., 2025) and PnP-Flow (Martin et al., 2025). Reward-guided generation can be formulated as sampling from a tilted prior, with value functions assessing terminal rewards from intermediate states (Uehara et al., 2025). Sequential Monte Carlo and Feynman-Kac methods correct guided proposals using importance weights (Wu et al., 2023; Skreta et al., 2025). Riemannian MeanFlow uses terminal map predictions for reward lookahead (Woo et al., 2026), and FMTT combines flow-map lookahead with trajectory weighting (Sabour et al., 2025). Meta Flow Maps and Diamond Maps learn stochastic maps for efficient value estimation (Potaptchik et al., 2026; Holderrieth et al., 2026). We adapt deterministic lookahead and trajectory tilting to population-valued states, deriving the associated Riemannian weight dynamics and identifying the approximation introduced by imperfect flow-map consistency.

3 Wasserstein Flow Maps

3.1 Flows and Flow Maps

Let $(\mathcal{M}, g)$ be a complete Riemannian manifold. We write $T_x\mathcal{M}$ for the tangent space at x, $\langle \cdot, \cdot \rangle_g$ and $\| \cdot \|_g$ for its inner product and norm, d_g for the geodesic distance, $\exp_x(\cdot)$ for the exponential map, and $\log_x(y)$ its local inverse on a normal neighborhood of x.

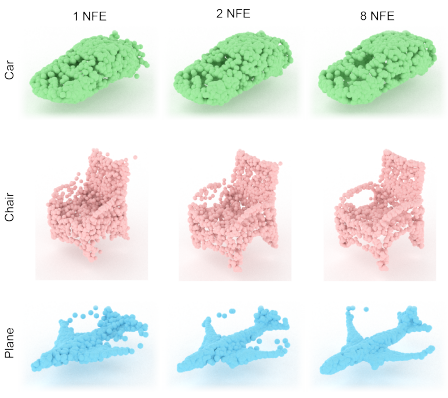


Figure 2: **Few-step generation on ShapeNet.** Point clouds produced by Wasserstein flow map with 1, 2, 8 step generation.

A time-dependent vector field $v_t(x) \in T_x\mathcal{M}$, $t \in [0, 1]$, generates trajectories $\dot{x}_t = v_t(x_t)$. When these dynamics are well-posed, their two-time solution operator, the flow map $\mathcal{X}_{s,t} : \mathcal{M} \to \mathcal{M}$, sends a state at time s to its state at time t and satisfies

$$\partial_t \mathcal{X}_{s,t}(x) = v_t(\mathcal{X}_{s,t}(x)), \qquad \mathcal{X}_{s,s}(x) = x. \tag{3.1}$$

For an initial distribution $q_s \in \mathcal{P}_2(\mathcal{M})$, where $\mathcal{P}_2(\mathcal{M})$ denotes the probability measures with finite second moment, the flow induces the probability path $q_t = (\mathcal{X}_{s,t})_{\#} q_s$. Generative modeling techniques such as flow matching and diffusion models leverage that connection to transport samples from a reference distribution q_0 to a target distribution q_1 by learning the corresponding vector field. However, that procedure requires costly numerical integration at inference time. Instead, flow map models directly approximate $\mathcal{X}_{s,t}$, enabling finite-time endpoint prediction without integrating the velocity field (Boffi et al., 2025; Davis et al., 2026; Woo et al., 2026).

Table 1: **Unconditional generation.** 1-NN-D (Eq. F.1) and cloud-kernel MMD (Eq. F.2) between generated and held-out populations, each under the Chamfer (CD) and earth-mover (EMD) base distances; lower is better for all four. Entries are mean±standard deviation over 5 generation seeds. Within each domain the three best entries per column are shaded (1st , 2nd , 3rd). App. H.4 reports all methods across NFE budgets.

Domain	Method	1-NN-D (CD) ↓	1-NN-D (EMD) ↓	MMD (CD) ↓	MMD (EMD) ↓	NFE
3D point cloud (ShapeNet)	PVD	0.113±0.020	0.108±0.019	1.35e-02±1e-03	7.41e-03±1e-03	1000
	PSF	0.121±0.019	0.110±0.013	1.79e-02±2e-03	1.01e-02±1e-03	1000
	FM	0.496±0.001	0.495±0.001	1.46e-01±3e-04	1.15e-01±9e-04	100
	SetFM	0.316±0.005	0.273±0.008	6.26e-02±4e-03	3.58e-02±2e-03	100
	WFM	0.212±0.010	0.178±0.011	5.98e-02±2e-03	3.18e-02±1e-03	100
	Flow map: Semigroup	0.213±0.006	0.187±0.009	1.85e-02±2e-03	8.68e-03±1e-03	8
	Flow map: Eulerian	0.249±0.010	0.181±0.009	1.71e-02±2e-03	9.30e-03±1e-03	8
	Flow map: Lagrangian	0.130±0.009	0.105±0.010	1.27e-02±1e-03	6.79e-03±1e-03	8
scRNA-seq	RFM	0.453±0.002	0.345±0.004	6.07e-03±1e-05	7.86e-04±5e-06	100
	SetRFM	0.203±0.006	0.072±0.014	8.64e-04±5e-05	1.34e-04±2e-05	100
	RWEFM	0.284±0.013	0.101±0.010	9.88e-04±3e-05	1.89e-04±2e-05	100
	Flow map: Semigroup	0.138±0.010	0.069±0.012	2.26e-04±6e-06	2.24e-05±2e-06	8
	Flow map: Eulerian	0.297±0.011	0.195±0.024	1.36e-03±1e-05	1.49e-04±6e-06	8
	Flow map: Lagrangian	0.150±0.010	0.110±0.012	2.04e-04±1e-05	2.40e-05±2e-06	8
Spatial transcriptomics	FM	0.500±0.000	0.497±0.000	8.80e-04±1e-05	4.18e-03±4e-05	100
	SetFM	0.338±0.020	0.198±0.041	6.36e-05±2e-06	9.52e-05±5e-06	100
	WFM	0.329±0.010	0.192±0.018	2.57e-04±2e-05	3.85e-04±4e-05	100
	Flow map: Semigroup	0.327±0.019	0.194±0.017	1.16e-04±5e-06	2.80e-04±1e-05	8
	Flow map: Eulerian	0.366±0.015	0.217±0.019	9.45e-05±6e-06	1.55e-04±2e-05	8
	Flow map: Lagrangian	0.301±0.027	0.168±0.013	9.90e-05±5e-06	2.35e-04±2e-05	8

Direct endpoint prediction is particularly useful for inference-time guidance, which repeatedly evaluates rewards on predicted terminal states. A learned approximation to $\mathcal{X}_{t,1}$ allows these rewards to be evaluated and differentiated with respect to intermediate states without repeatedly integrating to terminal time. Exact flow maps satisfy consistency under composition (semigroup), a forward evolution equation (Lagrangian), and a backward equation expressing endpoint invariance along an input trajectory (Eulerian) (Boffi et al., 2025). These identities provide the basis for practical flow-map training objectives. We next extend this framework to distribution-valued states.

3.2 Generative modeling on distributions

Remarkably, flow matching can be adapted to population-valued data by taking each sample to be a probability measure $\mu \in \mathcal{P}_2(\mathcal{M})$ and equipping this space with the 2-Wasserstein distance W_2 induced by d_g (Haviv et al., 2025; Atanackovic et al., 2025; Piening et al., 2026). The generative task is then to transport a reference law $\mathbb{P}_0$ over populations to a data law $\mathbb{P}_1$, with $\mathbb{P}_0, \mathbb{P}_1 \in \mathcal{P}_2(\mathcal{P}_2(\mathcal{M}))$. We describe population dynamics through a measure-conditioned velocity field $v_t(x, \mu) \in T_x\mathcal{M}$. For a fixed population μ, $v_t(\cdot, \mu)$ is a vector field on $\mathcal{M}$; its dependence on μ allows the velocity of each point to depend on the entire population. Starting from μ_s, the population trajectory μ_t evolves according to the continuity equation:

$$\partial_t \mu_t + \mathrm{div}_g\left(\mu_t v_t(\cdot, \mu_t)\right) = 0. \tag{3.2}$$

The equation is understood in the weak sense, allowing μ_t to be an empirical measure. Under standard regularity assumptions (App. C.1), Equation 3.2 has a unique solution for each initial population $\mu_s \in \mathcal{P}_2(\mathcal{M})$ and defines a distribution-valued flow map

$$\Psi_{s,t} : \mathcal{P}_2(\mathcal{M}) \to \mathcal{P}_2(\mathcal{M}), \qquad \Psi_{s,t}(\mu_s) := \mu_t. \tag{3.3}$$

Unlike Equation 3.1, $\Psi_{s,t}$ transports probability distributions and its pushforward transports laws over populations $\mathbb{P}_t = (\Psi_{s,t})_{\#}\mathbb{P}_s$.

3.3 Flow maps on the Wasserstein space

Equation 3.2 specifies the instantaneous population dynamics. Rather than numerically integrating these dynamics, we seek to learn their finite-time solution operator $\Psi_{s,t}$ directly. The exact operator satisfies three complementary identities: consistency under composition, forward evolution, and endpoint invariance along an input trajectory. Throughout, $\Psi_{t,t} = \mathrm{Id}$, and the differential identities are understood in the weak sense.

Semigroup. For any intermediate time $s \le r \le t$, we have $\Psi_{s,t} = \Psi_{r,t} \circ \Psi_{s,r}$. Thus, transporting a population directly from s to t agrees with transporting it through any intermediate time r. To identify the flow associated with the prescribed velocity field, its infinitesimal behavior on the diagonal must additionally agree with Equation 3.2.

Lagrangian (forward evolution). Fixing an initial population μ_s at time s, the continuity equation in Equation 3.2 readily yields the forward derivative:

$$\partial_t \Psi_{s,t}(\mu_s) + \mathrm{div}_g\left(\Psi_{s,t}(\mu_s) v_t(\cdot, \Psi_{s,t}(\mu_s))\right) = 0, \tag{3.4}$$

Eulerian (endpoint invariance). Fixing the terminal time t instead, and letting μ_r itself evolve according to Equation 3.2, moving the starting point forward along the same trajectory while shortening the remaining flow should leave the terminal population unchanged:

$$\frac{d}{dr} \int_{\mathcal{M}} \varphi(x)\, d\Psi_{r,t}(\mu_r)(x) = 0, \qquad r \in [s,t], \tag{3.5}$$

for every $\varphi \in C_c^\infty(\mathcal{M})$. Equivalently, $\Psi_{r,t}(\mu_r)$ remains constant along the input trajectory. These weak formulations apply to both continuous and empirical populations. We next specialize the identities to empirical measures and derive particle-level residuals for training.

3.4 Wasserstein flow maps on empirical distributions

The above distribution-level identities also apply to empirical populations. We represent a population by $X = (x_1, \dots, x_N) \in \mathcal{M}^N$, with empirical measure $\mu_X = \mathrm{emp}_N(X) := N^{-1} \sum_i \delta_{x_i}$.

Empirical parametrization. We identify our flow map estimator $\widehat{\Psi}_{s,t}$ with a shared tangent field $u_\theta(s,t,x,\mu) \in T_x\mathcal{M}$ parametrized by a neural network. Given the tangent field, the flow map estimator is computed using

$$T^\theta_{s,t}[\mu](x) = \exp_x\big((t-s) u_\theta(s,t,x,\mu)\big), \qquad \widehat{\Psi}^\theta_{s,t}(\mu) = \big(T^\theta_{s,t}[\mu]\big)_{\#}\mu. \tag{3.6}$$

This parametrization defines a map on $\mathcal{P}_2(\mathcal{M})$ whose empirical realization is exactly

$$\Phi^{N,\theta}_{s,t,i}(X) = T^\theta_{s,t}[\mu_X](x_i), \qquad \widehat{\Psi}^\theta_{s,t}(\mu_X) = \mu_{\Phi^{N,\theta}_{s,t}(X)}. \tag{3.7}$$

Practically, $u_\theta(s,t,x,\mu_X)$ is a permutation equivariant network that transport each point x in the population from s to t. This parameterization enforces $\Phi^{N,\theta}_{s,s} = \mathrm{Id}$, while the flow identities provide additional constraints for learning. Below, we suppress N and θ when discussing a candidate particle map $\Phi_{s,t}$, and write $X_t = \Phi_{s,t}(X_s)$ for its predicted configuration.

From distribution identities to particle residuals. For a test function $\varphi \in C_c^\infty(\mathcal{M})$, evaluating the predicted population gives $F_\varphi(\mu_{X_t}) = N^{-1} \sum_i \varphi(x_{i,t})$. Evaluating Equations 3.4 and 3.5 in the weak sense yields

$$\frac{1}{N} \sum_{i=1}^N \big\langle \nabla_g \varphi(x_{i,t}), R_{\alpha,i}(s,t,X_s) \big\rangle_g = 0, \qquad \alpha \in \{\mathrm{L}, \mathrm{E}\}, \tag{3.8}$$

for every $\varphi \in C_c^\infty(\mathcal{M})$, where the Lagrangian ($\alpha = L$) and Eulerian ($\alpha = E$) residuals are given by

$$R_{\mathrm{L},i}(s,t,X_s) := \partial_t \Phi_{s,t,i}(X_s) - v_t(x_{i,t}, \mu_{X_t}), \tag{3.9}$$

$$R_{\mathrm{E},i}(s,t,X_s) := \partial_s \Phi_{s,t,i}(X_s) + \sum_{j=1}^N d_{x_j} \Phi_{s,t,i}(X_s)\big[v_s(x_{j,s}, \mu_{X_s})\big]. \tag{3.10}$$

Complete derivations are provided in App. C.2.

Residuals in average-velocity coordinates. Minimizing the above residuals corresponds to enforcing the Lagrangian and Eulerian conditions of Equations 3.4 and 3.5. However, $u_\theta(s,t,x,\mu_X)$ emits tangent vectors, whereas R_{L} and R_{E} are based on predicted endpoints. Writing $U_{s,t,i}(X_s) := u_\theta(s,t,x_{i,s},\mu_{X_s})$ and $V_{s,i}(X_s) := v_s(x_{i,s},\mu_{X_s})$, we define the material derivative

$$\mathcal{D}^{V_s}_s U_{s,t}(X_s) := \partial_s U_{s,t}(X_s) + \nabla^{\mathcal{M}^N}_{V_s(X_s)} U_{s,t}(X_s), \tag{3.11}$$

where $\nabla^{\mathcal{M}^N}_{V_s(X_s)} U_{s,t}(X_s)$ is the covariant derivative of $U_{s,t}$ along V_s. The following proposition collects the semigroup, Lagrangian, and Eulerian conditions for flow maps generated by v_t.

Proposition 3.1 (Flow-map characterizations). *Let $X_t = \Phi_{s,t}(X_s)$, with smooth average velocity $U_{s,t}$. Where the logarithm smoothly inverts the parameterization, the flow-map conditions associated with v_t are*

$$\textbf{(Semigroup)} \quad \Phi_{s,t} = \Phi_{r,t} \circ \Phi_{s,r}, \qquad s \le r \le t, \tag{3.12}$$

$$\textbf{(Lagrangian)} \quad U_{s,t,i}(X_s) = d_2 \log_{x_{i,s}}(x_{i,t})\big[v_t(x_{i,t}, \mu_{X_t})\big] - (t-s)\partial_t U_{s,t,i}(X_s), \tag{3.13}$$

$$\textbf{(Eulerian)} \quad U_{s,t,i}(X_s) = (t-s)\big(\mathcal{D}_s^{V_s} U_{s,t}(X_s)\big)_i - \nabla^1_{v_s(x_{i,s},\mu_{X_s})} \log_{x_{i,s}}(x_{i,t}). \tag{3.14}$$

Here $d_2 \log_x(y)$ differentiates the endpoint y with x fixed, while $\nabla^1_a \log_x(y)$ differentiates the base point x covariantly along a, with y fixed. The semigroup condition additionally requires $U_{t,t} = V_t$ to identify the prescribed flow, which is implied by both differential conditions.

Subtracting each right-hand side from $U_{s,t,i}(X_s)$ defines the corresponding average-velocity residual $r_{\mathrm{L},i}, r_{\mathrm{E},i} \in T_{x_{i,s}}\mathcal{M}$. The proof is given in App. C.3.

Empirical objectives. The characterizations of Proposition 3.1 yield the ideal consistency penalties

$$\mathcal{C}_{\mathrm{SG}} = \mathbb{E}\left[\frac{1}{N}\sum_i d_g^2(x_{i,t}, \widetilde{x}_{i,t})\right], \; \mathcal{C}_{\mathrm{E}} = \mathbb{E}\left[\frac{1}{N}\sum_i \|r_{\mathrm{E},i}\|_g^2\right], \; \mathcal{C}_{\mathrm{L}} = \mathbb{E}\left[\frac{1}{N}\sum_i \|r_{\mathrm{L},i}\|_g^2\right], \tag{3.15}$$

where $\widetilde{X}_t = \Phi_{r,t}(\Phi_{s,r}(X_s))$. The expectations average over input populations from conditional training paths and sampled time pairs or triples, as specified in App. D.

In practice, we construct regression objectives with stopped targets using conditional velocities satisfying $V_t(X) = \mathbb{E}[V_t^{\mathrm{cond}} \mid X_t^{\mathrm{cond}} = X]$. Eulerian training uses the conditional velocity at the sampled input. Lagrangian training instead samples X_t^{cond}, predicts an earlier state $\widehat{X}_s = \Phi_{t,s}(X_t^{\mathrm{cond}})$, and uses an endpoint-anchored surrogate with cycle regularization. Semigroup training includes diagonal velocity supervision, which we also retain as a stabilizer for Lagrangian training. App. D specifies the conditional paths, parameterization, time sampling, and complete training losses.

4 Inverse Problems and Guidance on the Wasserstein Space

4.1 Inverse Problems in Distribution Space

Let $\mathbb{P}_{\mathrm{prior}} \in \mathcal{P}(\mathcal{P}_2(\mathcal{M}))$ denote the prior law over populations induced by the pretrained generative model. An inverse problem starts from an unknown target population $\mu^\star$ and a known observation operator $\mathcal{O} : \mathcal{P}_2(\mathcal{M}) \to \mathcal{Y}$. Examples of observation operators include cropping (*i.e.,* discarding all the probability mass in some region of $\mathcal{M}$) or aggregating (*i.e.,* taking $\mathcal{O}(\mu^\star) = \int \phi(x) d\mu^\star(x)$ for some function ϕ), among others.

Given the observation $y = \mathcal{O}(\mu^\star)$, the objective is to recover populations that are both plausible under $\mathbb{P}_{\mathrm{prior}}$ and consistent with y. We encode compatibility with the observation through a reward $r_y : \mathcal{P}_2(\mathcal{M}) \to \mathbb{R}$ and define the corresponding observation-informed target law

$$d\mathbb{P}_y(\mu) = \frac{\exp(r_y(\mu))}{Z_y}\, d\mathbb{P}_{\mathrm{prior}}(\mu), \qquad Z_y = \int_{\mathcal{P}_2(\mathcal{M})} \exp(r_y(\mu))\, d\mathbb{P}_{\mathrm{prior}}(\mu). \tag{4.1}$$

For our inverse tasks, we use $r_y(\mu) = -\lambda_y \ell_y(\mu)$, where $\ell_y : \mathcal{P}_2(\mathcal{M}) \to [0, \infty)$ measures observation mismatch and $\lambda_y \ge 0$ controls the strength of the penalty. This defines a soft consistency target: finite λ_y favors compatible populations without generally enforcing the observation exactly. When a probabilistic observation model is available, choosing $r_y(\mu) = \log p(y \mid \mu)$, instead recovers the Bayesian posterior under $\mathbb{P}_{\mathrm{prior}}$. Task-specific observations and rewards are given in App. G.2, G.3.

This formulation places population inverse problems within reward-guided generation. At particle count N, we evaluate the reward on the empirical population through $r_y^N(X) := r_y(\mu_X)$. Tilting the particle law by this reward pushing it forward through emp_N give exactly the corresponding tilted law over empirical populations (App. E.1).

Table 2: **Guiding unconditional priors for inverse problems.** Inference-time guidance on ShapeNet inpainting, MERFISH niche super-resolution, and bulk-to-single-cell deconvolution. FlowDPS and PnP-Flow run on the frozen WFM (RWEFM on scRNA-seq); Lookahead and FMTT run on the frozen flow map of Sec 5.1 (Lagrangian on ShapeNet and MERFISH, Semigroup on scRNA-seq); FT-WFM is the WFM fine-tuned per task and serves as a reference. ShapeNet numbers are averaged over the three categories of Table 7. Bold marks the best result among the frozen-prior methods.

Dataset		ShapeNet				MERFISH Spatial Niche				scRNA-seq		
Task		Inpainting				Super Resolution				Bulk-to-single-cell deconvolution		
		Blob (50 %)		Planar (50 %)		$M = 8$ (6.25%)		$M = 32$ (25%)				
Prior	Method	MMD↓	CD↓	MMD↓	CD↓	MMD↓	CD↓	MMD↓	CD↓	MMD↓	Pearson ↑	ΔARI↓
WFM	FT-WFM (task-specific)	0.181	0.029	0.171	0.032	0.167	0.351	0.120	0.354	0.093	0.936	0.013
	FlowDPS	0.266	0.044	0.267	0.050	0.269	0.548	0.209	0.476	0.076	0.948	0.021
	PnP-Flow	0.314	0.054	0.309	0.059	0.266	0.512	0.169	0.432	0.305	0.447	0.209
Flow map	Lookahead	**0.200**	**0.029**	0.227	**0.035**	**0.225**	0.455	**0.142**	**0.393**	**0.060**	**0.964**	**0.001**
	FMTT	0.204	0.034	**0.221**	0.040	0.249	**0.395**	0.209	0.402	0.073	0.952	0.017

4.2 Inference-time guidance and correction

We consider two complementary strategies to obtain samples from the tilted distribution $\mathbb{P}_y$: deterministic lookahead guidance and flow map trajectory tilting, which uses Feynman-Kac steering. Both use our Wasserstein flow maps to obtain fast and accurate samples from the reference distribution.

Deterministic map lookahead. In value-based guidance, the terminal tilt is propagated to intermediate states through the soft value function $h_t^y(X_t) = \log \mathbb{E}[e^{r_y^N(X_1)} \mid X_t]$, whose gradient provides a local direction for steering the generative model. For stochastic reference dynamics, this value determines an exact conditioned drift together with a reweighted initial law (App. E.2). In our case, the flow map predicts the deterministic terminal particle state $\widehat{X}_{1|t} := \Phi_{t,1}^\theta(X_t)$. This gives the lookahead approximation $\widehat{h}_t^y(X_t) := r_y^N(\widehat{X}_{1|t})$. We then steer the generation process by adding the reward gradient $G_t(X_t)$ to the reference velocity field $V_t(X_t)$ with

$$G_t(X_t) := \operatorname{grad} \widehat{h}_t^y(X_t) = \left(d\Phi_{t,1}^\theta(X_t)\right)^* \operatorname{grad} r_y^N\left(\widehat{X}_{1|t}\right) \tag{4.2}$$

Each guidance evaluation only requires a terminal flow map query and a vector-Jacobian product rather than integrating the reference dynamics to terminal time.

Flow map trajectory tilting (FMTT). Deterministic lookahead provides an efficient proxy for steering toward the tilted target, but does not in general sample from the prescribed tilted law $\mathbb{P}_y$. One can instead introduce at inference time an auxiliary stochastic process sharing the same reference particle marginals $\mathbb{P}_t$ as the learned flow, and apply a Feynman–Kac correction (Skreta et al., 2025) on the Riemannian particle lift $\mathcal{M}^N$. FMTT (Sabour et al., 2025) is an instantiation of Feynman-Kac steering with a particular choice of potential u_t that leverages the flow map $\Phi_{t,1}^\theta$ to evaluate the reward on a clean endpoint:

$$u_t^{\mathrm{FMTT}}(X_t) := \chi_t \, r_y^N\left(\Phi_{t,1}^\theta(X_t)\right), \tag{4.3}$$

where $\chi : [0,1] \to [0,1]$ is a scheduler with $\chi_0 = 0, \chi_1 = 1$. We derive the Riemannian Feynman-Kac dynamics and importance-weight updates in App. E.3. For FMTT, the full log-weight rate includes a correction term involving the residual of the backward (Eulerian) flow identity. This term vanishes for an exact flow map; our implementation omits it for the learned map. App. E.4 derives the full and simplified rates and bounds the resulting log-weight-rate error.

5 Experiments

We evaluate our approach in three different configurations: unconditional generation to assess the quality of the learned flow maps (Sec. 5.1), inverse problems on three different domains, and an example of general reward-guided generation on cell-type enrichment of spatial transcriptomics data (Sec. 5.2). Details about datasets, architectures, optimization and metrics are provided in App. F, and observation operators and rewards in App. G. Additional tasks and qualitative figures are in App. H and ablations in App. I.

Datasets. We consider three domains for our experiments: 3D point clouds from ShapeNet (Chang et al., 2015) ($N = 2048$ points in $\mathbb{R}^3$), patient-level single-cell RNA sequencing profiles (CZI Cell Science Program et al., 2025) ($N = 1024$ cells on $\mathbb{S}^{127}$), and MERFISH spatial niches (Zhang et al., 2021) ($N = 128$ cells in $\mathbb{R}^{16}$). Each population is one training sample; the train/validation/test splits contain 9,902/1,419/2,829 shapes, 20,953/1,047/1,047 patients, and 15,830/1,024/1,403 niches.

Baselines. We compare our Wasserstein flow maps trained with the Eulerian, Lagrangian and semigroup objectives (Sec. 3.4) against a wide range of generative baselines: WFM (Haviv et al., 2025) and its extension to Riemannian manifolds RWEFM (Haviv et al., 2026); standard and Riemannian flow matching (FM, RFM) (Lipman et al., 2023; Chen & Lipman, 2024) which operate per sample and ignore the population structure; set-encoder variants (SetFM,SetRFM) which captures particle interactions but do not leverage Wasserstein geometry; and 3D point cloud specific generative models such as PVD (Zhou et al., 2021) and PSF (Wu et al., 2022). For inference problems, we compare Wasserstein flow maps with lookahead and FMTT guidance against FT-WFM, a WFM model fine-tuned on each specific inverse problem task (FT-WFM, App. F.4); PnP-Flow (Martin et al., 2025) and FlowDPS (Kim et al., 2025), inverse problems methods applied on top of WFM.

Metrics. We evaluate unconditional generation using maximum mean discrepancy (MMD) over populations, with Chamfer (CD) and earth-mover (EMD) distances and classwise nearest-neighbor deviation (1-NN-D), which measures how far nearest-neighbor classification of real versus generated populations deviates from chance (lower is better). Inverse tasks use point-level MMD and CD against ground truth. Full definitions and task-specific diagnostics are in App. F.2.

5.1 Unconditional generation

Table 1 shows that Wasserstein flow maps match or improve on WFM/RWEFM across all four generation metrics in each domain while using 8 rather than 100 integration steps. The Lagrangian map achieves these improvements on ShapeNet and spatial niches, and the Semigroup map does so on single-cell data. For each domain we therefore fix the objective with the best generation metrics in Table 1 as the frozen prior for every inverse task in Sec. 5.2 and App. H. Figure 2 illustrates ShapeNet generation across step counts and App. H.4 provides evaluation-budget sweeps for all baselines. To further evaluate the benefits of Wasserstein flow maps, we compare direct map updates with numerical integration of its own velocity field given at the diagonal $s = t$ (App. I.1). Across all three domains, integration with 100 evaluations approaches the generation quality of the map with 8 evaluations, confirming the ability of the learned map to reproduce the generation quality of its associated flow with substantially fewer evaluations.

5.2 Inverse problems

ShapeNet inpainting We evaluate inpainting on the airplane, car, and chair categories of the ShapeNet dataset. We remove 50% of each point cloud using either a *blob* mask, which removes a contiguous region around randomly selected seeds, or a *planar* mask, which removes one side of an axis-aligned plane. We measure reconstruction quality in the missing region using MMD and CD, averaged across categories; the evaluation protocol is defined in App. F.2. Both lookahead and FMTT guidance on the Wasserstein flow map achieve lower MMD and CD than FlowDPS and PnP-Flow on both mask types (Table 2, left). FT-WFM provides a strong task-specific baseline but requires separate fine-tuning for each inpainting task, limiting its practicality when tasks or observation operators change. Using a single frozen prior, lookahead matches its Chamfer distance on the blob mask without any task-specific retraining. Figure 3 shows examples of reconstructions. App. H.1 provides additional qualitative comparisons and results on super-resolution and voxel deconvolution.

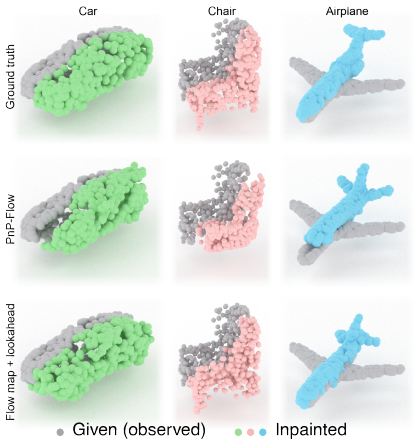


Figure 3: **ShapeNet inpainting** Ground truth and reconstructions by PnP-Flow on WFM and lookahead on flow map with 50% of the points removed.

Spatial transcriptomics niche super-resolution We evaluate population super-resolution on spatial transcriptomics niches from a MERFISH mouse primary motor cortex atlas. Each niche is a set of

128 cells neighboring a GABAergic neuron, represented by 16-dimensional PCA embeddings of gene-expression. We observe a uniformly sampled subset of $M = 8$ (6.25%) or $M = 32$ (25%) cells and reconstruct the remaining population (Table 2, middle). Lookahead guidance achieves better performance than FlowDPS and PnP-Flow at both observation densities. FT-WFM achieves the lowest errors overall but requires task-specific fine-tuning, whereas our methods reuse the same frozen flow map at both observation densities. App. H.3 provides qualitative reconstructions.

Single-cell deconvolution Bulk RNA sequencing measures aggregate expression across a cell population. Inferring the underlying cell-type composition and expression profiles from bulk measurements is an established yet challenging task in computational biology (Nguyen et al., 2024). Motivated by this problem, we study population reconstruction from aggregate observations in embedding space. Each patient is represented by 1024 unit-normalized SCimilarity embeddings (Heimberg et al., 2025) from the census dataset (CZI Cell Science Program et al., 2025), and the task is to reconstruct the population given only its mean embedding. We evaluate on 100 held-out patients using population MMD, Pearson correlation of cell-type compositions, and the absolute deviation from the ground-truth clustering adjusted Rand index (ARI) (App. F.2). Lookahead guidance on the frozen flow map achieves the best results on all three metrics (Table 2, right). It reduces MMD by 35% relative to FT-WFM (0.060 versus 0.093), while improving composition correlation and clustering ARI agreement. These improvements require no retraining of the generator and use 16 rather than 100 function evaluations. Figure 4 shows an example reconstruction. We additionally compare with non-negative least squares deconvolution (NNLS) (Lawson & Hanson, 1995) and DDNM (Wang et al., 2023) applied to the flow map. Both yield higher reconstruction MMD (0.164 and 0.192, respectively; not shown in Table 2), and poorer composition and clustering agreement than lookahead. App. H.2 reports cell-type inpainting and population super-resolution using the same pretrained prior.

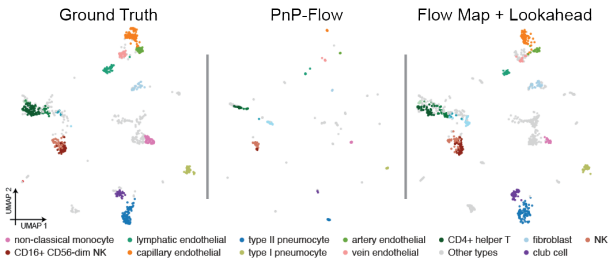


Figure 4: **scRNA-seq deconvolution.** True population, PnP-flow and flow-map (lookahead) for healthy lung tissue

Guided cell-type enrichment. We next guide generation toward a prescribed cell-type composition. Using the pretrained MERFISH niche model, we target the abundance of L2/3 IT cells through a composition reward (App. G.3). Targets come from held-out niches grouped into high, medium, and low abundance. Here, FMTT achieves better performance than lookahead in all three groups (Table 3). At high target abundance, it reduces the gap from 0.268 to 0.046, nearly matching FT-WFM (0.044). At medium and low abundance, FMTT outperforms FT-WFM without task-specific retraining. Enrichment is a reward-following task rather than a data-consistency task, where weighted correction most clearly outperforms steering.

Table 3: **Guided L2/3 IT enrichment on MERFISH niches.** Lookahead and FMTT steer the same frozen unconditional flow map toward requested cell-type abundances, with task-specific FT-WFM as a reference. Targets come from held-out niches with high, mid, and low abundances. Bold marks the best frozen-prior method.

Method	High ($p_{t^*} \approx 0.61$)			Mid (0.12)			Low (0.014)		
	Gap↓	1-NN-D↓ CD	1-NN-D↓ EMD	Gap↓	1-NN-D↓ CD	1-NN-D↓ EMD	Gap↓	1-NN-D↓ CD	1-NN-D↓ EMD
FT-WFM (task-specific)	0.044	0.280	0.270	0.078	0.500	0.487	0.032	0.450	0.425
Flow map: lookahead	0.268	0.485	0.425	0.095	0.450	0.369	0.049	0.330	0.285
Flow map: FMTT	**0.046**	**0.315**	**0.270**	**0.031**	**0.287**	**0.269**	**0.006**	**0.265**	**0.245**

6 Conclusions

We introduce Wasserstein flow maps for learning finite-time transport between population-valued states. Direct prediction of terminal populations makes the learned prior particularly useful for inference-time guidance, since downstream objectives can be evaluated and differentiated without repeatedly integrating the underlying flow. Across diverse inverse problems, flow map guidance improves over methods built on the corresponding frozen flow-matching priors and, in some settings, matches or outperforms task-specific fine-tuning. These results demonstrate the adaptability of Wasserstein flow maps to diverse observation operators and scientific objectives while reusing a single pretrained population prior.

AI USE STATEMENT

In this work, we used generative AI tools for experiment orchestration and for polishing the manuscript. Coding agents were used to clean and reformat datasets, write and run the scripts that launch, schedule and parallelize training runs, hyperparameter sweeps and evaluation jobs across GPUs, and collect/visualize their outputs. The models, training objectives, evaluation metrics and hyperparameter grids were specified by the authors; the tools executed sweeps over those grids and reported results. We did not use generative AI tools to develop the methodology or experiments, or interpret results. The remaining tasks in the required disclosure list (synthetic data generation, translation, qualitative or thematic analysis) are not applicable to this work. Additionally, we used generative AI tools to edit the paper for readability, grammar and consistency of notation, and assist for the writing of proofs. The authors are responsible for all content of this paper, including any produced with AI assistance.

REPRODUCIBILITY STATEMENT

App. D gives the exact training losses, the parameterization, time sampling, differential implementation and optimization setting (Tables 5 and 6); App. E gives the inference-time algorithms, observation operators, rewards and selected hyperparameters (Table 4); App. F.2 defines the evaluation metrics. Datasets are public (ShapeNet, the CZ CELLxGENE census with SCimilarity embeddings, and the MERFISH mouse primary motor cortex atlas) and their preprocessing is described in Sec. 5 and App. F. Source code and configuration files for all experiments will be released with the camera-ready version.

REFERENCES

Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient Flows in Metric Spaces and in the Space of Probability Measures*. Birkhäuser, 2 edition, 2008.

Lazar Atanackovic, Xi Zhang, Brandon Amos, Mathieu Blanchette, Leo J. Lee, Yoshua Bengio, Alexander Tong, and Kirill Neklyudov. Meta flow matching: Integrating vector fields on the wasserstein manifold. In *International Conference on Learning Representations*, 2025.

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *International Conference on Learning Representations*, 2024.

Nicholas M. Boffi, Michael S. Albergo, and Eric Vanden-Eijnden. Flow map matching with stochastic interpolants: A mathematical framework for consistency models. *Transactions on Machine Learning Research*, 2025.

Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

Ricky T. Q. Chen and Yaron Lipman. Flow matching on general geometries. In *International Conference on Learning Representations*, 2024.

Hyungjin Chung, Jeongsol Kim, Michael T. McCann, Michael L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations*, 2023.

CZI Cell Science Program, Shibla Abdulla, Brian Aevermann, Pedro Assis, Seve Badajoz, Sidney M Bell, Emanuele Bezzi, Batuhan Cakir, Jim Chaffer, Signe Chambers, J Michael Cherry, Tiffany Chi, Jennifer Chien, Leah Dorman, Pablo Garcia-Nieto, Nayib Gloria, Mim Hastie, Daniel Hegeman, Jason Hilton, Timmy Huang, Amanda Infeld, Ana-Maria Istrate, Ivana Jelic, Kuni Katsuya, Yang Joon Kim, Karen Liang, Mike Lin, Maximilian Lombardo, Bailey Marshall, Bruce Martin, Fran McDade, Colin Megill, Nikhil Patel, Alexander Predeus, Brian Raymor, Behnam Robatmili, Dave Rogers, Erica Rutherford, Dana Sadgat, Andrew Shin, Corinn Small, Trent Smith, Prathap Sridharan, Alexander Tarashansky, Norbert Tavares, Harley Thomas, Andrew Tolopko, Meghan Urisko, Joyce Yan, Garabet Yeretssian, Jennifer Zamanian, Arathi Mani, Jonah Cool, and Ambrose Carr. Cz cellxgene discover: a single-cell data platform for scalable exploration, analysis and

modeling of aggregated data. *Nucleic Acids Research*, 53(D1):D886–D900, 01 2025. ISSN 1362-4962. doi: 10.1093/nar/gkae1142. URL https://doi.org/10.1093/nar/gkae1142.

Oscar Davis, Michael S. Albergo, Nicholas M. Boffi, Michael M. Bronstein, and Avishek Joey Bose. Generalised flow maps for few-step generative modelling on Riemannian manifolds. In *International Conference on Learning Representations*, 2026.

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. 2406–2422, 2022.

Nicola Gigli. On the differential structure of metric measure spaces and applications. *Memoirs of the American Mathematical Society*, 236(1113):1–91, 2015.

Doron Haviv, Aram-Alexandre Pooladian, Dana Pe'Er, and Brandon Amos. Wasserstein flow matching: Generative modeling over families of distributions. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 22238–22258. PMLR, 2025.

Doron Haviv, Edward De Brouwer, Rishabh Anand, Rex Ying, Aïcha Bentaieb, Gabriele Scalia, and Hector Corrada Bravo. When riemann flows with wasserstein: Generative modeling of probability distributions on manifolds. 2026. URL https://arxiv.org/abs/2609.25659.

Graham Heimberg, Tony Kuo, Daryle J. DePianto, Omar Salem, Tobias Heigl, Nathaniel Diamant, Gabriele Scalia, Tommaso Biancalani, Shannon J. Turley, Jason R. Rock, Héctor Corrada Bravo, Josh Kaminker, Jason A. Vander Heiden, and Aviv Regev. A cell atlas foundation model for scalable search of similar human cells. *Nature*, 638(8052):1085–1094, 2025. doi: 10.1038/s41586-024-08411-y. URL https://doi.org/10.1038/s41586-024-08411-y.

Peter Holderrieth, Douglas Chen, Luca Eyring, Ishin Shah, Giri Anantharaman, Yutong He, Zeynep Akata, Tommi Jaakkola, Nicholas Matthew Boffi, and Max Simchowitz. Diamond maps: Efficient reward alignment via stochastic flow maps. *arXiv preprint arXiv:2602.05993*, 2026.

Elton P. Hsu. *Stochastic Analysis on Manifolds*, volume 38 of *Graduate Studies in Mathematics*. American Mathematical Society, 2002.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6:695–709, 2005.

Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Jeongsol Kim, Bryan Sangwoo Kim, and Jong Chul Ye. FlowDPS: Flow-driven posterior sampling for inverse problems. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12328–12337, 2025.

Charles L. Lawson and Richard J. Hanson. *Solving Least Squares Problems*, volume 15. SIAM, Philadelphia, PA, 1995. ISBN 978-0-89871-356-5. doi: 10.1137/1.9781611971217.

Qin Li, Maria Oprea, Li Wang, and Yunan Yang. Stochastic inverse problem: Stability, regularization and wasserstein gradient flow. *arXiv preprint arXiv:2410.00229*, 2024.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Tianyu Liu, Edward De Brouwer, Archit Verma, Alsu Missarova, Tony Kuo, Nathaniel Diamant, Hanchen Wang, Minsheng Hao, Tommaso Biancalani, Hector Corrada Bravo, et al. Learning multicellular representations of single-cell transcriptomics data enables characterization of patient-level disease states. *Cell Systems*, 17(5), 2026.

John Lott. On tangent cones in wasserstein space. *Proceedings of the American Mathematical Society*, 145:3127–3136, 2017.

David Lüdke, Enric Rabasseda Raventós, Marcel Kollovieh, and Stephan Günnemann. Unlocking point processes through point set diffusion. In *International Conference on Learning Representations*, 2025.

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. RePaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11461–11471, 2022.

Zhaoyang Lyu, Zhifeng Kong, Xudong Xu, Liang Pan, and Dahua Lin. A conditional point diffusion-refinement paradigm for 3d point cloud completion. In *International Conference on Learning Representations*, 2022.

Ségolène Martin, Anne Gagneux, Paul Hagemann, and Gabriele Steidl. PnP-Flow: Plug-and-play image restoration with flow matching. In *International Conference on Learning Representations*, 2025.

Robert J. McCann. Polar factorization of maps on riemannian manifolds. *Geometric and Functional Analysis*, 11(3):589–608, 2001.

Hung Nguyen, Ha Nguyen, Duc Tran, Sorin Draghici, and Tin Nguyen. Fourteen years of cellular deconvolution: methodology, applications, technical evaluation and outstanding challenges. *Nucleic Acids Research*, 52(9):4761–4783, 2024.

Felix Otto. The geometry of dissipative evolution equations: The porous medium equation. *Communications in Partial Differential Equations*, 26(1–2):101–174, 2001.

Benedetto Piccoli and Francesco Rossi. Transport equation with nonlocal velocity in wasserstein spaces: Convergence of numerical schemes. *Acta Applicandae Mathematicae*, 124:73–105, 2013.

Moritz Piening, Richard Duong, and Gabriele Steidl. Generalized wasserstein flow matching: Transport plans, everywhere, all at once. *arXiv preprint arXiv:2605.08424*, 2026.

Peter Potaptchik, Adhi Saravanan, Abbas Mammadov, Alvaro Prat, Michael S. Albergo, and Yee Whye Teh. Meta flow maps enable scalable reward alignment. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*. PMLR, 2026.

Amirmojtaba Sabour, Michael S. Albergo, Carles Domingo-Enrich, Nicholas M. Boffi, Sanja Fidler, Karsten Kreis, and Eric Vanden-Eijnden. Test-time scaling of diffusions with flow maps. *arXiv preprint arXiv:2511.22688*, 2025.

Diego Sanchez Espinosa, Erik H. Thiede, and Yunan Yang. Cryo-em as a stochastic inverse problem. *arXiv preprint arXiv:2509.05541*, 2025.

Marta Skreta, Tara Akhound-Sadegh, Viktor Ohanesian, Roberto Bondesan, Alan Aspuru-Guzik, Arnaud Doucet, Rob Brekelmans, Alexander Tong, and Kirill Neklyudov. Feynman–kac correctors in diffusion: Annealing, guidance, and product of experts. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 55906–55949. PMLR, 2025.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

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.

Masatoshi Uehara, Yulai Zhao, Chenyu Wang, Xiner Li, Aviv Regev, Sergey Levine, and Tommaso Biancalani. Inference-time alignment in diffusion models with reward-guided generation: Tutorial and review. *arXiv preprint arXiv:2501.09685*, 2025.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011.

Yinhuai Wang, Jiwen Yu, and Jian Zhang. Zero-shot image restoration using denoising diffusion null-space model. In *International Conference on Learning Representations*, 2023.

Dongyeop Woo, Marta Skreta, Seonghyun Park, Kirill Neklyudov, and Sungsoo Ahn. Riemannian meanflow. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*. PMLR, 2026.

Lemeng Wu, Dilin Wang, Chengyue Gong, Xingchao Liu, Yunyang Xiong, Rakesh Ranjan, Raghuraman Krishnamoorthi, Vikas Chandra, and Qiang Liu. Fast point cloud generation with straight flows, 2022. URL https://arxiv.org/abs/2212.01747.

Luhuan Wu, Brian L. Trippe, Christian A. Naesseth, David M. Blei, and John P. Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 31372–31403, 2023.

Jian Xu, Delu Zeng, John Paisley, and Qibin Zhao. What do flow-based inverse solvers approximate? a posterior-transport view. *arXiv preprint arXiv:2606.24516*, 2026.

Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows, 2019. URL https://arxiv.org/abs/1906.12320.

Xumin Yu, Yongming Rao, Ziyi Wang, Zuyan Liu, Jiwen Lu, and Jie Zhou. PoinTr: Diverse point cloud completion with geometry-aware transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12498–12507, 2021.

Meng Zhang, Stephen W Eichhorn, Brian Zingg, Zizhen Yao, Kaelan Cotter, Hongkui Zeng, Hongwei Dong, and Xiaowei Zhuang. Spatially resolved cell atlas of the mouse primary motor cortex by merfish. *Nature*, 598(7879):137–143, 2021.

Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion, 2021. URL https://arxiv.org/abs/2104.03670.

APPENDIX CONTENTS

A Notations and flow-map preliminaries **15**

A.1 Notation . . . 15

A.2 Flow matching and exact flow maps . . . 15

A.3 Riemannian finite-time average velocity . . . 16

B Finite-particle Wasserstein geometry **16**

B.1 Metric scaling and the empirical quotient . . . 16

B.2 Conditional paths and exchangeability . . . 17

B.3 The inner coupling and the scope of "Wasserstein" . . . 17

C Proofs for Wasserstein flow maps **17**

C.1 Existence and uniqueness of the distribution flow . . . 17

C.2 From distribution identities to empirical residuals . . . 18

C.3 From particle residuals to average-velocity characterizations . . . 19

D Training objectives and implementation **20**

D.1 Conditional training . . . 21

D.2 Losses . . . 21

D.3 Differential implementation . . . 21

D.4 Parameterization, time sampling and objective settings . . . 22

E Inference-time methods **23**

E.1 The inverse target on the particle lift . . . 23

E.2 Deterministic steering . . . 23

E.3 Feynman–Kac steering on the particle lift . . . 24

E.4 FMTT: the reference path, exact rates, and approximation . . . 26

E.5 Reference paths and score recovery . . . 27

F Experimental details **27**

F.1 Model architecture and optimization . . . 28

F.2 Metric definitions . . . 29

F.3 Cell-type conditioning and the composition inferrer. . . . 30

F.4 Task-specific fine-tuning (FT-WFM) . . . 31

G Inverse Problem Details **31**

G.1 Noiseless observation models and reward tilts . . . 31

G.2 ShapeNet inverse problems . . . 32

G.3 Single-cell and spatial-niche inverse problems . . . 32

G.4 Guidance settings and hyperparameters . . . 33

H Additional Experiments **33**
H.1 ShapeNet . . . 33
H.2 Single cell . . . 36
H.3 Spatial niche . . . 37
H.4 Generation quality against number of function evaluations . . . 39

I Ablations **40**
I.1 Sampling from the learned velocity field . . . 40
I.2 Cell generation without context . . . 40
I.3 Euclidean versus Riemannian single-cell flow maps . . . 43

A Notations and flow-map preliminaries

This appendix fixes notation and collects the standard point-valued flow-matching and flow-map facts that Sec. 3 lifts to distribution-valued states.

A.1 Notation

$(\mathcal{M}, g)$, d_g, $\exp_x$, $\log_x$	one-particle manifold, geodesic distance, exponential and logarithm maps
$\mathcal{P}_2(\mathcal{M})$, W_2	measures on $\mathcal{M}$ with finite second moment, Wasserstein distance induced by d_g
$\mu \in \mathcal{P}_2(\mathcal{M})$; $\mathbb{P} \in \mathcal{P}(\mathcal{P}_2(\mathcal{M}))$	one population; a law over populations ($\mathbb{P}_0$ source, $\mathbb{P}_1$ data, $\mathbb{P}_{\text{prior}}$ learned prior, $\mathbb{P}_y$ tilted target)
$X = (x_1, \dots, x_N) \in \mathcal{M}^N$, $\mu_X = \text{emp}_N(X)$	labeled population and its empirical measure
$g_N = N^{-1} \oplus g$, $\bar{g}_N = \oplus g$	averaged and unscaled product metrics on $\mathcal{M}^N$
$v_t(x, \mu) \in T_x\mathcal{M}$; $V_t(X)_i = v_t(x_i, \mu_X)$	measure-conditioned velocity; its product (instantaneous) field on $\mathcal{M}^N$
$\mathcal{X}_{s,t}$; $\Psi_{s,t}$, $\widehat{\Psi}^{\theta}_{s,t}$	exact point flow on $\mathcal{M}$; exact and learned distribution flow on $\mathcal{P}_2(\mathcal{M})$
$\Phi_{s,t}$, $\Phi^{\theta}_{s,t}$	exact particle flow on $\mathcal{M}^N$; learned particle map ($\Phi^{N,\theta}_{s,t}$ in Sec. 3.4, N suppressed elsewhere)
$U_{s,t}(X)$, $u_\theta(s, t, x, \mu)$, $\hat{x}_1(x, s, t)$	average-velocity field on $\mathcal{M}^N$; its per-particle network; the endpoint it predicts
$\mathcal{D}^V_s$, $d_2 \log$, $\nabla^1 \log$	material derivative; endpoint and base-point differentials of log
$\mathcal{O}$, y; r_y, ℓ_y, λ_y	observation operator and observation; reward, energy and strength, $r_y = -\lambda_y \ell_y$
b_t, ς_t, D_t, s_t, ω_t, χ_t	reference drift, noise amplitude, diffusivity, score, FK weight, annealing schedule
$D_{\to}$, M, K, n	one-sided Chamfer discrepancy; observed points; walkers; test objects

A.2 Flow matching and exact flow maps

Let p_t be a probability path transported by a time-dependent velocity field v_t. In Euclidean space it satisfies the continuity equation

$$\partial_t p_t + \nabla \cdot (p_t v_t) = 0. \tag{A.1}$$

Conditional flow matching (Lipman et al., 2023) constructs a tractable conditional path X_t^{cond} with conditional velocity v_t^{cond} such that the marginal velocity is

$$v_t(x) = \mathbb{E}\big[v_t^{\text{cond}} \mid X_t^{\text{cond}} = x\big]. \tag{A.2}$$

The same conditional-to-marginal construction extends to Riemannian state spaces using geometric conditional paths (Chen & Lipman, 2024). Given v_t, its exact two-time flow map $\mathcal{X}_{s,t}$ solves

$$\frac{d}{dt}\mathcal{X}_{s,t}(x) = v_t(\mathcal{X}_{s,t}(x))\,, \qquad \mathcal{X}_{s,s} = \mathrm{Id}\,. \tag{A.3}$$

Whenever the flow is well-posed, uniqueness of trajectories implies the composition property

$$\mathcal{X}_{r,t} \circ \mathcal{X}_{s,r} = \mathcal{X}_{s,t}, \qquad s \le r \le t. \tag{A.4}$$

For a non-autonomous flow this is more precisely a two-time flow-composition, or cocycle, property.

A.3 Riemannian finite-time average velocity

Let $(\mathcal{M}, g)$ be a Riemannian manifold with exponential map $\exp_x : T_x\mathcal{M} \to \mathcal{M}$. On a normal neighborhood where a smooth logarithm branch is chosen, define for $s \neq t$

$$U_{s,t}(x) := \frac{1}{t-s}\log_x(\mathcal{X}_{s,t}(x))\,. \tag{A.5}$$

Equivalently,

$$\mathcal{X}_{s,t}(x) = \exp_x((t-s)U_{s,t}(x))\,. \tag{A.6}$$

The vector $U_{s,t}(x) \in T_x\mathcal{M}$ is the initial velocity of the geodesic joining x to the endpoint $\mathcal{X}_{s,t}(x)$ over duration $t-s$. It is therefore a finite-time average or log-velocity and need not coincide with the instantaneous velocity along the actual flow trajectory. For a smooth exact flow,

$$\lim_{s\to t} U_{s,t}(x) = v_t(x), \tag{A.7}$$

and we use the continuous extension $U_{t,t} := v_t$. A learned Riemannian flow-map model directly parameterizes a candidate two-time field $U^\theta_{s,t}$ and map

$$\Phi^\theta_{s,t}(x) := \exp_x\big((t-s)U^\theta_{s,t}(x)\big)\,, \qquad v^\theta_t(x) := U^\theta_{t,t}(x). \tag{A.8}$$

The exponential-map parameterization alone does not imply that $\Phi^\theta_{s,t}$ is the exact flow generated by v^θ_t. Flow-map learning therefore enforces differential and/or composition characterizations of an exact solution operator (Boffi et al., 2025; Davis et al., 2026; Woo et al., 2026). For later reference, the exact forward and backward flow equations are

$$\partial_t\mathcal{X}_{s,t}(x) = v_t(\mathcal{X}_{s,t}(x))\,, \tag{A.9}$$

$$\partial_s\mathcal{X}_{s,t}(x) + d\mathcal{X}_{s,t}(x)\,[v_s(x)] = 0. \tag{A.10}$$

Sec. 3.4 specializes these identities to the measure-conditioned particle lift and rewrites them in the finite-time average-velocity parameterization used for training.

B Finite-particle Wasserstein geometry

This appendix records what the particle lift of Sec. 3.4 does and does not identify with Wasserstein geometry, and the two design choices whose justification is longer than the choices themselves.

B.1 Metric scaling and the empirical quotient

Because $\bar{g}_N = N g_N$ with a constant factor, the Levi–Civita connection, geodesics, exponential and logarithm maps, and parallel transport coincide for the two metrics, while the metric-dependent operators scale as

$$\mathrm{grad}_{g_N} f = N\,\mathrm{grad}_{\bar{g}_N} f, \qquad \Delta_{g_N} = N\Delta_{\bar{g}_N}, \qquad \varsigma_{g_N} = \varsigma_{\bar{g}_N}/\sqrt{N}, \tag{B.1}$$

the last by matching Brownian generators. Adjoints of maps between product tangent spaces are unchanged when source and target metrics are rescaled by the same constant. We use g_N for Wasserstein-compatible distances and losses, and $\bar{g}_N$ for stochastic formulas so that a fixed noise amplitude keeps the same per-particle meaning across N.

For $X, Y \in \mathcal{M}^N$, the quotient distance induced by g_N is

$$\min_{\pi \in S_N} d^2_{g_N}(X, \pi Y) = \min_{\pi \in S_N} \frac{1}{N} \sum_i d^2_g(x_i, y_{\pi(i)}) = W_2^2\big(\mathrm{emp}_N(X), \mathrm{emp}_N(Y)\big), \tag{B.2}$$

so $\mathcal{M}^N/S_N$ is isometric to the equal-weight empirical stratum $\mathcal{E}_N(\mathcal{M})$ with W_2, not to all of $\mathcal{P}_2(\mathcal{M})$. Otto's formal Riemannian calculus and the metric differential structure of Ambrosio–Gigli–Savaré and Gigli concern the ambient Wasserstein space and should not be conflated with this finite quotient (Otto, 2001; Ambrosio et al., 2008; Gigli, 2015). The lift also does not identify $T_X\mathcal{M}^N$ with the Wasserstein tangent cone at $\mathrm{emp}_N(X)$: a labeled particle carries one tangent vector per atom, whereas the tangent cone at an atomic measure can represent infinitesimal splitting of one atom's mass into several transport directions (Lott, 2017). Our differential calculations are statements on the smooth finite-particle lift, with the empirical W_2 quotient used for distances.

B.2 Conditional paths and exchangeability

For a training pair (X^0, X^1), the labeled interpolation $x^{\mathrm{cond}}_{t,i} = \exp_{x^0_i}(t \log_{x^0_i}(x^1_i))$ is always a product-manifold geodesic for the chosen correspondence. If the assignment minimizes equation B.2, its empirical image is a constant-speed W_2 geodesic (McCann, 2001; Ambrosio et al., 2008); without an optimal assignment it remains a valid conditional flow-matching teacher on the product lift. Repeated particles, nonunique assignments and cut-locus points can make the quotient logarithm multivalued; all differential objectives are evaluated on selected smooth local branches. To obtain a permutation-equivariant marginal velocity from labeled conditional paths, the lifted pair law should be jointly exchangeable under simultaneous permutations. Then $V_t(X) = \mathbb{E}[V^{\mathrm{cond}}_t \mid X^{\mathrm{cond}}_t = X]$ admits a permutation-equivariant version and the standard conditional-to-marginal continuity-equation argument applies on $\mathcal{M}^N$. Conditioning on one particle alone defines a different field in general, because every output component is population-conditioned.

B.3 The inner coupling and the scope of "Wasserstein"

We draw X^0 independently of X^1 and fix the particle correspondence, so the conditional path is a geodesic of the labeled product manifold, while its empirical image is a W_2 geodesic only when that correspondence realizes an optimal empirical transport plan, which we do not impose. An optimal inner coupling straightens conditional paths and so reduces discretization error (Tong et al., 2024), which helps an integrated sampler at few function evaluations; a flow map instead obtains few-step accuracy by learning the solution operator. The fixed independent correspondence is also what makes the analytic conditional score of App. E.5 available, since a data-dependent or optimal coupling would contribute an additional source-conditional term. Throughout, "Wasserstein" refers to the distribution-valued state, to the equal-weight empirical W_2 metric induced by the quotient equation B.2, and to the continuum analysis of App. C.1; it does not assert that our conditional paths are W_2 geodesics.

C Proofs for Wasserstein flow maps

C.1 Existence and uniqueness of the distribution flow

The following are sufficient conditions for well-posedness of equation 3.2. Assume $(\mathcal{M}, g)$ is complete, connected, and without boundary, and that $v_t(x, \mu)$ is measurable in t and C^1 in x. Fix $o \in \mathcal{M}$ and suppose there are nonnegative $a, L_x, L_\mu \in L^1(0, 1)$ such that, for almost every t and all $x \in \mathcal{M}$ and $\mu, \nu \in \mathcal{P}_2(\mathcal{M})$,

$$\begin{aligned} \|\nabla_x v_t(x, \mu)\|_{\mathrm{op}} &\le L_x(t), \\ \|v_t(x, \mu) - v_t(x, \nu)\|_g &\le L_\mu(t) W_2(\mu, \nu), \\ \|v_t(x, \mu)\|_g &\le a(t)\big(1 + d_g(x, o) + W_2(\mu, \delta_o)\big). \end{aligned} \tag{C.1}$$

Here $\nabla_x v_t$ is the spatial covariant derivative. The first two bounds control dependence on position and on the population; the last is a linear-growth bound controlling second moments. Then every

$\mu_s \in \mathcal{P}_2(\mathcal{M})$ admits a unique W_2-continuous weak solution on $[s,1]$, given by the self-consistent characteristics

$$\mu_t = (T_{s,t})_{\#}\mu_s, \qquad \partial_t T_{s,t}(x) = v_t(T_{s,t}(x), \mu_t), \qquad T_{s,s}(x) = x, \tag{C.2}$$

where the differential identity holds almost everywhere in time.

To see this, freeze a continuous candidate population path in the characteristic ODE. Completeness and the spatial and growth bounds give a global characteristic flow. Pushing μ_s forward by this flow defines a contraction on sufficiently short time intervals, using the measure Lipschitz bound. The growth bound controls the second moment, so the fixed-point construction extends to $[s,1]$. Comparing coupled characteristics and applying Grönwall's inequality gives

$$W_2(\mu_t, \nu_t) \le \exp\left(\int_s^t [L_x(r) + L_\mu(r)]\, dr\right) W_2(\mu_s, \nu_s). \tag{C.3}$$

The spatial regularity also ensures that every W_2-continuous weak solution is represented by its characteristic flow; hence this estimate proves uniqueness for the continuity equation itself. Thus $\Psi_{s,t}(\mu_s) := \mu_t$ is well defined, and uniqueness implies its semigroup property. For the characteristic approach to nonlocal transport in Euclidean space, see Piccoli & Rossi (2013).

In particular, take $\mu_s = \mu_X$ atomic. Then equation C.2 gives $\Psi_{s,t}(\mu_X) = \mu_{\Phi_{s,t}(X)}$ with $\Phi_{s,t} = T_{s,t}^{\otimes N}$ acting coordinate-wise, and each $x_{i,t} = T_{s,t}(x_i)$ solves the self-consistent particle system $\dot{x}_{i,t} = v_t(x_{i,t}, \mu_{X_t})$. The equal-weight empirical stratum is therefore invariant, and the distribution flow restricts to it exactly. This is the hypothesis equation C.4 of Lemma C.1 below, which the learned map equation 3.7 satisfies by construction.

C.2 From distribution identities to empirical residuals

Lemma C.1 (Empirical weak flow-map identities)**.** *Assume the particle dynamics generated by $V_r(Z)_j = v_r(z_j, \mu_Z)$ admit local solutions. Let $\Phi_{s,t} : \mathcal{M}^N \to \mathcal{M}^N$ be a C^1 family realizing a distribution-valued family $\Psi_{s,t}$ on empirical measures:*

$$\Psi_{s,t}(\mu_X) = \mu_{\Phi_{s,t}(X)}, \qquad \mu_X = \frac{1}{N}\sum_{i=1}^N \delta_{x_i}. \tag{C.4}$$

Fix an input population X_s and write $X_t = \Phi_{s,t}(X_s)$, with components $x_{i,t}$. If Ψ satisfies the Lagrangian identity equation 3.4, then

$$\frac{1}{N}\sum_{i=1}^N \langle \nabla_g \varphi(x_{i,t}), R_{\alpha,i}(s,t,X_s)\rangle_g = 0 \qquad \text{for every } \varphi \in C_c^\infty(\mathcal{M}) \tag{C.5}$$

holds with $\alpha = \mathrm{L}$. If Ψ satisfies the Eulerian identity equation 3.5 along solutions of equation 3.2, the same conclusion holds with $\alpha = \mathrm{E}$. Here the particle residuals are those defined in equation 3.9 and equation 3.10.

Proof. Fix $\varphi \in C_c^\infty(\mathcal{M})$.

Lagrangian identity. Keep s and X_s fixed. Testing equation 3.4 against φ, with $\mu_t = \Psi_{s,t}(\mu_{X_s})$, gives

$$\frac{d}{dt}\int_{\mathcal{M}} \varphi\, d\mu_t = \int_{\mathcal{M}} \langle \nabla_g \varphi(x), v_t(x,\mu_t)\rangle_g\, d\mu_t(x). \tag{C.6}$$

This uses the distributional definition of divergence and does not require μ_t to have a density. By equation C.4, $\mu_t = \mu_{X_t}$. The chain rule therefore gives

$$\begin{aligned}\frac{d}{dt}\int_{\mathcal{M}} \varphi\, d\mu_t &= \frac{1}{N}\sum_{i=1}^N \frac{d}{dt}\varphi(\Phi_{s,t,i}(X_s)) \\ &= \frac{1}{N}\sum_{i=1}^N \langle \nabla_g \varphi(x_{i,t}), \partial_t \Phi_{s,t,i}(X_s)\rangle_g .\end{aligned}$$

The right-hand side of equation C.6 is likewise the empirical average

$$\frac{1}{N}\sum_{i=1}^{N}\langle \nabla_g \varphi(x_{i,t}), v_t(x_{i,t}, \mu_{X_t})\rangle_g .$$

Subtracting these expressions proves equation C.5 for $\alpha = \mathrm{L}$.

Eulerian identity. Fix the terminal time t and let $Z_r = (z_{1,r}, \dots, z_{N,r})$ be an input trajectory satisfying

$$Z_s = X_s, \qquad \dot{z}_{j,r} = v_r(z_{j,r}, \mu_{Z_r}).$$

Its empirical measure solves equation 3.2 in the weak sense: for every $\psi \in C_c^\infty(\mathcal{M})$,

$$\begin{aligned}\frac{d}{dr}\int_{\mathcal{M}} \psi \, d\mu_{Z_r} &= \frac{1}{N}\sum_{j=1}^{N}\langle \nabla_g \psi(z_{j,r}), v_r(z_{j,r}, \mu_{Z_r})\rangle_g \\ &= \int_{\mathcal{M}} \langle \nabla_g \psi(x), v_r(x, \mu_{Z_r})\rangle_g \, d\mu_{Z_r}(x).\end{aligned}$$

Consequently, equation 3.5 applies along μ_{Z_r}. Put $Y_r = \Phi_{r,t}(Z_r)$, so that $Y_s = X_t$ and $\Psi_{r,t}(\mu_{Z_r}) = \mu_{Y_r}$ by equation C.4. For each output particle, the chain rule at $r = s$ yields

$$\begin{aligned}\frac{d}{dr} y_i(r)\Big|_{r=s} &= \partial_s \Phi_{s,t,i}(X_s) + \sum_{j=1}^{N} d_{x_j}\Phi_{s,t,i}(X_s)[\dot{z}_{j,s}] \\ &= \partial_s \Phi_{s,t,i}(X_s) + \sum_{j=1}^{N} d_{x_j}\Phi_{s,t,i}(X_s)\big[v_s(x_{j,s}, \mu_{X_s})\big] \\ &= R_{\mathrm{E},i}(s,t,X_s).\end{aligned}$$

Here $y_i(r)$ denotes component i of Y_r, and $d_{x_j}\Phi_{s,t,i}$ includes the dependence on particle j through the empirical context. Differentiating the empirical test average now gives

$$\begin{aligned}0 &= \frac{d}{dr}\int_{\mathcal{M}} \varphi \, d\Psi_{r,t}(\mu_{Z_r})\Big|_{r=s} \\ &= \frac{1}{N}\sum_{i=1}^{N}\left\langle \nabla_g \varphi(y_i(s)), \frac{d}{dr} y_i(r)\Big|_{r=s}\right\rangle_g \\ &= \frac{1}{N}\sum_{i=1}^{N}\langle \nabla_g \varphi(x_{i,t}), R_{\mathrm{E},i}(s,t,X_s)\rangle_g ,\end{aligned}$$

which proves equation C.5 for $\alpha = \mathrm{E}$. Neither argument requires distinct particles. □

For a learned family $\widehat{\Psi}$ with empirical realization Φ, the same calculations identify the two weak residuals. Their vanishing follows when the corresponding distribution-level flow identities hold; it is not automatic for an arbitrary parameterized map.

C.3 From particle residuals to average-velocity characterizations

Proof of the differential statements in Proposition 3.1. Fix s, t, X_s and write $X_t = \Phi_{s,t}(X_s)$ and $\Delta = t - s$. The exponential parameterization and the compatible logarithm branch give

$$\log_{x_{i,s}}(x_{i,t}) = \Delta U_{s,t,i}(X_s). \tag{C.7}$$

Let

$$J_i := d_2 \log_{x_{i,s}}(x_{i,t}) : T_{x_{i,t}}\mathcal{M} \longrightarrow T_{x_{i,s}}\mathcal{M}.$$

This endpoint differential is invertible on the chosen neighborhood. We show that applying J_i to the particle residuals yields precisely the average-velocity residuals, with a minus sign for the Eulerian one.

Lagrangian characterization. Keep s and X_s fixed. Differentiating equation C.7 in t gives

$$J_i\big[\partial_t \Phi_{s,t,i}(X_s)\big] = U_{s,t,i}(X_s) + \Delta\, \partial_t U_{s,t,i}(X_s).$$

Applying J_i to the residual equation 3.9 therefore gives

$$\begin{aligned} J_i[R_{\mathrm{L},i}] &= J_i\big[\partial_t \Phi_{s,t,i}(X_s) - v_t(x_{i,t}, \mu_{X_t})\big] \\ &= U_{s,t,i}(X_s) + \Delta\, \partial_t U_{s,t,i}(X_s) - J_i\big[v_t(x_{i,t}, \mu_{X_t})\big] \\ &= r_{\mathrm{L},i}. \end{aligned} \tag{C.8}$$

Since J_i is invertible, $R_{\mathrm{L},i} = 0$ if and only if $r_{\mathrm{L},i} = 0$. Isolating $U_{s,t,i}(X_s)$ in the latter equality gives the Lagrangian characterization equation 3.13.

Eulerian characterization. Fix t and move the input along a trajectory Z_r satisfying $Z_s = X_s$ and $\dot{Z}_r = V_r(Z_r)$. Put $Y_r = \Phi_{r,t}(Z_r)$. As in the preceding proof, $y_i(s) = x_{i,t}$ and

$$\frac{d}{dr} y_i(r)\bigg|_{r=s} = \partial_s \Phi_{s,t,i}(X_s) + \sum_{j=1}^{N} d_{x_j} \Phi_{s,t,i}(X_s)\big[v_s(x_{j,s}, \mu_{X_s})\big] = R_{\mathrm{E},i}. \tag{C.9}$$

Along this input trajectory, the logarithm identity reads

$$\log_{z_{i,r}}(y_i(r)) = (t - r) U_{r,t,i}(Z_r).$$

Both sides are tangent vectors at the moving base point $z_{i,r}$. Taking their covariant derivatives at $r = s$ gives

$$\begin{aligned} &\nabla^1_{v_s(x_{i,s}, \mu_{X_s})} \log_{x_{i,s}}(x_{i,t}) + J_i[R_{\mathrm{E},i}] \\ &\qquad = -U_{s,t,i}(X_s) + \Delta\big(\mathcal{D}_s^{V_s} U_{s,t}(X_s)\big)_i. \end{aligned} \tag{C.10}$$

The two terms on the left account for motion of the base point and the endpoint, respectively. On the right, differentiating $t - r$ gives $-U_{s,t,i}$, while differentiating the field along Z_r gives the material derivative defined in equation 3.11. In particular, this derivative includes the change in empirical context caused by all input particles. Rearranging yields

$$\begin{aligned} -J_i[R_{\mathrm{E},i}] &= U_{s,t,i}(X_s) - \Delta\big(\mathcal{D}_s^{V_s} U_{s,t}(X_s)\big)_i \\ &\quad + \nabla^1_{v_s(x_{i,s}, \mu_{X_s})} \log_{x_{i,s}}(x_{i,t}) \\ &= r_{\mathrm{E},i}. \end{aligned} \tag{C.11}$$

Invertibility of J_i shows that $R_{\mathrm{E},i} = 0$ if and only if $r_{\mathrm{E},i} = 0$. Isolating $U_{s,t,i}(X_s)$ gives the Eulerian characterization equation 3.14.

Diagonal condition. The parameterization gives $\Phi_{s,s}(X_s) = X_s$. At the diagonal, $d_2 \log_x(x) = \mathrm{Id}$ and $\nabla^1_a \log_x(x) = -a$. Using the smooth extension of U to $s = t$, the two transformed residuals reduce to

$$r_{\mathrm{L},i}(s, s, X_s) = r_{\mathrm{E},i}(s, s, X_s) = U_{s,s,i}(X_s) - v_s(x_{i,s}, \mu_{X_s}).$$

Thus either differential characterization implies $U_{s,s}(X_s) = V_s(X_s)$. □

The transformations above establish equivalence of the individual particle residuals and their average-velocity counterparts, without requiring distinct particles. To infer these individual conditions from the weak identities in Lemma C.1, distinct output atoms suffice: smooth compactly supported test functions can have independently prescribed gradients at those atoms. At coincident atoms, the weak identities constrain only the sum of particle residuals at each occupied location.

D Training objectives and implementation

This appendix gives the conditional targets, the exact losses, the differential implementation of the material derivative and logarithm differentials, and every training-time setting. It expands the "Objectives" paragraph of Sec. 3.4.

D.1 Conditional training

The characterizations of Prop. 3.1 are written in terms of the marginal field V_t, which is not available from endpoint samples. As in conditional flow matching, draw endpoint populations (X^0, X^1) under the population coupling and fixed particle correspondence of App. B.3 and define the product-manifold interpolation and its velocity

$$x_{t,i}^{\mathrm{cond}} = \exp_{x_i^0}\big(t \log_{x_i^0}(x_i^1)\big), \qquad V_{t,i}^{\mathrm{cond}} = \frac{1}{1-t} \log_{x_{t,i}^{\mathrm{cond}}}(x_i^1) \quad (t < 1), \tag{D.1}$$

so that the marginal product field satisfies $V_t(X) = \mathbb{E}[V_t^{\mathrm{cond}} \mid X_t^{\mathrm{cond}} = X]$ (App. B.2). Whenever a regression target is affine in its velocity argument, replacing V_t by V_t^{cond} gives an unbiased conditional target; the argument takes place in the common tangent space $T_X\mathcal{M}^N$ and applies on Euclidean and Riemannian base manifolds alike. The Eulerian target is linear in the velocity direction, so V_s^{cond} is the usual conditional teacher. The Lagrangian target is linear in its endpoint velocity, but that velocity is evaluated at the *model* endpoint rather than the sampled conditional endpoint, which is why we use the endpoint-anchored surrogate below. The exact geodesic conditional velocity is bounded despite the $(1-t)^{-1}$ in its formula, since along the selected minimizing geodesic $\|V_{t,i}^{\mathrm{cond}}\|_g = d_g(x_i^0, x_i^1)$; endpoint truncation (below) is therefore a numerical and log-branch choice, not a remedy for a velocity blow-up.

D.2 Losses

Write $\mathrm{sg}[\cdot]$ for a stop-gradient.

Semigroup. With $X_s = X_s^{\mathrm{cond}}$, $\widehat{X}_t^{\mathrm{dir}} = \Phi_{s,t}^\theta(X_s)$ and $\widehat{X}_t^{\mathrm{cmp}} = \Phi_{r,t}^\theta(\Phi_{s,r}^\theta(X_s))$,

$$\mathcal{L}_{\mathrm{sg}} = \mathbb{E}\frac{1}{N}\sum_i d_g^2\big((\widehat{X}_t^{\mathrm{dir}})_i, \mathrm{sg}[(\widehat{X}_t^{\mathrm{cmp}})_i]\big), \quad \mathcal{L}_{\mathrm{diag}} = \mathbb{E}\frac{1}{N}\sum_i \big\|u_\theta(s, s, x_{s,i}^{\mathrm{cond}}, \mu_{X_s^{\mathrm{cond}}}) - V_{s,i}^{\mathrm{cond}}\big\|_g^2, \tag{D.2}$$

where the diagonal anchor $\mathcal{L}_{\mathrm{diag}}$ is needed because composition alone does not identify the generator.

Eulerian. With $X_s = X_s^{\mathrm{cond}}$ and $\widehat{X}_t = \Phi_{s,t}^\theta(X_s)$,

$$\mathcal{L}_{\mathrm{eul}} = \mathbb{E}\Big\|U_{s,t}^\theta(X_s) - \mathrm{sg}\big[(t-s)\mathcal{D}_s^{V_s^{\mathrm{cond}}} U_{s,t}^\theta(X_s) - \nabla^1_{V_s^{\mathrm{cond}}} \mathrm{Log}_{X_s}^{\mathcal{M}^N}(\widehat{X}_t)\big]\Big\|_{g_N}^2, \tag{D.3}$$

where the material derivative acts on the entire cloud including the co-evolving measure context.

Lagrangian. The exact target needs the marginal velocity at the model endpoint. We therefore draw $X_t = X_t^{\mathrm{cond}}$, back-predict $\widehat{X}_s = \Phi_{t,s}^\theta(X_t)$, and train

$$\begin{gathered}\mathcal{L}_{\mathrm{lag}} = \mathbb{E}\Big\|U_{s,t}^\theta(\widehat{X}_s) - \mathrm{sg}\big[d_2 \mathrm{Log}_{\widehat{X}_s}^{\mathcal{M}^N}(X_t)[V_t^{\mathrm{cond}}] - (t-s)\partial_t U_{s,t}^\theta(\widehat{X}_s)\big]\Big\|_{g_N}^2, \\ \mathcal{L}_{\mathrm{cyc}} = \mathbb{E}\, d_{g_N}^2\big(\Phi_{s,t}^\theta(\Phi_{t,s}^\theta(X_t)), X_t\big).\end{gathered} \tag{D.4}$$

Here V_t^{cond} is an unbiased teacher for $V_t(X_t)$ given the sampled endpoint, so the conditional expectation of the target given X_t is the exact Lagrangian target. Exactness additionally requires conditioning on $\widehat{X}_s$ to carry the same information as conditioning on X_t, that is, $\Phi_{t,s}^\theta$ injective on the support; the cycle loss monitors this condition rather than implying it.

D.3 Differential implementation

The product field is $U_{s,t}^\theta(X) = (u_\theta(s,t,x_i,\mu_X))_{i=1}^N$ and, for $V \in T_X\mathcal{M}^N$, $\mathcal{D}_s^V U_{s,t}^\theta(X) = \partial_s U_{s,t}^\theta(X) + \nabla_V^{\mathcal{M}^N} U_{s,t}^\theta(X)$. Because the measure context co-evolves with the cloud, the ith output differentiates through all particle coordinates: in an embedded implementation this is one ambient JVP through the full particle input followed by tangential projection, which is the covariant derivative by the Gauss formula. The logarithm derivative $\nabla_v^1 \mathrm{Log}_X^{\mathcal{M}^N}(Y)$ holds the candidate endpoint Y fixed while differentiating the source. When computing $\partial_t U_{s,t}^\theta(\widehat{X}_s)$ in the Lagrangian loss, $\widehat{X}_s$ and its measure context are held fixed.

On the unit sphere, for $x, y \in \mathbb{S}^{d-1}$ with $0 < \vartheta = d_g(x,y) < \pi$, let $e_x = \log_x(y)/\vartheta$ and $e_y = P_{x\to y} e_x$. Decomposing $v = v_\parallel + v_\perp$ and $w = w_\parallel + w_\perp$ relative to these radial directions,

$$\nabla^1_v \log_x(y) = -v_\parallel - \vartheta \cot\vartheta\, v_\perp, \qquad d_2 \log_x(y)[w] = P_{y\to x}\big(w_\parallel + \tfrac{\vartheta}{\sin\vartheta} w_\perp\big). \tag{D.5}$$

These are smooth away from the antipodal cut locus, and their $\vartheta \to 0$ limits give the diagonal identities $\nabla^1_v \log_x(x) = -v$ and $d_2 \log_x(x) = \mathrm{Id}$ used in Sec. 3.4. Both are implemented in closed form and verified against finite differences.

D.4 Parameterization, time sampling and objective settings

Parameterization. The network predicts the endpoint $\hat{x}_1(x,s,t)$ as a residual on its input, $\hat{x}_1 = \Pi_{\mathcal{M}}(x + \delta_\theta(x,s,t))$, with the output projection initialized at scale 10^{-3} (not zero: a zero-initialized head places $\hat{x}_1$ exactly at x, where the spherical logarithm is singular). The finite-time map is the geodesic step

$$u_{s,t}(x) = \frac{\log_x(\hat{x}_1(x,s,t))}{\max(1-s,\varepsilon)}, \qquad \Phi^\theta_{s,t}(x) = \exp_x\big((t-s)u_{s,t}(x)\big), \qquad \varepsilon = 0.05, \tag{D.6}$$

so the diagonal $s = t$ is the identity and the velocity field is $V_s = u_{s,s}$. For $s > 1-\varepsilon$ the terminal estimate undershoots the network's own endpoint prediction by the factor $(1-s)/\varepsilon$; we clip rather than extrapolate because extrapolating biases the map.

Time sampling. $s \sim \mathcal{U}(0, s_{\max})$ with $s_{\max} = 0.94$; with probability 0.33 the sample is placed on the diagonal $t = s$, otherwise $t = s + (1-s)\mathcal{U}(0,1)$, and the intermediate time is $r = s + (t-s)\mathcal{U}(0,1)$. There is deliberately no forced $t = 1$: concentrating mass on the largest, least reliable gap destabilizes the bootstrap. Together with $s_{\max}$ this is the endpoint truncation referred to above.

Objectives as implemented. Let $\mathcal{A}[\delta] = \|\delta\|^2/\mathrm{sg}[\|\delta\|^2 + c]^p$ denote the adaptive-L^2 normalizer with $p = 0.5$, $c = 10^{-3}$.

- **Semigroup**: $\mathcal{A}$ applied to the endpoint-space residual between $\Phi^\theta_{s,t}$ and $\mathrm{sg}[\Phi^\theta_{r,t}(\Phi^\theta_{s,r})]$, with a stop-gradient on both composed legs. We use the endpoint-space form because the velocity-space form with its self-referential $1/(t-s)$ target explodes when training from scratch.
- **Eulerian**: one forward-mode JVP with tangent $(v_{\mathrm{true}}, 1)$ yields the material derivative along the flow, which automatically contains the cross-particle term; the target is $\mathrm{sg}[(t-s)\mathcal{D}_s u + \mathrm{corr}]$ with $\mathrm{corr} = v_{\mathrm{true}}$ in the Euclidean case and $\mathrm{corr} = -\nabla^1_v \log_x(y) = v_\parallel + \vartheta\cot\vartheta\, v_\perp$ on the sphere. On the sphere $\mathcal{D}_s u$ is tangent-projected before use, because the ambient JVP of a tangent field acquires a spurious normal component as the base point moves.
- **Lagrangian**: a JVP in t only, with the base point fixed; target $\mathrm{sg}[d_2 \log_x(y)[V_t] - (t-s)\partial_t u_{s,t}]$, wrapped in the self-distilled form: the regression point is the back-mapped $\mathrm{sg}[\Phi^\theta_{t,s}(x_t)]$ and the cycle term $\|\Phi^\theta_{s,t}(\Phi^\theta_{t,s}(x_t)) - x_t\|^2$ is added at weight 1.0 with gradients through both legs.

The semigroup and Lagrangian losses carry the diagonal anchor $\|u_{s,s} - V_s\|^2$ at weight 1.0 in velocity space; the Eulerian branch does not, since it self-anchors (Sec. 3.4). We keep the anchor on the Lagrangian branch as an optimization stabilizer even though that identity also self-anchors.

Riemannian handling. Spherical operations are reformulated to be autodiff-safe: inner products are clipped into $[-1+10^{-7}, 1-10^{-7}]$ so that $\sin\vartheta$ is bounded away from zero (a `where`-guarded formulation differentiates both branches and produces NaN through $\arccos'$), and the exponential map uses a soft norm $\sqrt{\|v\|^2 + 10^{-12}}$.

Score model. In the Euclidean domains the score is an algebraic plug-in from the diagonal field: with $b_s(x) = (\hat{x}_1(x,s,s) - x)/(1-s)$, $\nabla \log p_s(x) = (s\, b_s(x) - x)/(1-s)$, which is equation E.17. This identity is exact only for a unit-variance isotropic source; several runs use a data-fitted anisotropic source, in which case the plug-in omits a Σ_0^{-1} factor. It enters only through the nuisance drift

$\frac{1}{2}\varsigma_t^2 \nabla \log p_s$, not through the reward, so we retain it and flag the approximation. On the sphere no such identity holds and we train a parallel tangent-valued score head on the shared backbone by conditional score regression against the analytic target of Prop. E.3,

$$\nabla_{x_t} \log p_t(x_t \mid x_1) = (d-2)\Big[\tfrac{\cot\vartheta}{1-t} - \cot\phi\Big]\hat{n}_{\text{away}}, \qquad \phi = \arccos\langle x_t, x_1\rangle,\ \vartheta = \tfrac{\phi}{1-t},$$

which is exact for a uniform source on the sphere (enforced in all score-head runs). The regression is scale-free: the squared error is divided by a per-cloud $\mathbb{E}\|\text{target}\|^2$ so that a unit weight balances the adaptive-L^2 flow losses. The score interval is $s \in [0, 0.9]$ with a $(1-s)^2$ time weight, and $1-t$ is floored at 10^{-3} inside the target.

E Inference-time methods

This appendix first specifies the finite-particle realization of the target in Sec. 4.1, then details the two families of inference-time methods in Sec. 4.2: deterministic steering (lookahead, FlowDPS, PnP-Flow) and weighted correction (the Riemannian Feynman–Kac corrector, the reference score it needs, and its FMTT instantiation). Guidance hyperparameters are listed in App. G.4.

E.1 The inverse target on the particle lift

At fixed particle count N, let P_1 be the law of the generator output $X_1 \in \mathcal{M}^N$ and define the prior over represented populations by

$$\mathbb{P}^N_{\text{prior}} := \text{emp}_N \#P_1 = \text{Law}(\text{emp}_N(X_1))\,. \tag{E.1}$$

This is the finite-particle realization of $\mathbb{P}_{\text{prior}}$ in Sec. 4.1. For the data-consistency energies used in our inverse experiments, write

$$r_y(\mu) := -\lambda_y \ell_y(\mu), \qquad \ell_y^N(X) := \ell_y(\text{emp}_N(X)), \qquad r_y^N(X) := r_y(\text{emp}_N(X)), \tag{E.2}$$

where $\ell_y : \mathcal{P}_2(\mathcal{M}) \to [0,\infty)$ and $\lambda_y \geq 0$. The population target is

$$\mathrm{d}\mathbb{P}^N_y(\mu) = \frac{e^{r_y(\mu)}}{Z_y}\,\mathrm{d}\mathbb{P}^N_{\text{prior}}(\mu), \qquad Z_y = \int e^{r_y(\mu)}\,\mathrm{d}\mathbb{P}^N_{\text{prior}}(\mu). \tag{E.3}$$

The same construction includes a likelihood reward $r_y(\mu) = \log p(y \mid \mu)$, but our reported inverse experiments use deterministic observations and the energies in App. G.1.

Since the lifted reward depends on X only through $\text{emp}_N(X)$, tilting commutes with the empirical pushforward: for any law P on $\mathcal{M}^N$ with $0 < Z := \int e^{r_y^N}\,\mathrm{d}P < \infty$,

$$\text{emp}_N \#\left(\frac{e^{r_y^N}}{Z}P\right) = \frac{e^{r_y}}{Z}\ \text{emp}_N \#P. \tag{E.4}$$

Indeed, integrating a test function $f(\mu)$ against either side gives

$$Z^{-1}\int f(\text{emp}_N(X)) e^{r_y(\text{emp}_N(X))}\,\mathrm{d}P(X).$$

Thus a terminal particle law proportional to $e^{r_y^N} P_1$ has exactly the desired population pushforward $\mathbb{P}^N_y$. This identity needs only measurability and a finite normalizer; the smoothness assumptions for the dynamics are separate. All exact inverse-inference statements below are at fixed N and make no claim about a nondegenerate $N \to \infty$ target. We suppress the superscript N on particle maps, writing $\Phi^\theta_{s,t}$ for $\Phi^{N,\theta}_{s,t}$.

E.2 Deterministic steering

Soft values and stochastic conditioning. For a stochastic reference process with marginal density p_t, drift b_t, noise amplitude ς_t and diffusivity $D_t = \varsigma_t^2/2$, the generator is $\mathcal{L}_t f = \langle b_t, \text{grad}_{\bar{g}_N} f\rangle_{\bar{g}_N} +$

$D_t\Delta_{\bar{g}_N} f$. Define $H_t^y(X) := \mathbb{E}[e^{r_y^N(X_1)} \mid X_t = X]$ and $h_t^y := \log H_t^y$. Under the regularity needed for the backward Kolmogorov equation,

$$\partial_t H_t^y + \mathcal{L}_t H_t^y = 0, \qquad H_1^y = e^{r_y^N}. \tag{E.5}$$

The terminally tilted path law has drift $b_t + 2D_t \operatorname{grad}_{\bar{g}_N} h_t^y$ and initial density proportional to $p_0 e^{h_0^y}$. The drift follows by expanding $H^{-1}[\mathcal{L}_t(Hf) - f\mathcal{L}_t H]$; its terminal density is proportional to $p_1 e^{r_y^N}$. In general this construction requires both the exact stochastic conditional expectation and the reweighted initial law. The deterministic endpoint reward used for lookahead need not equal this stochastic soft value.

Deterministic endpoint conditioning. For an exact deterministic endpoint map, $X_1 = \Phi_{t,1}(X_t)$ conditionally, so the terminal reward has no conditional Jensen gap. An arbitrary reward-gradient perturbation of the trajectory nevertheless need not realize the desired terminal tilt (Xu et al., 2026): exact conditioning with the unchanged deterministic map is a reweighting of the source,

$$\mathrm{d}\pi_0^y(X) = \frac{e^{r_y^N(\Phi_{0,1}(X))}\,\mathrm{d}p_0(X)}{\int e^{r_y^N(\Phi_{0,1}(\bar{X}))}\,\mathrm{d}p_0(\bar{X})}, \qquad (\Phi_{0,1})_{\#}\pi_0^y = \frac{e^{r_y^N} p_1}{\int e^{r_y^N}\,\mathrm{d}p_1}, \tag{E.6}$$

so lookahead guidance, FlowDPS and PnP-Flow are practical deterministic approximations, while the weighted construction of App. E.3 is what realizes prescribed tilted marginals.

Lookahead guidance. For a reward differentiable at the predicted endpoint, or an almost-everywhere weak gradient of a locally Lipschitz reward, the through-map gradient is $G_t(X) = (d\Phi_{t,1}^\theta(X))^* \operatorname{grad}_{g_N} r_y^N(\Phi_{t,1}^\theta(X))$ and a Riemannian step of size γ_t is $X^+ = \operatorname{Exp}_X^{\mathcal{M}^N}(\gamma_t G_t(X))$. The adjoint and gradient use the same metric; using the $\bar{g}_N$ gradient at fixed N scales it by $1/N$, absorbed into the tuned step size, and does not turn the heuristic into an exact sampler. In the reported discrete scheme, with $L = \ell_y^N(\Phi_{s,1}^\theta(X))$ and $g = \operatorname{grad}_{\bar{g}_N} L$ evaluated at the pre-step state,

$$X_t = \operatorname{retract}\Big(X,\ \Phi_{s,t}^\theta(X),\ \frac{(t-s)\,\lambda_y}{\sqrt{L} + 10^{-8}}\, g\Big)\,: \tag{E.7}$$

the unguided map first advances the cloud to $\Phi_{s,t}^\theta(X)$ and retract then applies the stored correction in the negative-gradient direction of the energy. In Euclidean domains this means subtracting the third argument from the advanced state. The $1/\sqrt{L}$ factor is the DPS-style adaptive normalization; the $(t-s)$ factor keeps the sum of the nominal guidance coefficients equal to λ_y across a partition of $[0,1]$, although the state-dependent gradients and normalization still depend on the discretized trajectory. On a manifold, the gradient at X must be transferred to the tangent space at the advanced state before the geodesic retraction; tangent projection provides a local approximation to parallel transport. Each step costs two map evaluations, one at $(s,1)$ and one at (s,t), and one vector–Jacobian product.

Riemannian FlowDPS and PnP-Flow. Write $r_y^N = -\lambda_y \ell_y^N$. For FlowDPS (Kim et al., 2025) we predict $\widehat{X}_{0|t}$ and $\widehat{X}_{1|t}$, correct the clean endpoint by a Riemannian descent step on ℓ_y^N, optionally perturb the source prediction, and re-interpolate geometrically: if $\widetilde{X}_{0|t} = (\tilde{a}_i^t)$ and $\widetilde{X}_{1|t} = (\tilde{b}_i^t)$, then $x_i^{t+\Delta t} = \exp_{\tilde{a}_i^t}((t+\Delta t)\log_{\tilde{a}_i^t}(\tilde{b}_i^t))$, a product-manifold geodesic interpolation; if the optional perturbation approximates a diffusion step, its tangent-space scale is proportional to $\sqrt{\Delta t}$. FlowDPS differentiates the reward at the endpoint estimate only, without backpropagating through the map, and carries no $(t-s)$ factor, so it is not step-count invariant. For PnP-Flow (Martin et al., 2025), each outer iteration takes a task-specific proximal step on ℓ_y^N (masked pull, geodesic slerp, bulk-mean shift or per-voxel mean shift), reprojects onto the learned path, and applies the time-t denoiser, with proximal weight $\gamma = t^\alpha$, $\alpha \in \{0.1, 0.3, 0.5\}$; on the sphere the fidelity gradient is tangent-projected before the retraction and the reprojection draws the source from the spherical conditional of App. E.5 rather than adding ambient Gaussian noise. Both baselines run on the pretrained WFM/RWEFM.

E.3 Feynman–Kac steering on the particle lift

FMTT is an instance of the Feynman–Kac corrector of (Skreta et al., 2025): a guided SDE whose samples carry importance weights chosen so that the weighted marginal at time t is a prescribed

tilt $\pi_t \propto p_t e^{u_t}$ of the reference marginal p_t. This subsection states the Riemannian form of that construction on $(\mathcal{M}^N, \bar{g}_N)$, which is what the FMTT rate of App. E.4 rests on. Gradients, divergences and Laplacians are the intrinsic operators of $\bar{g}_N$ (App. B.1 converts to g_N) and densities are relative to its Riemannian volume.

Let p_t be a positive $C^{1,2}$ density solving the reference Fokker–Planck equation

$$\partial_t p_t = -\operatorname{div}(p_t b_t) + D_t \Delta p_t, \qquad D_t := \tfrac{1}{2}\varsigma_t^2, \tag{E.8}$$

with score $s_t := \operatorname{grad} \log p_t$ and probability-flow velocity $V_t := b_t - D_t s_t$. Positivity on the region visited by the guided dynamics is an assumption: it is immediate for the Euclidean path, and on the sphere it is inherited for every $t < 1$ from a terminal law with full support, since the conditional equation E.18 is supported on a cap of radius $(1-t)\pi$; we train and sample on $t \le 1 - \varepsilon$. Let $u_t \in C^{1,2}$ be a tilt with $\pi_t = Z_t^{-1} p_t e^{u_t}$, and assume the boundary and integrability conditions needed to normalize π_t and to test the equations below.

Proposition E.1 (Pinned Riemannian Feynman–Kac corrector)**.** *The process*

$$\begin{aligned} \mathrm{d}X_t &= \big[b_t + D_t \operatorname{grad} u_t\big](X_t)\,\mathrm{d}t + \varsigma_t\,\mathrm{d}W_t^{\bar{g}_N}, \\ \mathrm{d}\log \omega_t &= g_t(X_t)\,\mathrm{d}t, \end{aligned} \qquad g_t := \partial_t u_t + \langle \operatorname{grad} u_t, V_t\rangle, \tag{E.9}$$

started from $X_0 \sim p_0$ with $\omega_0 = e^{u_0(X_0)}$, has normalized weighted marginal π_t: $\mathbb{E}[\omega_t \varphi(X_t)]/\mathbb{E}[\omega_t] = \int \varphi \,\mathrm{d}\pi_t$ for every smooth compactly supported φ.

Proof. Write $\widetilde{b}_t = b_t + D_t \operatorname{grad} u_t$ and $f = \pi_t/p_t = e^{u_t}/Z_t$, so that $\operatorname{grad} f = f \operatorname{grad} u_t$ and $\operatorname{grad}\log \pi_t = s_t + \operatorname{grad} u_t$. In the Fokker–Planck operator of the guided drift, $\mathcal{L}^*_{\widetilde{b}_t}\pi_t = -\operatorname{div}(\pi_t \widetilde{b}_t) + D_t \Delta \pi_t$, the guidance term cancels against the diffusion term because its coefficient is D_t, leaving $-\operatorname{div}(\pi_t b_t) + D_t \operatorname{div}(\pi_t s_t)$. The product rule gives $\operatorname{div}(\pi_t b_t) = f \operatorname{div}(p_t b_t) + \pi_t \langle \operatorname{grad} u_t, b_t\rangle$ and $\operatorname{div}(\pi_t s_t) = f \Delta p_t + \pi_t \langle \operatorname{grad} u_t, s_t\rangle$, so by equation E.8, $\mathcal{L}^*_{\widetilde{b}_t}\pi_t = \frac{\pi_t}{p_t}\partial_t p_t - \pi_t \langle \operatorname{grad} u_t, V_t\rangle$. Differentiating $\pi_t = Z_t^{-1} p_t e^{u_t}$ in time and subtracting, $\partial_t \pi_t = \mathcal{L}^*_{\widetilde{b}_t}\pi_t + \pi_t\big(g_t - \partial_t \log Z_t\big)$, and integrating over $\mathcal{M}^N$ gives $\partial_t \log Z_t = \mathbb{E}_{\pi_t}[g_t]$. Hence $\rho_t := \pi_t \exp \int_0^t \mathbb{E}_{\pi_r}[g_r]\,\mathrm{d}r$ solves the unnormalized equation $\partial_t \rho_t = \mathcal{L}^*_{\widetilde{b}_t}\rho_t + g_t \rho_t$, which by Itô's formula is the equation solved by the weighted law of (X_t, ω_t); normalizing recovers π_t, and $\omega_0 = e^{u_0(X_0)}$ matches the initial weighted law up to the scalar Z_0. □

The rate

$$g_t = \partial_t u_t + \langle \operatorname{grad} u_t, V_t\rangle \tag{E.10}$$

is the material derivative of the tilt along the probability flow. For a general guidance coefficient κ_t in place of D_t the same calculation adds $-(D_t - \kappa_t)\big(\Delta u_t + \langle s_t, \operatorname{grad} u_t\rangle + \|\operatorname{grad} u_t\|^2\big)$ to g_t, so $\kappa_t = D_t$ is the unique choice that removes every second-order derivative of the tilt; this is why Table 4 lists no separate guidance scale for FMTT. The argument uses only the divergence product rule and never commutes a gradient with a Laplacian, so no curvature term appears: it is the intrinsic counterpart of the Euclidean derivation of (Skreta et al., 2025), not a new theorem. The proposition is an identity for the ideal continuous-time process with mutually consistent p_t, b_t and s_t; our experiments approximate it through a learned velocity and score, time discretization and a finite ensemble of walkers.

Remark E.2 (Locally Lipschitz tilts). The rewards we use are only locally Lipschitz (hard one-sided Chamfer energies, App. G.1). If u_t is locally bounded, absolutely continuous in time with locally integrable $\partial_t u_t$, and in $W^{1,\infty}_{\mathrm{loc}}$ in space for almost every t, the cancellation above holds distributionally by the Sobolev chain rule and equation E.9 holds weakly with the same g_t almost everywhere; since π_t is absolutely continuous with respect to Riemannian volume, the choice of representative on null sets is immaterial. We assume the guided martingale problem is well-posed for the resulting bounded but discontinuous drift. If $u_t(X) = \chi_t r(\Phi^\theta_{t,1}(X))$ with a C^1 map, the same holds by the weak chain rule. The hard-voxel energy of App. G.2 is discontinuous and is not covered; its FMTT implementation is a piecewise-smooth approximation.

Table 4: **Inference-time hyperparameters per domain and task.** λ_y is the reward strength in $r_y^N = -\lambda_y \ell_y^N$, selected for each method and task on validation data; ς is the process-noise amplitude of the FMTT reference SDE; K is the number of FMTT walkers.

Domain	Task	Method	λ_y	ς	K	NFE
ShapeNet	inpainting	Lookahead	2	—	1	32
		FMTT	32	0.05	4	32
	super-resolution	Lookahead	4	—	1	32
		FMTT	64	0.05	4	32
	voxel deconvolution	Lookahead	1	—	1	32
		FMTT	16	0.02	4	32
Single cell	cell-type inpainting	Lookahead	32	—	1	16
		FMTT	128	0.30	4	16
	super-resolution	Lookahead	32	—	1	16
		FMTT	128	0.15	4	16
	bulk deconvolution	Lookahead	32	—	1	16
		FMTT	256	0.15	4	16
Spatial niche	guided enrichment	Lookahead	8	—	1	8
		FMTT	32	0.10	8	8
	super-resolution	Lookahead	8	—	1	8
		FMTT	64	0.10	4	8

E.4 FMTT: THE REFERENCE PATH, EXACT RATES, AND APPROXIMATION

We take V_t to be the instantaneous velocity transporting the pretrained path and set

$$b_t = V_t + D_t s_t, \qquad \mathrm{d}X_t = [V_t + D_t s_t](X_t)\,\mathrm{d}t + \varsigma_t\,\mathrm{d}W_t^{\bar{g}_N}, \tag{E.11}$$

whose one-time marginals are p_t under the usual regularity assumptions (Song et al., 2021; De Bortoli et al., 2022; Hsu, 2002). This is the common-path requirement: V_t and s_t must be the velocity and score of the same p_t. Although the collapsed FMTT rate carries no explicit score after the cancellation in equation E.10, the score is still needed to simulate equation E.11. In our model $V_t(X) = U_{t,t}^\theta(X)$ is the diagonal field rather than a secant extracted from $\Phi_{t,1}^\theta$.

Let $\chi : [0,1] \to [0,1]$ be differentiable with $\chi_0 = 0$ and $\chi_1 = 1$, and let u_t^{FMTT} be the endpoint potential equation 4.3. Since $\Phi_{1,1}^\theta = \mathrm{Id}$, it satisfies $u_0 = 0$ and $u_1 = r_y^N$, so with its full log-weight rate it targets the terminal particle density $Z_y^{-1} p_1 e^{r_y^N}$ and, by equation E.4, the population law $\mathbb{P}_y^N$.

Exact FMTT rate. Write $F_t(X) := \Phi_{t,1}^\theta(X)$. Its guided drift is $b_t + D_t \chi_t (dF_t)^* \operatorname{grad}_{\bar{g}_N} r_y^N(F_t)$, where the adjoint pulls the endpoint reward gradient back to $T_X \mathcal{M}^N$. Define the backward-flow residual

$$\mathcal{R}_t^\theta(X) := \partial_t F_t(X) + dF_t(X)[V_t(X)] \in T_{F_t(X)}\mathcal{M}^N. \tag{E.12}$$

Applying the time and space chain rules to $u_t^{\mathrm{FMTT}}(X) = \chi_t r_y^N(F_t(X))$ in equation E.10 yields

$$\begin{aligned} g_t^{\mathrm{FMTT}}(X) = \chi_t'\, r_y^N(F_t(X)) \\ + \chi_t \big\langle \operatorname{grad}_{\bar{g}_N} r_y^N(F_t(X)), \mathcal{R}_t^\theta(X) \big\rangle_{\bar{g}_N}. \end{aligned} \tag{E.13}$$

Both vectors in the residual term lie at the predicted endpoint. The backward (Eulerian) flow identity makes $\mathcal{R}_t^\theta = 0$ for the exact flow of V_t; a learned map need not satisfy it.

Implemented collapsed rate. The reported FMTT implementation omits this residual term and uses

$$\tilde{g}_t^{\mathrm{FMTT}}(X) := \chi_t'\, r_y^N\big(\Phi_{t,1}^\theta(X)\big), \tag{E.14}$$

which is exact when the learned map satisfies the backward flow identity. With $\delta_t^\theta(X) := \|\mathcal{R}_t^\theta(X)\|_{\bar{g}_N}$, Cauchy–Schwarz gives the pointwise rate-error bound

$$|g_t^{\mathrm{FMTT}}(X) - \tilde{g}_t^{\mathrm{FMTT}}(X)| \le \chi_t \| \operatorname{grad}_{\bar{g}_N} r_y^N(F_t(X))\|_{\bar{g}_N}\, \delta_t^\theta(X). \tag{E.15}$$

Integrating the right-hand side along a common guided path bounds the corresponding log-weight error; it is not by itself a bound on terminal sampling error. For the locally Lipschitz hard-Chamfer

rewards the full rate is interpreted almost everywhere through the weak gradient and chain rule of Remark E.2. Exact targeting requires the full rate, a consistent reference velocity and score, and the stated regularity; the collapsed rate, learned fields, time discretization and finite walker ensemble introduce separate approximations.

E.5 Reference paths and score recovery

The corrector needs the score of the same reference marginal path as the learned probability-flow velocity. We use two path-specific constructions.

Euclidean linear–Gaussian path. For $X_t = \alpha_t X_0 + \beta_t X_1$ with $X_0 \sim \mathcal{N}(0, I)$ independent of X_1, velocity and score are related by

$$V_t(x) = \frac{\dot\beta_t}{\beta_t} x + \alpha_t\Big(\alpha_t \frac{\dot\beta_t}{\beta_t} - \dot\alpha_t\Big) s_t(x), \qquad \beta_t \neq 0, \tag{E.16}$$

since $s_t(x) = -\mathbb{E}[X_0 \mid X_t = x]/\alpha_t$ and $V_t(x) = \dot\alpha_t \mathbb{E}[X_0 \mid X_t = x] + \dot\beta_t \mathbb{E}[X_1 \mid X_t = x]$. At $\alpha_t = 1 - t$, $\beta_t = t$,

$$s_t(x) = \frac{tV_t(x) - x}{1 - t}, \qquad 0 < t < 1, \tag{E.17}$$

so a diagonal velocity error ϵ_t induces the score error $\widehat{s}_t - s_t = \frac{t}{1-t}\epsilon_t$, which is why we retain the diagonal anchor as a stabilizer near the terminal endpoint. This is the relation used for ShapeNet and niches.

Spherical reference path. The single-cell score uses the coupling employed in training: source particles i.i.d. uniform on $\mathbb{S}^{d-1}$, the source cloud independent of the endpoint cloud, and a fixed correspondence. A data-dependent or OT coupling would contribute an additional source-conditional term, so the formula should not be reused unchanged for such a path. Let x_0 be uniform on $\mathbb{S}^{d-1}$, independent of x_1, and $x_t = \mathrm{slerp}(x_0, x_1, t)$. If $\vartheta = d_g(x_0, x_1)$ and $\phi = d_g(x_t, x_1)$ then $\phi = (1 - t)\vartheta$. Write $\omega_k := |\mathbb{S}^k|$.

Proposition E.3 (Conditional spherical density and score)**.** *For $d \geq 3$, $0 \leq t < 1$ and $0 < \phi < (1 - t)\pi$, the conditional density relative to Riemannian volume is*

$$\rho_t(x \mid x_1) = \frac{1}{\omega_{d-1}(1 - t)} \Big[\frac{\sin(\phi/(1 - t))}{\sin \phi}\Big]^{d-2}, \tag{E.18}$$

zero outside the cap $\phi \leq (1 - t)\pi$, with interior score

$$\nabla_g \log \rho_t(x \mid x_1) = (d - 2)\Big[\frac{1}{1 - t} \cot\Big(\frac{\phi}{1 - t}\Big) - \cot \phi\Big]\widehat{n}, \tag{E.19}$$

where $\widehat{n}$ points in the direction of increasing distance from x_1.

Proof. For fixed x_1 the uniform-source angle has density proportional to $\sin^{d-2} \vartheta$. The change of variables $\phi = (1 - t)\vartheta$ gives radial density $\propto (1 - t)^{-1} \sin^{d-2}(\phi/(1 - t))$; dividing by the spherical shell area $\omega_{d-2} \sin^{d-2} \phi$ yields equation E.18, and differentiating its logarithm gives equation E.19. □

Whenever differentiation can be interchanged with mixing over endpoints, $\mathrm{grad}_{\bar g_N} \log p_t(X) = \mathbb{E}[\mathrm{grad}_{\bar g_N} \log p_t(X \mid X_1) \mid X_t = X]$, so conditional score regression estimates the marginal score of this interpolation path (Hyvärinen, 2005; Vincent, 2011; De Bortoli et al., 2022). The conditional score diverges at the moving cap boundary; the conditional Fisher information is finite exactly for $d \geq 4$ and scales as $(1 - t)^{-1}$ in the bulk. With $d = 128$ and training restricted to $t \leq 1 - \varepsilon$ neither is binding, so the $(1 - t)^2$ time weight of App. D.4 is a stabilization choice rather than an integrability requirement.

F Experimental details

This appendix specifies the data, models, optimization, fine-tuning protocol, metrics, and the observation operator and reward of every task.

Table 5: **Architecture and parameter counts per domain.** All models share the permutation-equivariant set transformer of App. F.1; W/L/H/MLP are width, depth, number of heads and MLP width. "Cond type." is how the conditional inputs enters each block: adaptive layer norm (adaLN) or an additive shift; d is the dimension of $\mathcal{M}$ per point. "Cond input." lists conditioning inputs; the niche models are unconditional.

Domain	Model	W/L/H/MLP	Cond type.	d	Cond input.	Params
ShapeNet	Flow map	512/6/4/512	additive	3	category	10.0M
	WFM	512/6/4/512	adaLN	3	category	25.5M
Single cell	Flow map	512/6/4/512	adaLN	128	tissue+disease+per-cell type centroid	25.6M
	RWEFM	512/6/4/512	adaLN	128	tissue+disease+per-cell type centroid	25.6M
Spatial niche	Flow map	512/6/4/512	additive	16	—	10.0M
	WFM	512/6/4/512	adaLN	16	—	22.3M

Table 6: **Optimization settings per domain.** Flow maps: Adam, cosine schedule decaying to $0.1\times$ the peak learning rate after 2000 warmup steps, gradient-norm clipping at 1.0, parameter EMA with decay 0.999, no dropout. WFM/RWEFM: the original WFM recipe, exponential decay by $\gamma = 0.99$ every Δ steps (`decay_steps`), no warmup, clipping or EMA, dropout 0.1. Batch size counts populations; the per-population loss averages over points. Wall-clock is total training time on a single NVIDIA B200 GPU.

Domain	Model	LR	Schedule	Batch	Steps	Wall-clock (h)
ShapeNet	Flow map	3×10^{-4}	cosine	64	500k	22–35
	WFM	3×10^{-4}	exp (Δ=5000)	64	500k	55
Single cell	Flow map	10^{-4}	cosine	32	250k	10–14
	RWEFM	2×10^{-4}	exp (Δ=2500)	32	250k	12
Spatial niche	Flow map	3×10^{-4}	cosine	64	100k	0.3–0.6
	WFM	2×10^{-4}	exp (Δ=1000)	64	500k	3

F.1 Model architecture and optimization

Backbone. All models share a permutation-equivariant DiT-style set transformer: per-point linear embedding, L pre-norm blocks of full self-attention over points followed by a two-layer MLP, and a per-point output head. No positional encoding or point ordering enters anywhere, which is what makes the network equivariant; all population operations are masked means or softmax attention, so a checkpoint trained at one particle count can be evaluated at another. Table 5 lists per-task architecture, conditioning and parameter counts. All training and evaluation runs on a single NVIDIA B200 GPU per job.

Optimization. All models use Adam with batch counted in clouds (the per-cloud loss averages over points). Flow maps use a cosine schedule decaying to $0.1\times$ the peak, 2000 warmup steps, global gradient-norm clipping at 1.0, an EMA of the parameters (decay 0.999) and no dropout. WFM/RWEFM follow the original WFM recipe: exponential decay ($\gamma = 0.99$ per `decay_steps`), no warmup, no clipping, no EMA, dropout 0.1. Table 6 gives per-task values.

Optimal-transport matching. WFM and RWEFM recompute an entropic optimal-transport plan between the source and target populations at every training step and regress on the resulting displacements. On ShapeNet and the spatial niche the assignment is sampled from the plan, with $\varepsilon = 0.002$ and Sinkhorn run to its convergence threshold; on single-cell data it is the barycentric projection, with $\varepsilon = 0.001$ and the iteration count fixed once at initialization. Flow maps use independent coupling and solve no transport problem during training.

The two times s, t in flow maps. Both times pass through the same Fourier feature ladder and are then concatenated and projected once to a single embedding, $e_{s,t} = W[\phi(s); \phi(t)]$. The ladder uses linearly spaced frequencies $k \cdot 2\pi/d \cdot \kappa$; the scale $\kappa < 1$ lowers the maximum frequency and thereby bounds ∂_t of the network, which stabilizes the differential objectives. We use $\kappa = 1.0$ for semigroup, 0.25 for Lagrangian and 0.1 for Eulerian on ShapeNet and niches, and $\kappa = 0.1$ for all three on single cell.

F.2 METRIC DEFINITIONS

Our metrics compare objects at three levels. *Set-level* metrics (Sec. F.2.1) compare a set of generated clouds to a set of real clouds and score unconditional generation. *Instance-level* metrics (Sec. F.2.2) compare one reconstruction to its ground-truth cloud and score every inverse task. *Diagnostics* (Sec. F.2.3) probe properties the instance-level metrics are insensitive to: fine structure, composition and over-separation. All three build on two distances between clouds, which we fix first.

Cloud distances. For clouds X, Y the Chamfer distance is $\mathrm{CD}(X,Y) = \mathbb{E}_x \min_y \|x-y\|^p + \mathbb{E}_y \min_x \|x-y\|^p$, and the earth-mover distance (EMD) is approximated by an entropic-Sinkhorn OT cost, so every EMD column is an approximation. On the sphere the Euclidean norm is replaced by the geodesic distance.

F.2.1 SET-LEVEL METRICS (UNCONDITIONAL GENERATION)

Each observation is an entire cloud and we generate as many clouds as there are reference clouds, following the point-cloud evaluation of (Yang et al., 2019; Zhou et al., 2021) as used by WFM (Haviv et al., 2025). Both metrics below are computed once under CD and once under EMD.

1-NN deviation. For equally sized sets of real and generated clouds, let a_{real} and a_{gen} be the classwise leave-one-out nearest-neighbour accuracies in the pooled set. We report

$$\text{1-NN-D} := \tfrac{1}{2}\big(|a_{\mathrm{real}} - \tfrac{1}{2}| + |a_{\mathrm{gen}} - \tfrac{1}{2}|\big) \in [0, \tfrac{1}{2}], \tag{F.1}$$

minimized at 0 when both classwise accuracies equal $\frac{1}{2}$; taking absolute deviations before averaging prevents opposite classwise deviations from cancelling into a spuriously good pooled score.

Cloud-kernel MMD. With a base cloud distance $d \in \{\mathrm{CD}, \mathrm{EMD}\}$,

$$\mathrm{MMD}^2(\mu,\nu) = \mathbb{E}_{\mu\otimes\mu}[k(x,x')] + \mathbb{E}_{\nu\otimes\nu}[k(y,y')] - 2\,\mathbb{E}_{\mu\otimes\nu}[k(x,y)], \qquad k(x,y) = e^{-d(x,y)/\sigma}, \tag{F.2}$$

with $\sigma = 0.1$, where x, y range over clouds. Set-level statistics compare distributions of clouds; they do not certify that any single generated population is realistic, which is why the inverse tasks are scored per instance.

F.2.2 INSTANCE-LEVEL METRICS (INVERSE TASKS)

Every inverse task reports two numbers between a reconstruction $\hat{X}$ and its ground truth X: an MMD between the two point sets and the Chamfer distance. The only task dependence is *which points enter the comparison*.

Point-kernel MMD. $\sqrt{\max(\widehat{\mathrm{MMD}}^2, 0)}$ in the form of equation F.2 but with x, y ranging over the points of the two clouds and a Gaussian kernel $k(x,y) = e^{-\|x-y\|^2/(2\sigma^2)}$, with the median-heuristic bandwidth $\sigma = \sqrt{\mathrm{median}(\|\cdot\|^2)/2}$ on the pooled pairwise squared distances of the two sets compared. This is the MMD in every inverse-task table.

Which points are compared. For ShapeNet inpainting, each reconstructed point is assigned to the removed region if its nearest ground truth point lies there. MMD and CD are then computed between the points so assigned and the true removed points. For niche super-resolution, MMD is computed on the whole niche and CD on the missing region, so MMD measures global plausibility and CD measures how well the gap is filled. For cell-type inpainting, masked MMD compares the regenerated cells with the true cells of the held-out type, and for population super-resolution it compares the regenerated cells with the true missing cells. For bulk deconvolution nothing is observed at the point level, so MMD is computed on the whole cloud.

F.2.3 DIAGNOSTICS

CD and MMD are driven by coarse shape, so we add diagnostics for three properties they barely register. Each is defined once; what differs across domains is only how a cell is assigned a type.

Fine structure: within-voxel spread (point clouds). Voxel deconvolution reports one average per cell, so how points are arranged *inside* a cell is never observed. For a cloud X, bin the points into an $R \times R \times R$ grid over X's own bounding box and average, over the cells holding at least two points, the RMS distance from a cell's points to their own mean:

$$\mathrm{WV}_R(X) := \frac{1}{|\mathcal{V}_X^{\geq 1}|} \sum_{v \in \mathcal{V}_X^{\geq 1}} \Big(\frac{1}{n_v(X)} \sum_{i:\, x_i \in v} \|x_i - m_v(X)\|_2^2 \Big)^{1/2}, \tag{F.3}$$

reported against the ground-truth value.

Type assignment. On single cell, a cell's type is the arg-max cosine similarity to the L2-normalized type centroids. On niches, types are assigned by a kNN classifier ($k = 15$, distance-weighted) fit on the atlas in raw PCA-16 space; for real niches we use the atlas labels of the member cells. Everything below is computed from these labels.

Composition. The *composition* of a cloud is its normalized type histogram π. We report *composition Pearson*, the correlation between reconstructed and true compositions (single cell); *composition TV*, $\frac{1}{2}\sum_t |\hat{\pi}_t - \pi_t|$ (niches); and, for guided enrichment, the *enrichment gap* $|\hat{\pi}_{t^\star} - p_{t^\star}|$ between the generated abundance of the target type $t^\star$ and the requested abundance $p_{t^\star}$.

Coverage. *Cell-type accuracy* is exact-match agreement on the held-out type in cell-type inpainting. *Rare-type recall* is the fraction of types present in the ground truth but absent from the observation that appear in the reconstruction, the types a trivial densifier can never produce.

Over-separation: clustering ARI. The adjusted Rand index between centroid labels and k-means labels (k = number of distinct centroid labels) on the same cloud. ARI is not monotone: values above the ground-truth reference indicate over-separated clusters, so it is reported as an absolute Δ against the ground-truth value.

F.3 Cell-type conditioning and the composition inferrer.

Conditioning tokens. Every cell slot of the single-cell models receives the atlas centroid of one cell type, together with tissue and disease labels (Table 5). Centroids are global per-type means over the SCimilarity atlas, so they carry no patient-specific information and act purely as type-identity tokens; patient-specific placement is left to the generator and, in inverse tasks, to guidance. The ground-truth composition supplies these tokens during training and in the unconditional generation of Table 1, so the single-cell generation reported there is conditional on composition, for the flow maps and for RWEFM alike.

Composition inferrer. In inverse tasks the ground-truth composition is unavailable, and arms that require composition tokens obtain them from an inferrer f_ϕ. Both variants share one architecture: a 3-layer, 4-head transformer of width 128 over the $V = 203$ type tokens, each carrying its centroid, with tissue (17) and disease (11) embeddings. The *bulk* inferrer (0.46M parameters) maps the mean embedding y to a composition over the 203 types, so that self-attention over centroid-carrying tokens becomes a learned analogue of the mixture fit $y \approx \sum_c \pi_c M[c]$ that NNLS solves in closed form, with a softmax head in place of the hard zeros of NNLS. The *cell-type* inferrer (0.43M) maps the type histogram of the observed cells to the composition of the unobserved ones. It is trained on a mixture of masking regimes (random per-type subsampling, removal of whole types, and partial under-sampling of a single type) rather than on the operator used at test time, so that it recovers types absent from the observation instead of echoing its input. Two instances appear across the arms, one trained on random subsampling alone and one on the full mixture; the inpainting arms that use external tokens additionally take a held-out-type predictor of the same architecture.

Training and use at inference. All inferrers are trained on the training patients only, for 8000 steps at batch 128 with AdamW (learning rate 3×10^{-4}, weight decay 10^{-4}, gradient-norm clipping 1.0) under a soft cross-entropy against the target composition, in under ten minutes on a single NVIDIA B200 GPU. At inference the predicted composition is converted to N centroid tokens for the generator: the training-free arms consume these tokens directly, whereas the fine-tuned WFM

arms estimate composition from their own conditioning. No held-out cell, and no quantity computed from held-out cells, enters an inferrer or the generator; the inferrer is the only trained component specific to an observation type, and the generator is never retrained.

F.4 Task-specific fine-tuning (FT-WFM)

Every fine-tune starts from the base WFM by a partial parameter merge: shared backbone weights are loaded and freshly introduced conditioning parameters are randomly initialized. We train each for 100k steps at batch size 64 and learning rate 5×10^{-5} with the same exponential decay as base training. The flow-matching loss is restricted to the masked slots (observed slots contribute zero loss gradient, and their OT weights are set to 10^{-10} before renormalization). Mask strategies are task-matched, for instance `random` with reveal fraction $\sim \mathcal{U}[0.0625, 0.25]$ for super-resolution, `celltype` (hold out one whole type) for cell-type inpainting, and extra pseudobulk conditioning for deconvolution.

G Inverse Problem Details

G.1 Noiseless observation models and reward tilts

Observation protocol. All inverse observations are generated deterministically from a clean held-out target $X^\star$, $y = \mathcal{O}(X^\star)$, where $\mathcal{O}$ is the task-specific masking, subsampling, aggregation or voxel-summary operation. We do not add an independent measurement-noise variable: for these tasks, adding coordinate noise after deleting cells or points would define a different corruption problem, so the operators themselves define the observations. The inference target is the Gibbs tilt equation E.3, which we do not interpret as Bayesian posterior sampling under a known sensor-noise model. Noiseless data consistency is common in generative inverse problems: RePaint conditions a pretrained diffusion model directly on a known image region (Lugmayr et al., 2022), and DDNM formulates its basic linear restoration problem as $y = Ax$ before treating noisy measurements separately (Wang et al., 2023); other methods study noisy likelihood models (Kawar et al., 2022; Chung et al., 2023), and we cite them as neighboring work rather than as the observation model used here.

Partial-set rewards. For super-resolution and inpainting the observation mechanism is point selection: uniform subsampling and structured deletion respectively. We use the one-sided Chamfer energy for both inference and reporting; its asymmetry requires the candidate population to explain every observed point without penalizing plausible mass in the unobserved region. As a finite minimum of smooth squared-distance functions it is continuous and locally Lipschitz, hence differentiable almost everywhere by Rademacher's theorem, and Remark E.2 covers the resulting weak formulation; at differentiability points its gradient is that of the active nearest-neighbour assignment. The Gibbs strength λ_y multiplies the hard discrepancy directly.

Reward strength. Any finite $\lambda_y \geq 0$ defines a valid Gibbs tilt for the nonnegative energies used here, but large values are not numerically benign. In deterministic steering, λ_y multiplies the gradient and therefore trades off directly with the numerical step size, so the two are not separately identifiable. In FMTT, by contrast, λ_y changes the terminal target itself: larger values produce sharper tilts but more variable importance weights, lower effective sample size and stiffer guided dynamics. We select finite λ_y on validation data; for a fixed task, lookahead and FMTT use the same reward with strengths selected separately on validation data. The Chamfer energy constrains at most M of the N points, so its effective strength depends on M/N and a value tuned at one particle count does not transfer to another; at large λ_y it is minimized by placing a few points on the observations while the rest is governed by the prior alone, the concrete form of the singular zero-temperature limit.

G.2 SHAPENET INVERSE PROBLEMS

Each shape is a Euclidean population $X = (x_1, \dots, x_N) \in (\mathbb{R}^3)^N$. For a partial cloud $Y = (y_1, \dots, y_M)$ define the one-sided Chamfer discrepancy

$$D_{\to}(Y, X) := \frac{1}{M} \sum_j \min_i \|y_j - x_i\|_2^2, \tag{G.1}$$

used both for inference and reporting.

Blob and planar inpainting. The blob mask removes a contiguous neighbourhood grown around random seed points; the planar mask cuts the cloud with an axis-aligned plane along a random axis at the median and removes one side. Both remove 50% of the points. The reward is $r^{\mathrm{SN}}_{\mathrm{inp}}(X; Y) = -\lambda_{\mathrm{inp}} D_{\to}(Y, X)$.

Super-resolution. Uniformly retain a sparse fraction of the points (6.25% observed) and use $r^{\mathrm{SN}}_{\mathrm{SR}}(X; Y) = -\lambda_{\mathrm{SR}} D_{\to}(Y, X)$.

Voxel deconvolution. Partition the normalized ambient cube into an $R \times R \times R$ voxel grid $\mathcal{V}$. For a voxel v containing candidate points, define the hard voxel mean $m_v(X) := n_v(X)^{-1} \sum_{i: x_i \in v} x_i$ with $n_v(X) := \#\{i : x_i \in v\}$, and the observation $y_v = m_v(X^\star)$. Let $\mathcal{V}_y$ be the voxels occupied by the clean observation and use the full-target penalty when an observed voxel is empty:

$$\begin{aligned} e_v(X; y_v) &:= \begin{cases} \|m_v(X) - y_v\|_2^2, & n_v(X) > 0, \\ \|y_v\|_2^2, & n_v(X) = 0, \end{cases} \\ \ell_{\mathrm{vox}}(X; y) &:= \frac{1}{2|\mathcal{V}_y|} \sum_{v \in \mathcal{V}_y} e_v(X; y_v), \qquad r^{\mathrm{SN}}_{\mathrm{deconv}} = -\lambda_{\mathrm{vox}} \ell_{\mathrm{vox}}. \end{aligned} \tag{G.2}$$

Within a region of particle space where all voxel assignments and occupancies are fixed the reward is smooth, but across voxel boundaries $m_v(X)$ or the occupancy branch can jump, so the reward is generally discontinuous and is not covered by Prop. E.1 or Remark E.2; the reported hard-voxel FMTT runs use the within-cell gradients as a piecewise-smooth approximation and are not claimed to be exactly target-correct. In practice we assign candidate points to voxel slots by a balanced Hungarian matching on stop-gradient positions at each step, which removes the within-voxel smearing that a fixed-order assignment produces; under this matching the reported runs minimize the equivalent per-point energy $\frac{1}{2|\mathcal{V}_y|} \sum_i \|x_i - y_{v(i)}\|^2$, where $v(i)$ is the voxel assigned to point i, which shares minimizers with equation G.2 when each occupied voxel holds its balanced share of points.

G.3 SINGLE-CELL AND SPATIAL-NICHE INVERSE PROBLEMS

Each patient is $X = (x_1, \dots, x_N)$ with $x_i \in \mathbb{S}^{127}$. For a partial observation $Y = (y_1, \dots, y_M)$ the one-sided *chordal* discrepancy is

$$D^{\mathrm{cell}}_{\to}(Y, X) := \frac{1}{M} \sum_j \min_i \|y_j - x_i\|_2^2 = \frac{1}{M} \sum_j \min_i \big(2 - 2\langle y_j, x_i \rangle\big). \tag{G.3}$$

For unit vectors chordal and geodesic distances induce the same nearest-neighbour ordering, and the chordal form is what the implementation evaluates; it is continuous and locally Lipschitz in X, with gradients through the active assignments in the sense of Remark E.2.

Cell-type inpainting. Remove all cells of one selected type and use $r^{N}_{\mathrm{inp}}(X; Y) = -\lambda_{\mathrm{inp}} D^{\mathrm{cell}}_{\to}(Y, X)$. The identity of the held-out type is never supplied. The composition inferrer of App. F.3 predicts it from the type histogram of the surviving cells, and that prediction provides the centroid tokens of the regenerated slots. The reward is the one-sided discrepancy from the observed cells to the whole candidate population; no candidate cell is masked out before it is scored.

Population super-resolution. Uniformly retain a sparse subset of M cells and use $r^{N}_{\mathrm{SR}}(X; Y) = -\lambda_{\mathrm{SR}} D^{\mathrm{cell}}_{\to}(Y, X)$; the inferrer predicts the full composition from the histogram of the M observed cells. The same reward form is used for spatial-niche super-resolution with $x_i \in \mathbb{R}^{16}$ and the Euclidean squared distance.

Bulk-to-single deconvolution. The observation is the exact ambient mean embedding $y = \mathcal{A}_{\text{bulk}}(X^\star) = N^{-1}\sum_i x_i^\star$, from which the inferrer predicts the composition that conditions every arm and we use the quadratic energy and reward

$$\ell_{\text{bulk}}(X; y) := \tfrac{1}{2}\left\| \tfrac{1}{N}\sum_i x_i - y \right\|_2^2, \qquad r^N_{\text{deconv}}(X; y) := -\lambda_{\text{bulk}}\ell_{\text{bulk}}(X; y). \tag{G.4}$$

The candidate mean is not renormalized onto the sphere, so the forward operator is linear; λ_{bulk} controls the tolerated aggregate mismatch and is not a measurement-noise precision. Niche deconvolution uses the same energy with the niche pseudobulk in $\mathbb{R}^{16}$.

Guided enrichment (spatial niche). The reward is a target on a learned summary rather than a data-consistency term. Let f_ψ be a per-cell type classifier trained on the atlas and evaluated in the unscaled 16-dimensional PCA space (the $[-1, 1]$ training rescale is inverted before applying it), and let $\hat\pi(X) = N^{-1}\sum_i \text{softmax}(f_\psi(x_i))$ be the induced 22-way composition. Given a target type $t^\star$ and requested abundance $p_{t^\star} \in [0, 1]$,

$$\ell_{\text{comp}}(X; t^\star, p_{t^\star}) := \big(\hat\pi_{t^\star}(X) - p_{t^\star}\big)^2, \qquad r_{\text{comp}} = -\lambda_{\text{comp}}\ell_{\text{comp}}; \tag{G.5}$$

only the target coordinate is constrained and the remaining 21 types are left to the prior.

G.4 Guidance settings and hyperparameters

Shared conventions. All inference-time methods for a fixed task use the same terminal reward $r^N_y = -\lambda_y \ell^N_y$; the strength λ_y is selected separately for each method on validation data; λ_y is distinct from both the numerical step size and the process-noise amplitude ς_t. Reported values refer to rewards differentiated in $\bar g_N$. The numerical scale of guidance depends on the task-specific energy and its normalization: partial-set energies average over observed points, voxel energies over observed voxel summaries, and bulk energies penalize the population mean. Consequently, the same numerical value of λ_y need not produce comparable guidance across tasks or population sizes. We select λ_y on validation data for each task.

FMTT. It runs a Riemannian Euler–Maruyama scheme over a uniform grid with process noise $\varsigma_t^2 = \varsigma^2(1 - t)$ and full guided drift $V_t + D_t s_t + D_t \,\text{grad}_{\bar g_N} u_t$. The annealing schedule is $\chi_t = t^\beta$ with $\beta = 1$ in every reported run, so $\chi_0 = 0$ and $\chi_1 = 1$.FMTT differentiates the reward through the flow map $\Phi^\theta_{t,1}$ and accumulates the collapsed rate equation E.14, so each of the K walkers carries a log-weight

$$\log \omega_k \;=\; \int_0^1 r^N_y\big(\Phi^\theta_{s,1}(X^k_s)\big)\,\mathrm{d}s \;\approx\; \sum_i r^N_y\big(\Phi^\theta_{t_i,1}(X^k_{t_i})\big)\,\Delta t_i, \tag{G.6}$$

The walkers are initialized from p_0 with zero log-weight and evolve independently. We take the *sampling* form of FMTT, where the returned sample is drawn as $k \sim \text{Categorical}(w)$ with $w_k = \omega_k / \sum_j \omega_j$.On the sphere the reward gradient and the Brownian increment are both tangent-projected before a single exponential-map step.

H Additional Experiments

This appendix collects results not shown in the main text: per-category breakdowns and the super-resolution and voxel deconvolution tasks on ShapeNet (App. H.1), cell-type inpainting and population super-resolution on single cell (App. H.2), qualitative reconstructions on spatial niches (App. H.3), and the generation-quality sweeps over the number of function evaluations behind Table 1 (App. H.4).

H.1 ShapeNet

Inpainting. Table 7 breaks the pooled ShapeNet inpainting result of Table 2 down by mask and category. FT-WFM has the lowest MMD in all six cells and the lowest CD in five, the exception being airplanes under the blob mask, where lookahead is marginally ahead (0.021 against 0.023). Lookahead is the strongest training-free arm in MMD on four of the six cells and in CD on all of them; FMTT is ahead in MMD on airplanes and chairs under the planar mask (0.251 against 0.263

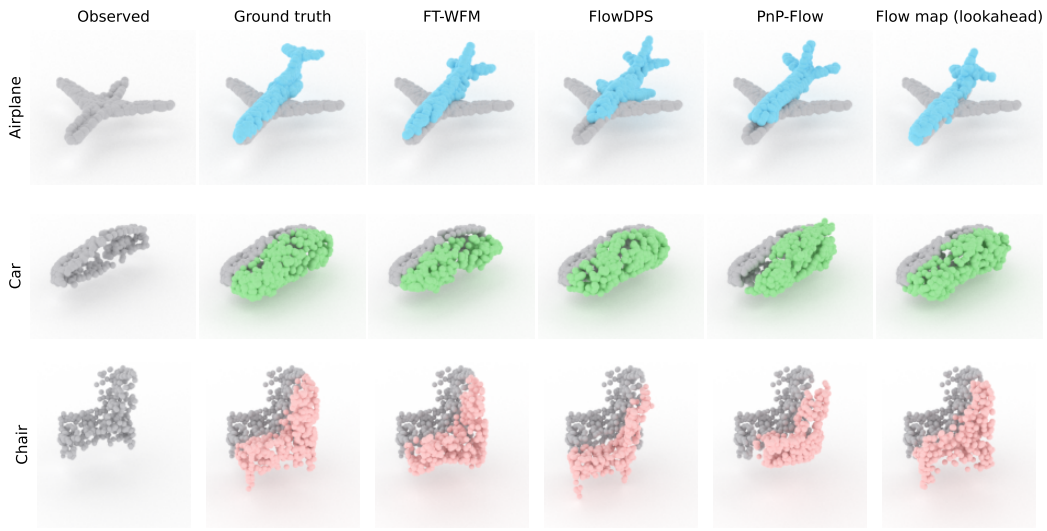


Figure 5: **ShapeNet inpainting under the planar mask.** One held-out object per category (airplane, car, chair) with 50% of the points removed by planar cut (App. G.2). Columns: observed points, ground truth, then reconstructions by FT-WFM, FlowDPS and PnP-Flow on the WFM prior and by lookahead on the frozen flow map. Quantitative results in Table 7.

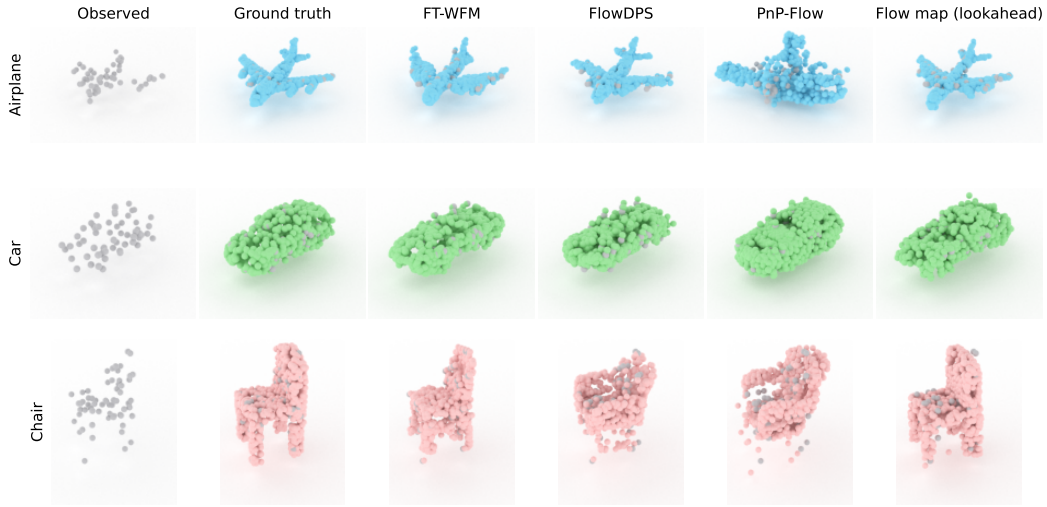


Figure 6: **ShapeNet super-resolution.** Same objects and column layout as Figure 5. The observation is 6.25% (128 of 2048) of the points, retained uniformly at random; every method regenerates the rest. Quantitative results in Table 8.

and 0.247 against 0.266). Both flow-map arms beat FlowDPS and PnP-Flow on every cell. The planar mask is the harder of the two for the flow-map arms: their gap to FT-WFM widens on every category, from at most 0.05 under the blob mask to up to 0.09 under the planar mask.

Super-resolution. We keep a uniformly random 6.25% of the 2048 points (128 observed) and ask each method to produce the points that were dropped; both columns of Table 8 score only those missing points against the true ones. The fine-tuned FT-WFM is best in every category on both metrics (category means of 0.096 masked MMD and 0.026 masked CD). Among the training-free arms, FMTT has the lowest masked MMD on airplanes and chairs and the lowest category mean (0.119 against 0.131 for lookahead), while lookahead is ahead on cars and has the lower masked CD in every category (0.033 against 0.036). Both flow-map arms are clearly ahead of the training-free

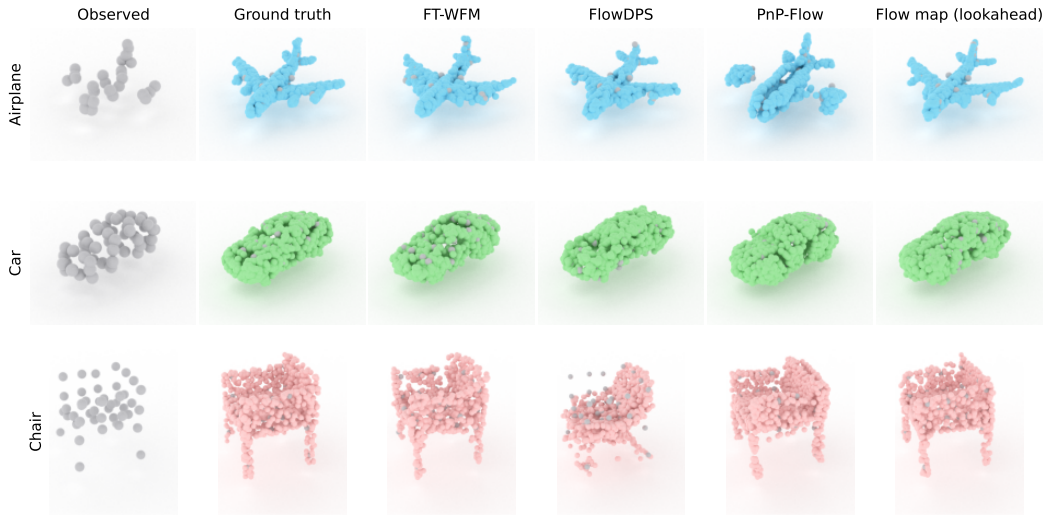


Figure 7: **ShapeNet voxel deconvolution.** Same objects and column layout as Figure 5. The observation is the mean position of the points in each occupied cell of an $R \times R \times R$ voxel grid (Eq. G.2); the observed-points column shows these voxel means, and all 2048 points are regenerated. Quantitative results in Table 9.

Table 7: **ShapeNet inpainting, per category.** Per-category breakdown of the pooled result in Table 2. 50% of the points are removed either by a blob mask (contiguous neighbourhoods grown around random seeds) or by a planar mask (one side of an axis-aligned cut at the median). FT-WFM is the WFM fine-tuned for this task (App. F.4); FlowDPS and PnP-Flow run on the frozen WFM; Lookahead and FMTT run on the frozen flow map. NFE is 100 for the WFM arms and 32 for the flow-map arms. Bold marks the best frozen-prior method.

		Blob mask						Planar mask					
		Airplane		Car		Chair		Airplane		Car		Chair	
Prior	Method	MMD↓	CD↓	MMD↓	CD↓	MMD↓	CD↓	MMD↓	CD↓	MMD↓	CD↓	MMD↓	CD↓
WFM	FT-WFM	0.215	0.023	0.146	0.027	0.182	0.037	0.217	0.028	0.117	0.027	0.179	0.040
	FlowDPS	0.282	0.031	0.186	0.034	0.330	0.068	0.283	0.041	0.185	0.036	0.332	0.072
	PnP-Flow	0.404	0.052	0.228	0.044	0.311	0.066	0.408	0.064	0.201	0.040	0.318	0.074
Flow map	Lookahead	**0.218**	**0.021**	**0.148**	**0.028**	**0.234**	**0.038**	0.263	**0.029**	**0.153**	**0.030**	0.266	**0.047**
	FMTT	0.226	0.025	0.151	0.029	0.236	0.049	**0.251**	0.033	0.164	0.033	**0.247**	0.053

solvers on the velocity prior, FlowDPS (0.159 / 0.038) and PnP-Flow (0.182 / 0.053), at 32 rather than 100 function evaluations. Reconstructions of this task tend to have the right shape but sit slightly off-centre. Masked MMD is very sensitive to such a global offset while CD is not, so we apply one post-hoc translation before scoring: shift the whole reconstruction so that its centroid coincides with the centroid of the observed points. This is legitimate because the points are retained uniformly at random, so the observed centroid is an unbiased estimate of the true centroid of the full cloud; the step reads only the observation and never the held-out points. It is applied identically to every method, and it does not change the ordering of the arms.

Voxel deconvolution. Table 9 reports the results. As the most naive baseline we add a bootstrap control, which places n_v points in each occupied cell at the reported mean y_v and jitters them with isotropic Gaussian noise of scale 0.02; it uses only what the measurement provides, with no prior and no model, so it marks what is reachable without generative modeling. On the $R = 4$ grid FMTT has the lowest MMD in every category (category mean 0.011), while FT-WFM, FlowDPS, lookahead and the bootstrap control are practically tied between 0.013 and 0.015 and PnP-Flow trails far behind (0.10). The coarser $R = 3$ grid is the discriminative setting: there FMTT (0.020) and lookahead (0.024) separate from FT-WFM (0.026), FlowDPS (0.028) and the bootstrap control (0.031). CD orders the arms the same way but compresses the gaps. Within-voxel spread, the fine structure the measurement never sees, favours the fine-tuned model: FT-WFM is closest to the ground-truth spread in every category cell, and the flow-map arms sit at roughly one and a half to two times its gap,

Table 8: **ShapeNet super-resolution, per category.** 6.25% of the 2048 points (128) are retained uniformly at random and the remaining 1920 are regenerated. MMD and CD score only the missing points against the true ones, after the observed-centroid translation of App. H.1, which reads only the observation and is applied identically to every method. Best per column in bold.

		Airplane		Car		Chair	
Prior	Method	MMD ↓	CD ↓	MMD ↓	CD ↓	MMD ↓	CD ↓
WFM	FT-WFM	0.108	0.020	0.085	0.027	0.094	0.032
	FlowDPS	0.185	0.029	0.118	0.031	0.175	0.054
	PnP-Flow	0.236	0.050	0.127	0.038	0.182	0.070
Flow map	Lookahead	0.155	0.025	**0.094**	**0.030**	0.143	0.045
	FMTT	**0.135**	**0.028**	0.100	0.031	**0.121**	**0.048**

Table 9: **ShapeNet voxel deconvolution, per category.** The observation is the mean of the points in each occupied cell of an $R \times R \times R$ grid, $R \in \{3, 4\}$, and all 2048 points are regenerated. MMD and CD are computed on the whole cloud. $\Delta\mathrm{wv} = |\mathrm{wv}_R(\hat{X}) - \mathrm{wv}_R(X^\star)|$ is the gap to the ground-truth within-voxel spread (Eq. F.3), with references $\mathrm{wv}_3(X^\star) = 0.075$ and $\mathrm{wv}_4(X^\star) = 0.054$. Bootstrap control places n_v points at each observed voxel mean with isotropic Gaussian jitter of scale 0.02 and uses no prior. We apply the terminal range projection of Eq. H.1 to all methods but FT-WFM. Bold marks the best frozen-prior method.

			Airplane			Car			Chair		
R	Prior	Method	MMD↓	CD↓	Δwv ↓	MMD↓	CD↓	Δwv ↓	MMD↓	CD↓	Δwv ↓
3^3	—	Bootstrap (control)	0.0454	0.0234	0.0308	0.0253	0.0304	0.0441	0.0218	0.0322	0.0427
	WFM	FT-WFM	0.0340	0.0190	0.0094	0.0183	0.0253	0.0093	0.0268	0.0330	0.0040
		FlowDPS	0.0415	0.0231	0.0281	0.0201	**0.0244**	0.0197	0.0212	0.0331	0.0146
		PnP-Flow	0.1799	0.0543	0.0746	0.0820	0.0505	0.0697	0.1538	0.0742	0.0831
	Flow map	Lookahead	0.0361	0.0207	**0.0116**	0.0159	0.0255	**0.0142**	0.0213	0.0316	**0.0076**
		FMTT	**0.0321**	**0.0194**	0.0132	**0.0138**	0.0248	0.0153	**0.0152**	**0.0302**	0.0133
4^3	—	Bootstrap (control)	0.0198	0.0193	0.0152	0.0127	0.0246	0.0219	0.0112	0.0265	0.0218
	WFM	FT-WFM	0.0191	0.0174	0.0072	0.0099	0.0227	0.0053	0.0140	0.0292	0.0044
		FlowDPS	0.0190	0.0184	0.0124	0.0107	**0.0226**	0.0094	0.0145	0.0291	0.0055
		PnP-Flow	0.1329	0.0452	0.0559	0.0585	0.0464	0.0538	0.1190	0.0665	0.0640
	Flow map	Lookahead	0.0197	0.0182	**0.0087**	0.0093	0.0238	**0.0093**	0.0113	0.0279	**0.0059**
		FMTT	**0.0164**	**0.0174**	0.0091	**0.0085**	0.0231	0.0124	**0.0085**	**0.0267**	0.0092

since the terminal projection fixes each voxel mean but not the arrangement inside the voxel. Each training-free run ends with one step that makes the reconstruction match the measurement exactly, the deconvolution counterpart of the observed-centroid correction used for super-resolution. Match points to cells by a balanced Hungarian assignment (n_v points to cell v, minimising total squared distance to y_v), then move each cell's group by the single vector that puts their mean on y_v:

$$\hat{X} \;\longmapsto\; \hat{X} + \delta, \qquad \delta_i = y_{v(i)} - m_{v(i)}(\hat{X}). \tag{H.1}$$

H.2 Single Cell

Cell-type inpainting. All cells of one type are removed and must be regenerated without the identity of the missing type being supplied. Table 10 reports masked MMD on the held-out type and cell-type accuracy over $n = 100$ patients; Figure 8 shows two examples. Lookahead on the frozen flow map reduces masked MMD by 37% relative to the task-specific FT-WFM (0.385 vs. 0.612) at matching accuracy, and FMTT also improves on FT-WFM in masked MMD (0.480) at slightly lower accuracy (0.616 against 0.653). Both training-free arms on the velocity prior fail on this task.

Population super-resolution. M of 1024 cells are revealed, at $M = 16$ (1.6%) and $M = 256$ (25%). Table 11 reports masked MMD on the missing cells, rare-type recall and clustering ARI, together with a bootstrap control that merely resamples the observed cells. Lookahead attains the lowest masked MMD among the generative methods at both densities (0.171 and 0.089 vs. 0.187 and 0.113 for FT-WFM), while FMTT trails FT-WFM on this metric. Masked MMD alone is not sufficient, however: the bootstrap control, which can never produce an unobserved type, reaches 0.184 and 0.056, so it is competitive at $M = 16$ and best of all at $M = 256$. Rare-type recall separates the methods: FMTT has the highest recall at $M = 16$ (0.703), FlowDPS at $M = 256$

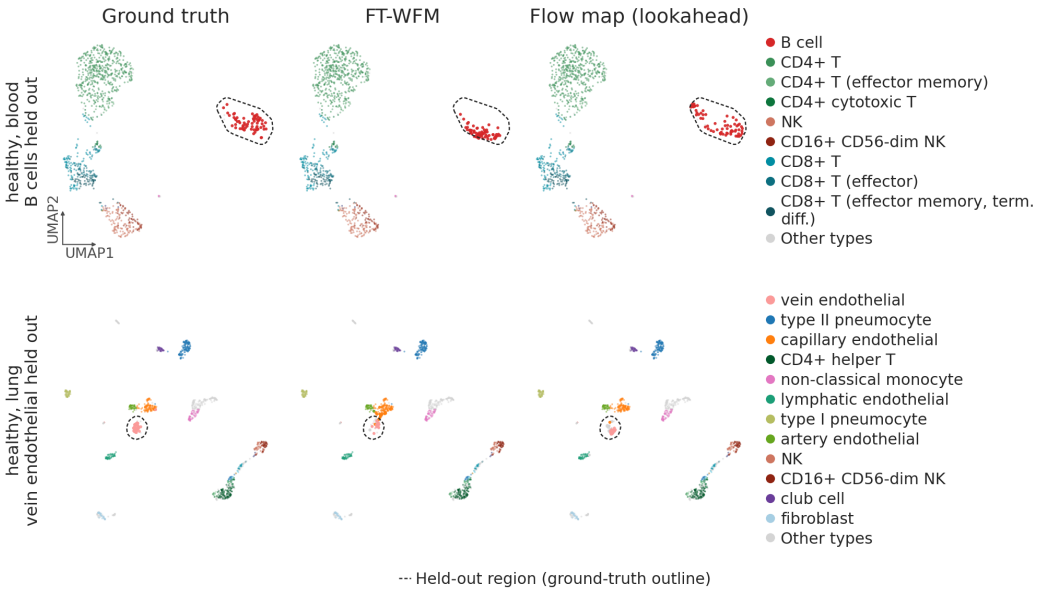


Figure 8: **Single-cell cell-type inpainting.** UMAP of two held-out patients in which every cell of one type is removed and must be regenerated without the identity of the missing type being supplied (B cells, top; vein endothelial cells, bottom). Panels: ground truth, FT-WFM, and lookahead on the frozen flow map, coloured by cell type. Scores are in Table 10.

Table 10: **Single-cell cell-type inpainting.** Every cell of one type is removed and regenerated blind (App. G.3). Masked MMD is the point-kernel MMD between the regenerated cells and the true held-out cells; cell-type accuracy is exact-match agreement of those cells with the held-out type (App. F.2.3). Means over $n = 100$ held-out patients. FT-WFM is the RWEFM fine-tuned for this task (App. F.4); FlowDPS and PnP-Flow run on the frozen RWEFM; Lookahead and FMTT run on the frozen flow map. NFE is 100 for the RWEFM arms and 16 for the flow-map arms. Best per column in bold.

Prior	Method	Masked MMD ↓	Cell-type acc. ↑
RWEFM	FT-WFM	0.612	0.653
	FlowDPS	0.819	0.023
	PnP-Flow	1.027	0.011
Flow map	Lookahead	**0.385**	**0.658**
	FMTT	0.480	0.616

(0.508), and at $M = 256$ FMTT (0.483) exceeds FT-WFM (0.441) while lookahead (0.432) does not. Figure 9 shows both densities on one patient.

H.3 Spatial niche

Generation. Figure 10 shows a composition embedding of all 1403 held-out GABAergic-centered niches in motor cortex, alongside the same number of niches generated unconditionally from noise with 8 map steps. Each point is one niche, placed by its cell-type composition vector, so the embedding summarizes what mixture of cell types surrounds the central cell. Generated niches occupy the same regions of the embedding as the held-out ones, with no visible gaps or spurious clusters.

Super resolution. Figure 11 shows super-resolution for one held-out niche of $N = 128$ cells from which only $M = 8 (6.25\%)$ cells are observed, marked with dark edges. The three panels show the ground truth, the fine-tuned WFM, and lookahead guidance on the frozen flow map, all embedded in a single UMAP of the 16-dimensional PCA space. Both methods place their reconstructed cells in the regions of the embedding occupied by the ground truth and reproduce its cell-type mixture, rather than collapsing onto the observed cells or onto a generic niche.

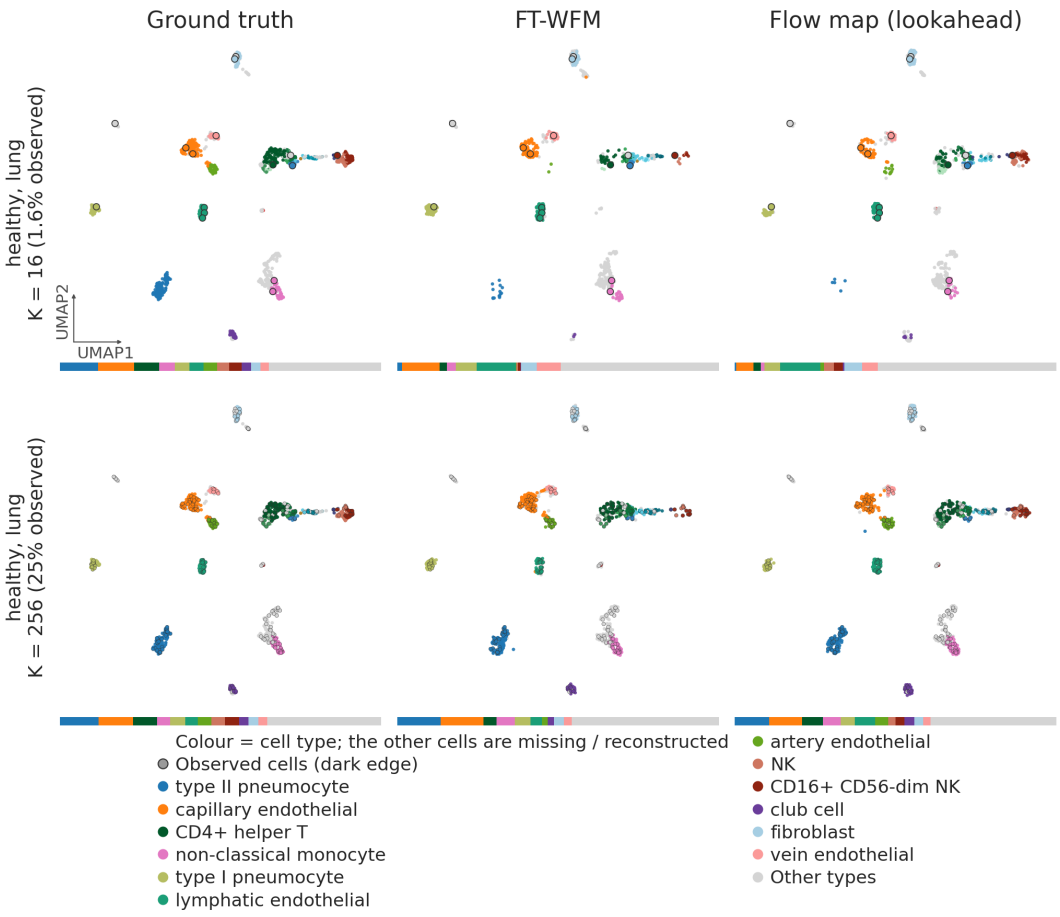


Figure 9: **Single-cell population super-resolution.** One held-out patient of $N = 1024$ cells with $M = 16$ (1.6%, top) and $M = 256$ (25%, bottom) cells observed (dark-edged markers). Panels: ground truth, FT-WFM, and lookahead on the frozen flow map, coloured by cell type. Types with no nearby observed cell, such as type II pneumocytes, are under-populated by every method. Scores are in Table 11.

Table 11: **Single-cell population super-resolution.** M of the 1024 cells are observed, at $M = 16$ (1.6%) and $M = 256$ (25%), and the rest are regenerated. Masked MMD is computed on the missing cells; rare-type recall is the fraction of types present in the ground truth but absent from the observation that the reconstruction recovers; clustering ARI (App. F.2.3) has no preferred direction and should match the ground-truth value, so the closest entry is in bold. Means over $n = 100$ held-out patients. The bootstrap control resamples the observed cells and therefore cannot recover unobserved types. Best per column in bold.

		$M = 16$ (1.6%)			$M = 256$ (25%)		
Prior	Method	Masked MMD ↓	Rare rec. ↑	ΔARI ↓	Masked MMD ↓	Rare rec. ↑	ΔARI↓
—	Bootstrap (control)	0.184	0.000	0.442	0.056	0.000	0.103
RWEFM	FT-WFM	0.187	0.660	**0.010**	0.113	0.441	0.005
	FlowDPS	0.241	0.692	0.054	0.161	**0.508**	0.045
	PnP-Flow	0.826	0.163	0.182	0.846	0.166	0.150
Flow map	Lookahead	**0.171**	0.683	0.018	**0.089**	0.432	0.014
	FMTT	0.206	**0.703**	0.021	0.129	0.483	**0.002**

Guided cell-type enrichment Figure 12 shows a qualitative example of guided generation with the composition reward of Eq. G.5. The target is a held-out niche from the high stratum, whose L2/3 IT abundance (0.61 on average over the stratum) is well above the population mean, so reaching it requires the generator to move substantially away from what it would produce unconditionally. The four panels show the target niche, the WFM fine-tuned with the requested fraction as a conditioning

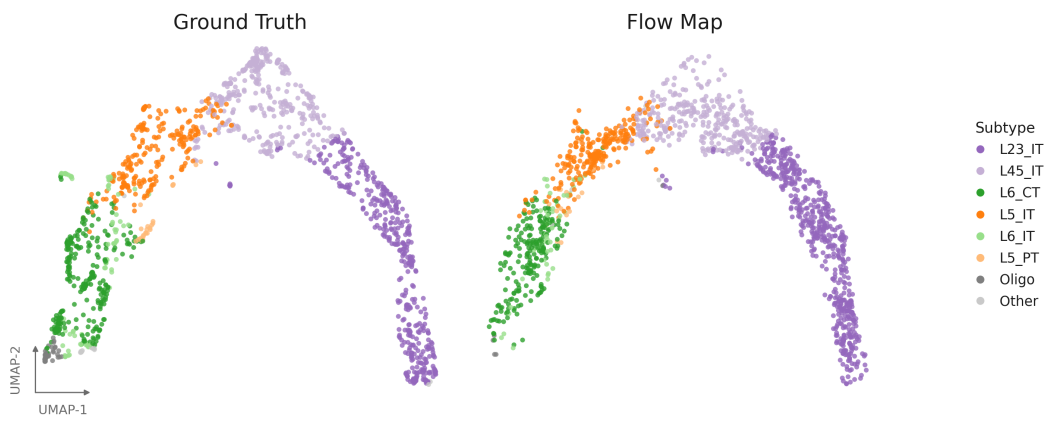


Figure 10: **Wasserstein flow maps generate cellular microenvironments in 8 steps.** Composition embedding of all 1403 held-out GABAergic-centered niches in motor cortex, colored by subtype. Niches generated unconditionally from noise reproduce the tissue landscape, recovering its superficial-to-deep laminar ordering unsupervised.

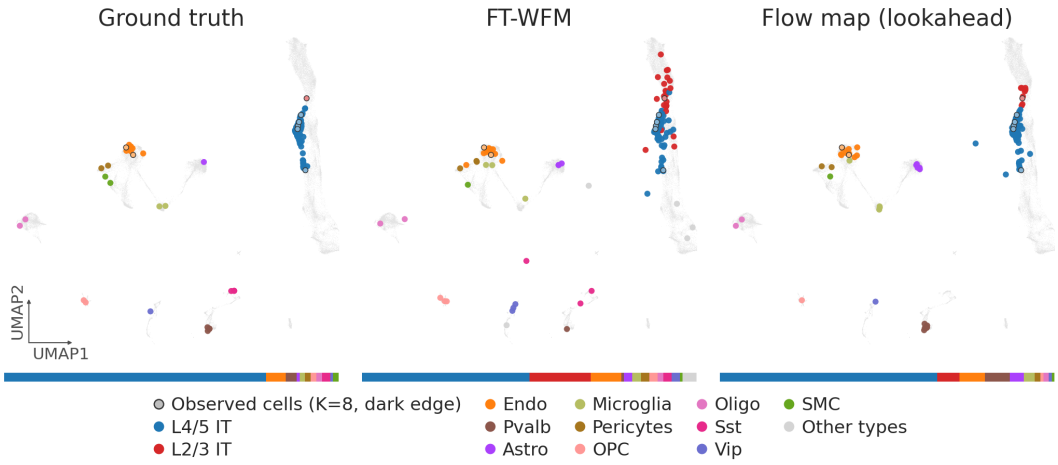


Figure 11: **Spatial niche super-resolution.** One held-out niche of $N = 128$ cells with $M = 8$ cells observed (dark-edged markers). Panels: ground truth, FT-WFM, and lookahead on the frozen flow map, shown in a shared UMAP of the 16-dimensional PCA embeddings and coloured by cell type. Quantitative results in Table 2.

input, and the two inference-time methods on the frozen flow map, lookahead and FMTT, again in a shared UMAP coloured by cell type. All three approaches increase the L2/3 IT fraction toward the requested value, visible as the growth of the corresponding cluster relative to the unconditional samples. The question the figure is meant to answer is whether that enrichment comes at the cost of realism: a method could reach the target fraction by adding L2/3 IT cells while leaving the rest of the niche unchanged, producing a composition that does not occur in tissue. The enrichment gap and realism scores that make this comparison quantitative, for all three abundance strata, are given in Table 3.

H.4 Generation quality against number of function evaluations

Figure 13 sweeps the evaluation budget for every method of Table 1, which reports the 8-evaluation flow maps and the 100-evaluation velocity baselines from these sweeps.

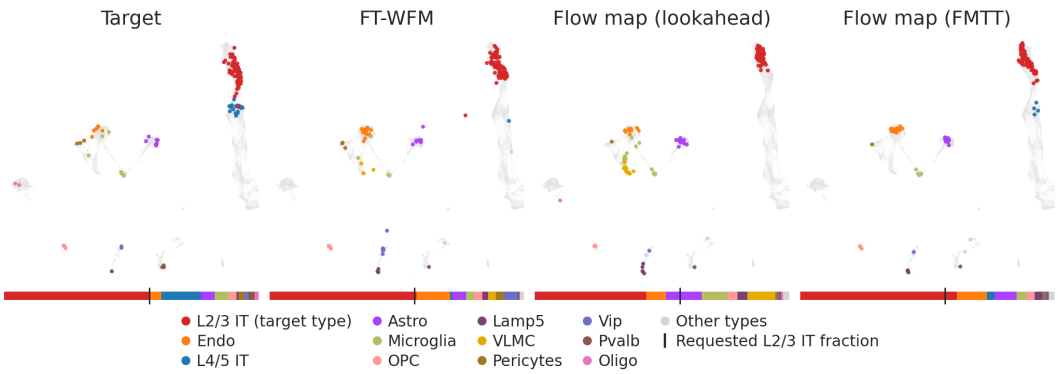


Figure 12: **Guided cell-type enrichment in spatial niches.** Generation steered by the composition reward of Eq. G.5 toward the L2/3 IT abundance of a held-out target niche from the high stratum (mean requested abundance 0.61). Panels: the target niche, the WFM fine-tuned with the requested fraction as a conditioning input, lookahead on the frozen flow map, and FMTT on the frozen flow map, in a shared UMAP coloured by cell type. Enrichment gaps and realism scores for all three strata are in Table 3.

I Ablations

I.1 Sampling from the learned velocity field

Table 1 compares a flow map at 8 function evaluations with WFM at 100, and the two arms differ in more than the sampler: training recipe, OT coupling, and budget change together. We therefore ablate the sampler itself, holding the trained network fixed and replacing the learned solution operator $\Phi_{s,t}$ with numerical integration of the velocity field the same network already defines. A trained flow map already carries its own velocity field: the diagonal

$$u(t,t)(x) \;=\; \frac{\log_x\big(\hat{x}_1(x;\, s=t,\, t=t)\big)}{\max(1-t,\,\varepsilon)}, \qquad \varepsilon = 0.05, \tag{I.1}$$

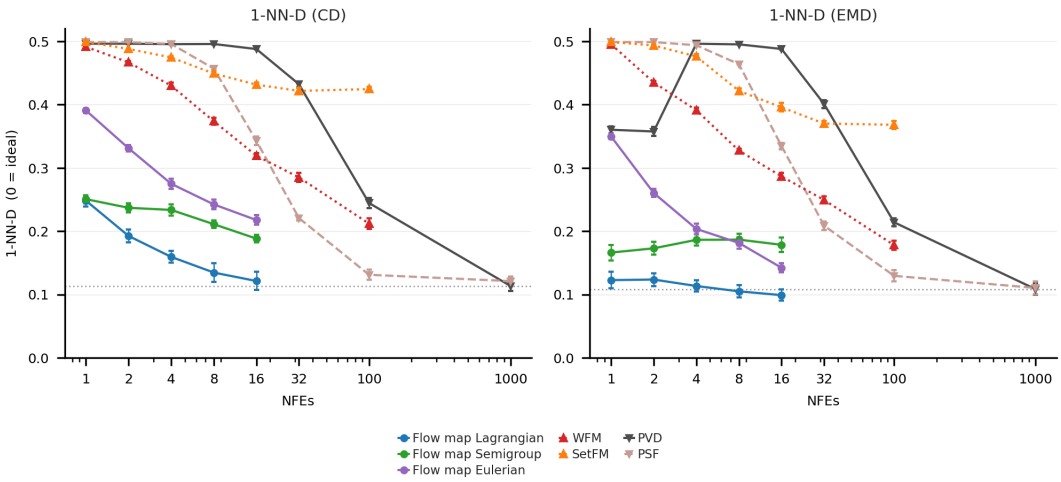
is the network evaluated at $s = t$, which is what the diagonal anchor loss trains. We take the Lagrangian checkpoint of Table 1, and run inference through the learned map and diagonal field integration.

As expected, the velocity field is poor at 1/2 steps where a map is not, and the field at 100 steps lands near the 8-step map and far from the independent-coupling velocity baseline (SetFM, or SetRFM on single cell) across all three domains (Figure 14) We note that two aspects of the parameterisation (App. D.4) may work against the integrated field and are worth separating from the comparison. Training samples $s \le s_{\max} = 0.94$, so $u(s,s)$ is untrained on the final interval, and the $\max(1-s,\varepsilon)$ in equation I.1 scales the velocity down once $s > 0.95$, so the trajectory undershoots. Replacing that tail with a single map jump $\Phi_{0.94,1}$ removes both, and the resulting hybrid improves on the 8-step map on Chamfer 1-NN-D in Euclidean domains, at 95 evaluations: 0.127 against 0.130 on ShapeNet and 0.257 against 0.301 on the niche. This hybrid is a single operating point and is not plotted.

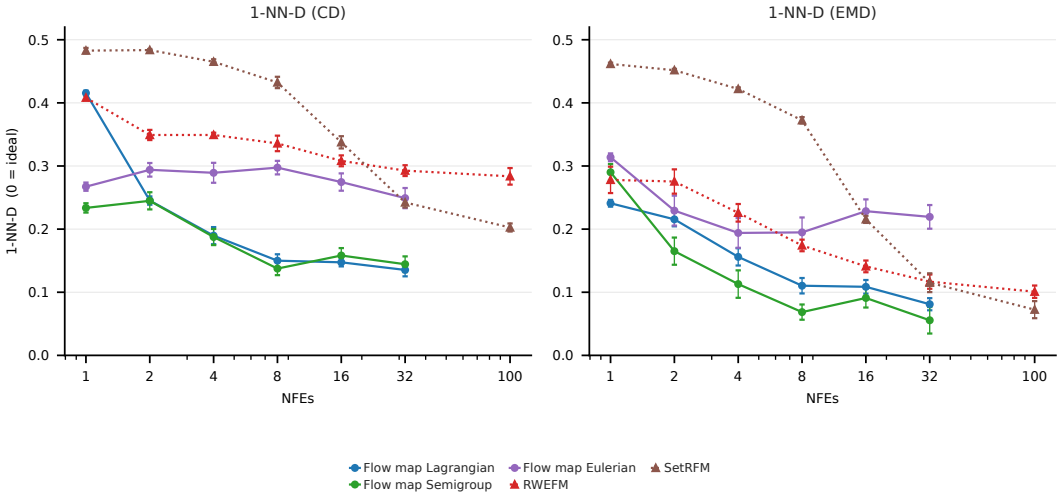
I.2 Cell generation without context

Cell-type inpainting conditions on the patient's surviving cells. To ask whether that context is doing real work, we compare two fine-tunes of the same base that differ only in whether it is supplied: a *no-context* arm conditioned on tissue, disease and the target cell-type centroid alone, with no observed cells, and a *with-context* arm that additionally sees the patient's non-target cells. Everything else — ground-truth type, metadata vector, random seed, solver — is held fixed. We stratify the 39 held-out (patient, type) pairs into equal-count tertiles by *deviation*, the distance between the patient's own mean embedding for that type and the global centroid for that type, i.e. how atypical this patient's version of the type is.

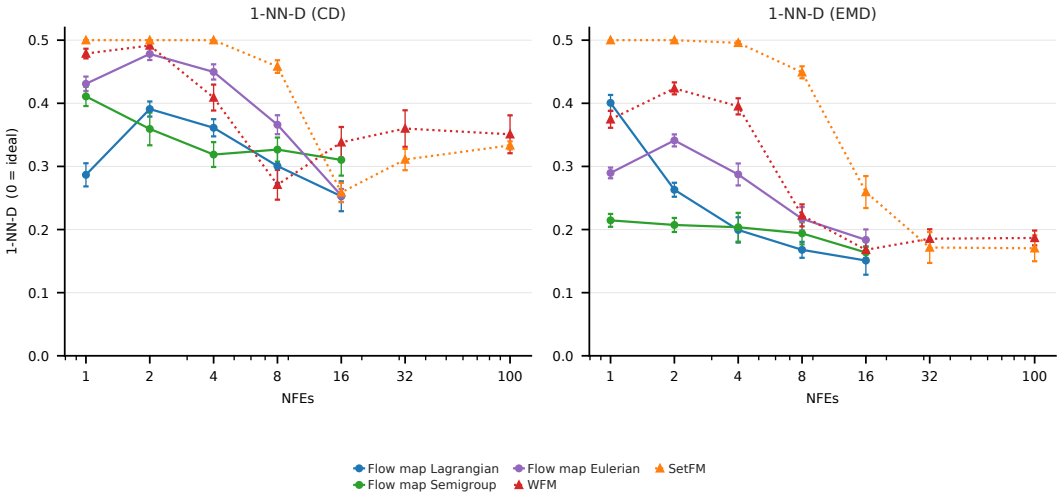
Context improves a lot in terms of geometry (Table 12), captured by tighter Chamfer distance by 3.3× and the with-context arm wins on Chamfer for 87% of pairs. The split by deviation is the



(a) ShapeNet

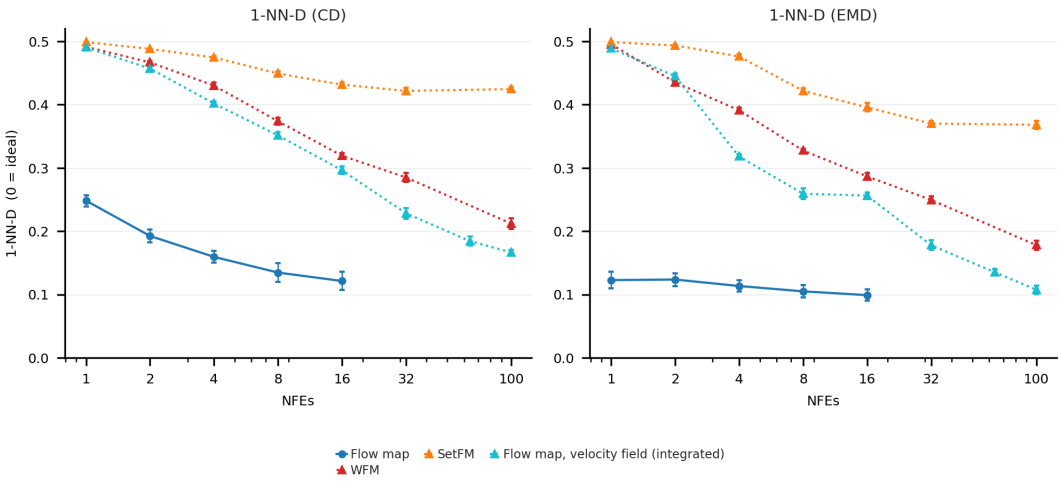


(b) Single cell

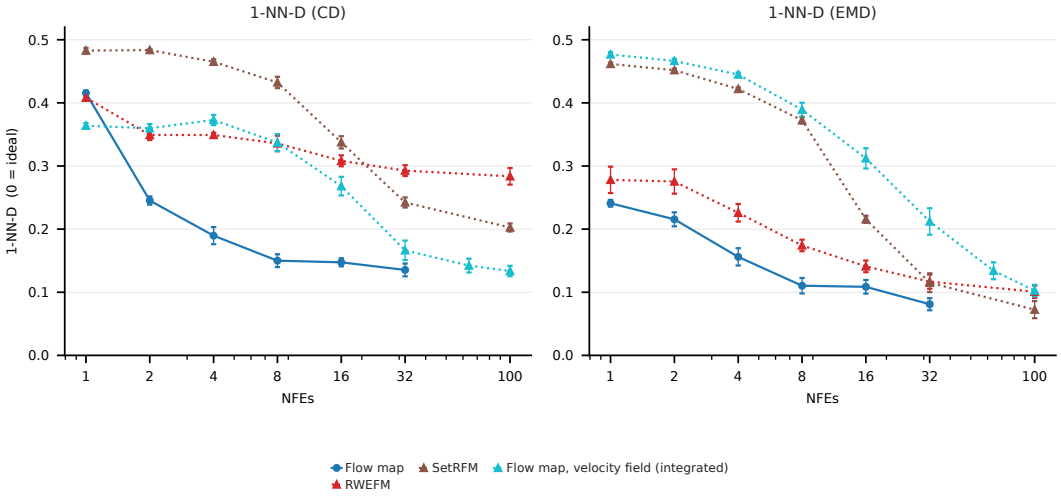


(c) Spatial niche

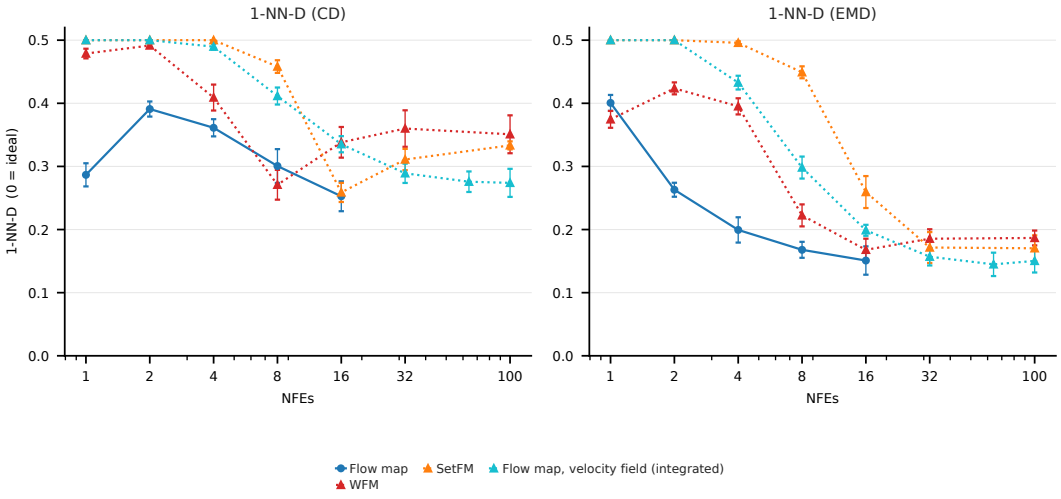
Figure 13: **Generation quality against the number of function evaluations, all methods.** 1-NN-D under both the CD and the EMD base distance on (a) ShapeNet, (b) single cell and (c) spatial niches for every method of Table 1. Error bars as in Figure 14. Table 1 reports the 8-evaluation flow maps and the 100-evaluation baselines from these sweeps.



(a) ShapeNet



(b) Single cell



(c) Spatial niche

Figure 14: **Sampling through the flow map versus integrating its own velocity field.** 1-NN-D under both the CD and the EMD base distance against the number of function evaluations across three datasets. Error bars are the standard deviation over 5 generation replicates of one training seed against a fixed reference set, and reflect sampling noise.

Table 12: **Does the observed context help?** Two fine-tunes of the same base model for cell-type inpainting that differ only in their conditioning: *no-context* sees tissue, disease and the target-type centroid but no observed cells; *with-context* additionally sees the patient's non-target cells. $n = 39$ held-out (patient, type) pairs, split into equal-count tertiles by *deviation*, the distance between the patient's own mean embedding of the target type and that type's global centroid, i.e. how atypical this patient's version of the type is. Exact accuracy is exact-match cell-type agreement; Chamfer is the symmetric squared CD of App. F.2 (the inpainting tables use a halved, unsquared convention and are on another scale); CD win rate is the fraction of pairs on which with-context has the lower Chamfer. Δ is with-context minus no-context for accuracy and no-context minus with-context for Chamfer, so positive is better in both columns.

		Exact acc. ↑		Chamfer ↓		CD win
Tier	Deviation	no-ctx → ctx	Δ	no-ctx → ctx	Δ	rate ↑
All (n=39)	—	0.652 → 0.706	+0.054	0.267 → 0.082	0.186	87%
Low (n=13)	0.09–0.15	0.727 → 0.808	+0.081	0.078 → 0.043	0.035	77%
Mid (n=13)	0.15–0.26	0.665 → 0.792	+0.127	0.221 → 0.094	0.127	85%
High (n=13)	0.27–0.87	0.564 → 0.517	−0.046	0.504 → 0.107	0.396	100%

Table 13: **Euclidean versus Riemannian single-cell flow maps.** The same architecture and objective trained either in the ambient space $\mathbb{R}^{128}$ or on the unit sphere $\mathbb{S}^{127}$ on which the embeddings lie. Metrics as in Table 1, at NFE $= 8$, over $n = 232$ held-out patients, from a single generation run. Within each objective the better geometry per column is in bold. The spherical rows are the single-cell flow-map rows of Table 1.

Objective	Geometry	1-NN-D (CD) ↓	1-NN-D (EMD) ↓	MMD (CD) ↓	MMD (EMD) ↓
Semigroup	Euclidean $\mathbb{R}^{128}$	0.204	**0.065**	5.52e-04	4.54e-05
	Sphere $\mathbb{S}^{127}$	**0.138**	0.069	**2.26e-04**	**2.24e-05**
Eulerian	Euclidean $\mathbb{R}^{128}$	0.439	0.375	3.64e-02	1.50e-03
	Sphere $\mathbb{S}^{127}$	**0.297**	**0.195**	**1.36e-03**	**1.49e-04**
Lagrangian	Euclidean $\mathbb{R}^{128}$	0.215	**0.110**	4.04e-04	3.30e-05
	Sphere $\mathbb{S}^{127}$	**0.150**	**0.110**	**2.04e-04**	**2.40e-05**

informative part: the Chamfer gain grows monotonically with atypicality, from 0.035 in the least deviant tertile to 0.396 (4.7×) in the most, where it wins every single pair — and deviation correlates with the Chamfer gain at $r = 0.73$ ($p \approx 10^{-7}$). Accuracy behaves differently and in fact *reverses* in the most atypical tertile (−0.046), while its correlation with deviation is not significant ($r = -0.20$, $p = 0.22$). So the context is not mainly telling the model *which* type to make, but *where that patient puts it*, which is exactly the patient-specific placement a population prior cannot supply on its own.

I.3 Euclidean versus Riemannian single-cell flow maps

The single-cell data lies on $\mathbb{S}^{128-1}$, but the same flow map can be trained in the ambient Euclidean space $\mathbb{R}^{128}$ instead. In Table 13, we compare the two at NFE $= 8$ and score with the same four statistics as Table 1: 1-NN-D and kernel MMD under both CD and EMD. The Riemannian formulation wins every MMD entry under both base distances and every 1-NN-D entry under CD; under EMD the two geometries tie for the Lagrangian objective and the Euclidean map is ahead for Semigroup, though that objective reverses under CD.