Learning Velocity Prior-Guided Hamiltonian-Jacobi Flows with Unbalanced Optimal Transport

Amy Xiang Wang

Abstract

The connection between optimal transport (OT) and control theory is well established, most prominently in the Benamou–Brenier dynamic formulation. With quadratic cost, the OT problem can be reframed as a stochastic control problem in which a density ρ_t evolves under a controlled velocity field v_t subject to the continuity equation $\partial_t \rho_t + \nabla \cdot (\rho_t v_t) = 0$. In this work, we introduce a velocity prior into the continuity equation and derive a new Hamilton–Jacobi–Bellman (HJB) formulation to learn dynamical probability flows. We further extend the approach to the unbalanced setting by adding a growth term, capturing mass variation processes common in scientific domains such as cell proliferation and differentiation. Importantly, our method requires training only a single neural network to model v_t, without the need for a separate model for the growth term g_t. By decomposing the total velocity as $v_{\text{total}} = v_{\text{prior}} + v_{\text{corr}}$, our approach can capture complex transport patterns that prior methods struggle to learn due to the curl-free limitation. Moreover, we incorporate Strang splitting for both training and inference, instead of the standard ODE integration for improved computational efficiency. We evaluate our method on synthetic and real-world single-cell RNA-seq datasets, demonstrating its ability to model curved trajectories, multiple branching cell fates and fast computation in high-dimensional dataset.

1. Introduction

From flow matching (FM) to action matching (AM), learning transport maps between distributions has been widely explored in recent years (Lipman et al., 2022; Albergo & Vanden-Eijnden, 2022; Liu et al., 2022; Neklyudov et al., 2023a). *Flow Matching (FM)* (Lipman et al., 2022) learns a time–dependent velocity field u_t that pushes ρ_0 to ρ_1 and can realize highly expressive transport paths; however, the original FM with independent coupling between source and target does not guarantee *least action* by minimizing the kinetic energy in the Benamou–Brenier sense. Instead, it trains u_t to match conditional expectations of displacement vectors under a chosen interpolation scheme, which may yield non-optimal flows.

Action Matching (AM) (Neklyudov et al., 2023a) addresses this limitation by parameterizing a scalar potential s_t whose gradient ∇s_t induces the transport, aligning with OT optimality conditions and yielding lower kinetic energy than unconstrained FM. The price is reduced expressiveness: ∇s_t is *curl-free*, so AM cannot directly represent rotational or cyclic dynamics that are common in scientific domains. From the Helmholtz decomposition perspective (Neklyudov et al., 2023a), any vector field u_t^* can be written as $u_t^* = \nabla s_t^* + w_t$, where w_t is divergence-free (Ambrosio et al., 2005, §8.4.2). Under this lens, AM retains only the gradient component and discards w_t, explaining both its energy efficiency and its inability to encode cycles.

In this paper, we seek a middle ground – expressive like FM, yet energy-aware like AM – by introducing a velocity prior v_{prior} and learning only a corrective potential. Even compared with energy-aware FM variants such as OT–CFM(Pooladian et al., 2023; Tong et al., 2023a), our approach achieves lower kinetic cost for the learned correction, as demonstrated in Table 1. Specifically, we decompose the velocity field as $v_{\text{total}}(t,x) = v_{\text{prior}}(t,x) + \nabla s_t(x)$. Here v_{prior} encodes known dynamics such as rotations or domain-specific effects such as RNA velocity in single-cell biology, while ∇s_t accounts for the OT-consistent gradient component. We train s_t by minimizing a prior–guided Hamilton–Jacobi residual that incorporates the prior, together with boundary terms that enforce $\rho_0 \to \rho_1$. This residualized formulation preserves gradient–based optimality conditions for the learned component, improves interpretability, and injects inductive bias without paying the kinetic-energy cost of learning the full velocity field. We name this approach as *Velocity Prior Hamiltonian-Jacobi Flow (VP-HJF)*.

Contributions We summarize our main contributions:

- **Velocity-prior guided Hamilton–Jacobi formulation.**

. Correspondence to: Amy Xiang Wang <xw914@nyu.edu>.

Preprint. September 1, 2026.

Table 1. Action Decomposition Comparison for Gaussian Translation (*note: control is full velocity for FM/OT-FM by definition)*

Method	Mean	Cov.	W_2	Control (∇s)	Kinetic (v_{tot})
Flow Matching	0.204	0.804	0.582	18.369	18.369
OT-FM	0.149	**0.659**	**0.402**	18.707	18.707
VP–HJF	0.102	0.791	0.577	**0.624**	**17.955**
Prior-only	**0.008**	0.171	1.351	0	36.250

We propose *Velocity Prior Hamilton–Jacobi Flow (VP–HJF)*, a prior-guided transport framework that decomposes the velocity field into a known prior and a learnable gradient correction, bridging the gap between the expressiveness of flow matching of modeling rotations or cycles and the energy-aware structure of action matching by learning a corrective model only instead of the full velocity field.

- **Single-network learning of transport and growth** We derive a prior-guided HJB residual objective that enables learning both transport corrections and mass variation using a single scalar potential network, without requiring a separate model.
- **Efficient Strang-splitting integration for prior-decomposed flows** We adapt Strang splitting, an operator splitting method well suited to our velocity decomposed method. This yields a stable and computationally efficient alternative to the standard ODE solvers.
- **Modeling curved and branching dynamics in single-cell data** We demonstrate that VP-HJF captures curved trajectories and branching structures in real-world scRNA-seq datasets, achieving competitive results.

2. Background

Dynamical Optimal Transport Beyond the classic static Monge–Kantorovich formulation in OT (Ambrosio et al., 2005; Villani et al., 2008), there exists a dynamical formulation known as the Benamou–Brenier problem which links OT with PDEs by representing the W_2 distance as the minimum kinetic energy where ρ_t is density and v_t is a velocity field with boundary conditions: $\rho_{|t=0} = \rho_0, \;\; \rho_{|t=1} = \rho_1$, (Benamou & Brenier, 2000):

$$W_2^2(\rho_0, \rho_1) = \inf_{\rho_t, v_t} \int_0^1 \int \tfrac{1}{2}\|v_t(x)\|^2 \rho_t(x)\, dx\, dt$$
$$\partial_t \rho_t + \nabla \cdot (\rho_t v_t) = 0. \tag{1}$$

Unbalanced Optimal Transport When total mass change over time such as following a growth-decay process in biology, we add a growth rate $g_t(x)$ term to the continuity equation to incorporate the weight changes (Chizat et al., 2018) with boundary conditions $\rho_{|t=0} = \rho_0, \;\; \rho_{|t=1} = \rho_1$:

$$\partial_t \rho_t(x) + \nabla \cdot \big(\rho_t(x)\, v_t(x)\big) = g_t(x)\, \rho_t(x), \tag{2}$$

The Wasserstein–Fisher–Rao distance $\mathrm{WFR}^2(\rho_0, \rho_1)$ is defined as the minimal action balancing transport cost and mass change:

$$\inf_{\rho, v, g} \int_0^1 \int \Big(\tfrac{1}{2}\|v_t(x)\|^2 + \tfrac{1}{2} g_t(x)^2\Big) \rho_t(x)\, dx\, dt, \quad \text{s.t. (2).} \tag{3}$$

Hamilton–Jacobi–Bellman (HJB). We recall the connection between optimal control and Hamilton–Jacobi–Bellman (HJB) theory. Consider a controlled system $\dot{x} = f(x, u, t)$ with running cost $L(x, u, t)$ and terminal cost $\psi(x)$, the value function $V(t, x) = \inf_{u(\cdot)} \int_t^1 L(x(s), u(s), s)\, ds + \psi(x(1))$ satisfies the HJB equation $\partial_t V(t, x) + H(x, \nabla V(t, x), t) = 0$, where $H(x, p, t) = \inf_u \{L(x, u, t) + p^\top f(x, u, t)\}$.

Action Matching (AM) AM fits a scalar potential s_θ to learn a energy-minimizing flow between distributions by minimizing the (un)balanced HJB residuals.

$$\mathcal{L}_{\text{uAM}} = \int_0^1 \mathbb{E}_{x \sim \rho_t} \left[\partial_t s_\theta + \tfrac{1}{2}\|\nabla_x s_\theta\|^2 + \frac{1}{2} s_\theta^2\right] dt, \tag{4}$$

with boundary constraints as: $\mathbb{E}_{x \sim \rho_0}[s_0(x)] - \mathbb{E}_{x \sim \rho_1}[s_1(x)]$.

3. Methodology

We introduce a velocity-prior guided approach, the *Velocity Prior Hamiltonian–Jacobi Flow (VP-HJF)*, which builds on the dynamic formulation of unbalanced optimal transport under the Wasserstein–Fisher–Rao (WFR) framework. Rather than optimizing the standard WFR objective directly, our approach incorporates a known velocity prior and learns a corrective transport and growth field, resulting in a prior-guided variant of the WFR action. In contrast to prior approaches that fit two separate networks—one for transport and one for growth (Zhang et al., 2024; Wang et al., 2025), our method trains a single neural network following (Neklyudov et al., 2023a).

Proposition 3.1 (Neklyudov et al., 2023a, Prop. 3.3)**.** *Suppose we have a continuous dynamic flow with density ρ_t. Under mild conditions, there exists a unique scalar potential function $\hat{s}_t(x)$ such that the unbalanced continuity equation (2) is satisfied, with the velocity field and growth function given by $v_t^*(x) = \nabla \hat{s}_t(x), g_t^*(x) = \hat{s}_t(x)$.*

Proposition 3.1 characterizes the optimal solution of the classical WFR problem, where the velocity field minimizing the kinetic–growth action is a gradient field. Building

on this characterization, VP–HJF introduces a *prior-guided variant* of the WFR formulation by incorporating problem-specific dynamics through an explicit velocity decomposition. Specifically, we decompose the velocity field into two parts: a known velocity prior and a learnable corrective velocity field component:

$$v_{\text{total}}(t,x) = v_{\text{prior}}(t,x) + v_{\text{corr}}(t,x), \tag{5}$$

where v_{prior} encodes known or domain knowledge (e.g. translations, rotations, RNA velocity), and v_{corr} is the data–driven learnable corrective component. In this way, the prior captures coarse dynamics while the model focuses on refinements such as correcting the residual transport and learning mass imbalance that the prior cannot explain. In essence, our approach improves interpretability and reduces the learning complexity through adding prior knowledge of the velocity field v_{prior} – leaving the learnable velocity field v_{corr} simpler learning tasks compared with other generative modeling methods of learning the entire velocity field v_{total}. Intuitively, our approach pays kinetic cost only for the *correction* to the prior drift and for the mass growth-decay component, making learning more efficient.

We now define the *velocity-prior guided unbalanced transport problem* – a prior-guided variant of WFR formulation that follows a least-action principle *with respect to the corrective dynamics*:

Definition 3.2 (Prior-guided unbalanced action)**.** Given a known velocity prior $v_{\text{prior}}(t,x)$, we define the prior-guided unbalanced transport action with boundary $\rho_{|t=0} = \rho_0,\ \rho_{|t=1} = \rho_1$ as

$$\underset{(\rho,\, v_{\text{corr}},\, g)}{\mathcal{A}} := \int_0^1 \int_\Omega \left(\tfrac{1}{2}\|v_{\text{corr}}\|^2 + \tfrac{1}{2}g^2\right) \rho_t \, dx\, dt, \tag{6}$$

$$s.t. \quad \partial_t \rho_t + \nabla\cdot\left(\rho_t(v_{\text{prior}} + v_{\text{corr}})\right) = g_t \rho_t.$$

Special case When $v_{\text{prior}} \equiv 0$, VP-HJF reduces to a purely gradient-driven flow. In this case, it becomes the standard WFR problem, and Proposition 3.1 applies directly.

Note that we do *not* optimize over ρ directly. Instead, ρ_t is *induced* by a parametric flow Φ_t^θ via $\dot{x} = v_{\text{prior}}(t,x) + \nabla s_\theta(t,x)$ and defined as $\rho_t^\theta = (\Phi_t^\theta)_{\#}\rho_0$. During training and evaluation, we sample particles from the normalized spacial distribution induced by this flow. Mass variation is modeled implicitly through the growth term $g(t,x) = s_\theta(t,x)$ appearing in the unbalanced continuity equation and the HJB objective (introduced below) rather than explicitly tracking the particle weights.

Prior-guided HJB residual Since solving for the prior-guided action minimization problem in Definition 3.2 is intractable in primal form, we turn to its dual formulation. The key derivation step is to introduce a scalar potential $s(t,x)$ as the Lagrange multiplier for the prior-guided continuity equation and applying the Fenchel–Young inequalities to the velocity field and growth term. We then obtain the following dual lower bound (see Appendix A for details):

$$\begin{aligned}\underset{(\rho,\, v_{\text{corr}},\, g)}{\mathcal{A}} \;\geq\; & \mathbb{E}_{x\sim\rho_1}[s_1(x)] - \mathbb{E}_{x\sim\rho_0}[s_0(x)] \\ & - \int_0^1 \int \rho_t \left(\partial_t s + \tfrac{1}{2}\|\nabla s\|^2 + \nabla s \cdot v_{\text{prior}}\right) dx\, dt \\ & - \int_0^1 \int \rho_t \, \tfrac{1}{2} s^2 \, dx\, dt. \end{aligned} \tag{7}$$

The bound is tight point-wise if and only if the primal variables satisfy

$$v_{\text{corr}}(t,x) = \nabla_x s(t,x), \qquad g(t,x) = s(t,x),$$

which shows that s can simultaneously control both the *corrective* transport $\nabla_x s$ and the local growth s under the prior-guided objective. We can then plug these back to the continuity equation to get the optimal particle dynamics and their log-weights evolve as

$$\begin{aligned} \frac{d}{dt}x(t) &= v_{\text{total}}(t,x) = v_{\text{prior}}(t,x) + \nabla_x s(t,x), \\ \frac{d}{dt}\log w_t(x(t)) &= s(t,x(t)). \end{aligned} \tag{8}$$

Corollary 3.3 (HJB residual objective)**.** *Motivated by the first-order optimality conditions induced by the prior-guided unbalanced transport action, we parameterize the potential $s_\theta(t,x)$ with a neural network and define the* velocity-prior guided HJB residual *as*

$$r_\theta(t,x) := \partial_t s_\theta + \tfrac{1}{2}\|\nabla_x s_\theta\|^2 + \nabla_x s_\theta \cdot v_{\text{prior}} + \tfrac{1}{2}s_\theta^2. \tag{9}$$

We then define the HJB residual objective

$$\begin{aligned}\mathcal{L}_{\text{HJB}}(\theta) = \mathbb{E}_{x\sim\rho_0}\left[s_\theta(0,x)\right] - \mathbb{E}_{x\sim\rho_1}\left[s_\theta(1,x)\right] \\ + \int_0^1 \mathbb{E}_{x\sim\rho_t}\left[r_\theta(t,x)^2\right] dt, \end{aligned} \tag{10}$$

which motivates the learned potential to satisfy the HJB stationary condition associated with the prior-guided action.

Practical estimation of the HJB objective In practice, we use squared residual to prevent positive and negative values from cancellation and optionally add an importance weight $w(t)$ trick to reduce variance (see Appendix C).

Since the evolving density ρ_t is not available in closed form, we use Monte Carlo sampling to approximate an implicit sample from the evolving density ρ_t. For continuous time dataset, we sample $t \sim \mathcal{U}(0,1)$ and interpolate the sample as $x_t = (1-t)x_0 + tx_1$, where (x_0, x_1) are

optionally OT-paired samples from adjacent time points for more complexed single-cell RNA-seq dataset. For discrete datasets with multiple time snapshots, we instead approximate ρ_t by a mixture of empirical marginals using $\rho_t \approx (1-\tau)\rho_k + \tau\rho_{k+1}$, and sample x_t accordingly by drawing from ρ_k or ρ_{k+1} with probabilities $1-\tau$ and τ. Moreover, we estimate the velocity prior $v_{\text{prior}}(t, x_t)$ as a Gaussian-weighted KNN average of nearby velocities.

Theorem 3.4 (Prior-guided HJB optimality)**.** *Assume $(\rho_t, v_{\text{corr}}, g_\theta)$ is feasible for the prior-guided unbalanced transport action in Definition 3.2. If the HJB residual defined in Corollary 3.3 satisfies $r_\theta(t,x) = 0$ for ρ_t-a.e. on $[0,1]\times\mathbb{R}^d$, then $(\rho_t, v_{\text{corr}}, g_\theta)$ satisfies the first-order (primal) optimality conditions of the prior-guided WFR action. In particular, the learned corrective field $v^*_{\text{corr}} = \nabla_x s_\theta$ and growth $g^* = s_\theta$ defines a minimizer of the prior-guided unbalanced transport objective. (See Appendix B).*

While the HJB residual enforces local first-order optimality conditions, it does not guarantee that the terminal distribution ρ_1^θ matches the target ρ_1. To address this, we design a two-part reconstruction objective: (i) a density matching term based on the sliced Wasserstein distance between the predicted $\hat{\rho}_1$ and ρ_1, and (ii) a mass alignment term that matches the global log-mass ratio. These two terms calibrate the terminal distribution's *shape* and *total mass*, complementing the HJB residual.

Reconstruction loss To align the terminal distribution, we use a sliced Wasserstein objective. Let $\hat{\rho}_1$ be the predicted terminal distribution, and ρ_1 be the target distribution, the sliced Wasserstein loss is defined as: $\text{SW}_2^2(\hat{\rho}_1, \rho_1) \approx \frac{1}{L}\sum_{\ell=1}^{L} W_2^2\Big(\langle\theta_\ell, \hat{X}_1\rangle, \langle\theta_\ell, Y\rangle\Big)$, where $\theta_\ell \sim \text{Unif}(\mathbb{S}^{d-1})$ are random projections and W_2 is the 1D Wasserstein distance.

To capture *global mass change*, we track the evolution of particle weights by the learned growth term. We initialize log-weights as $\log w_i(0) = 0$. The weights evolve through $\frac{d}{dt}\log w_i(t) = s_\theta\big(t, x_i(t)\big)$, yielding terminal mass $M(1) = \sum_i \exp(\log w_i(1))$.

To calculate the ground truth mass, we neither have access to the full density ρ_t nor to the absolute scale of $M(t)$. Instead, we use relative mass changes between adjacent time points, estimated from the number of observed particles at each time interval (t_k, t_{k+1}). For instance, we can approximate the ground-truth log mass ratio by $\log r_{\text{mass}} = \log\frac{M(t_{k+1})}{M(t_k)} \approx \log\left(\frac{N_{k+1}}{N_k}\right)$, Then the mass loss becomes $\mathcal{L}_{\text{mass}} = \big(\log\hat{r} - \log r_{\text{mass}}\big)^2$. Leading to the final reconstruction loss as:

$$\mathcal{L}_{\text{recon}} = \lambda_{\text{sw}}\,\text{SW}_2^2(\hat{\rho}_1, \rho_1) + \lambda_{\text{mass}}\big(\log\hat{r} - \log r_{\text{mass}}\big)^2 \tag{11}$$

Total objective Putting the pieces together, our total training loss is

$$\begin{aligned}\min_\theta\ \mathcal{L}(\theta) &= \mathbb{E}_{x\sim\rho_0}[s_0(x)] - \mathbb{E}_{x\sim\rho_1}[s_1(x)] \\ &+ \lambda_{\text{hjb}}\int_0^1 \mathbb{E}_{x\sim\rho_t}\big[\,r_\theta(t,x)^2\big]\,dt \\ &+\ \lambda_{\text{sw}}\,\text{SW}_2^2(\hat{\rho}_1,\rho_1) + \lambda_{\text{mass}}\big(\log\hat{r} - \log r_{\text{mass}}\big)^2\end{aligned} \tag{12}$$

where r_θ is the HJB term and $\hat{\rho}_1$ is the predicted distribution.

Strang splitting vs. standard ODE integration For our method, both training like to calculate the sliced Wasserstein loss and inference require integrating the learned dynamics $\dot{x} = v_{\text{prior}}(t,x) + v_{\text{corr}}(t,x)$. While we could use standard black-box ODE solver (e.g. `odeint`), this can be inefficient when the solvers do not exploit the intrinsic decomposition of the vector field or have different numerical properties when the prior dominates or is rotational.

Since VP-HJF explicitly decomposes the dynamics into a prior drift and a learnable corrective field, we adopt *Strang splitting*, a structure-aware operator-splitting method that alternately integrating two subflows (Strang, 1968).

Let $\Phi^{\text{prior}}_{\Delta t}$ and $\Phi^{\text{corr}}_{\Delta t}$ denote the time-Δt flow maps of $\dot{x} = v_{\text{prior}}(t,x)$ and $\dot{x} = v_{\text{corr}}(t,x)$, respectively. One Strang step from t_n to t_{n+1} is given by

$$x_{n+1} = \Phi^{\text{prior}}_{\Delta t/2} \circ \Phi^{\text{corr}}_{\Delta t} \circ \Phi^{\text{prior}}_{\Delta t/2}(x_n). \tag{13}$$

Equivalently, in operator form, let $\mathcal{A}$ and $\mathcal{B}$ denote the generators associated with v_{prior} and v_{corr} respectively. Strang splitting approximates the full evolution operator $e^{\Delta t(\mathcal{A}+\mathcal{B})}$ by:

$$e^{\Delta t(\mathcal{A}+\mathcal{B})} \approx e^{\frac{\Delta t}{2}\mathcal{A}} e^{\Delta t\,\mathcal{B}} e^{\frac{\Delta t}{2}\mathcal{A}} + \mathcal{O}(\Delta t^3), \tag{14}$$

which yields a globally second-order accurate and structure-preserving integrator.

In practice, this splitting improves numerical stability and reduces computational overhead of the ODE solvers. Empirically, we observe both improved accuracy and reduced runtime compared to standard `odeint` integration, especially in high dimensions. See Algorithm 1 for a complete summary of our approach.

4. Related works

Physics-constrained approaches Several trajectory-inference methods incorporate prior structure through *potential-based* modifications of Hamilton–Jacobi (HJ) equations, e.g. by adding a scalar potential $V_t(x)$ that induces conservative dynamics (Koshizuka & Sato, 2022; Neklyudov et al., 2023b; Tong et al., 2020). In (Neklyudov

et al., 2023b), this yields $\partial_t s + \frac{1}{2}\|\nabla s\|^2 + V_t + \frac{1}{2}s^2 = 0$ and the conservative second–order law $\ddot{X}_t = -\nabla V_t(X_t)$. In contrast, our method uses a *velocity-based prior* approach: a known velocity field v_{prior} enters directly as a drift in the continuity equation, producing the cross-term $\nabla s \cdot v_{\text{prior}}$. Because potential-based priors are curl-free and can not represent rotational or cyclic dynamics (Neklyudov et al., 2023a). Our formulation instead uses a flexible measured vector fields directly as v_{prior}, which can be considered as a free drift and learns only minimal optimal corrections and growth via the same scalar potential.

Most recently, CURLY-FM (Petrović et al., 2025) proposes a two-stage pipeline: it first learns a smooth global drift field from approximate velocities (e.g., RNA velocity), then solves a Schrödinger bridge problem with this learned nonzero drift to produce trajectories that follow the inferred curved manifold. VP–HJF differs in two key ways. First, our framework does *not* require an explicit first-stage training step to learn a reference drift. The prior field is injected directly as v_{prior} in the HJB formulation, and only the residual potential s_θ is trained. Second, the learned component is *energy-aware* by construction. Moreover in another work by Gu et al. (2024), the dynamic prior is assumed to be a clean, well-specified prior. Our setting is more flexible by using the corrective field v_{corr} to adjust noisy or misspecified priors.

Other trajectory inference approaches Trajectory inference has also advanced through flow matching(Haviv et al., 2024; Kapusniak et al., 2024; Atanackovic et al., 2024; Eyring et al., 2023), and Schrödinger bridge methods, which scale effectively to high-dimensional data. Recent SB variants further improve performance on single-cell datasets include (Huguet et al., 2022; Tong et al., 2023b; Shen et al., 2024; Hong et al., 2025; Pariset et al., 2023; Lavenant et al., 2024). For unbalanced settings, variational and regularized UOT methods such as TIGON, DeepRUOT, Var-RUOT, VGFM,and UMFSB directly learn transport dynamics and growth from snapshot data (Sha et al., 2024; Sun et al., 2025; Wang et al., 2025; Zhang et al., 2025). In particular, Var-RUOT also uses a single network to model both the velocity field and the growth term, but it did not incorporate known priors. Furthermore, Var-RUOT trains a global-in-time objective by explicitly simulating the dynamics over the entire time horizon on a dense grid, which can be computationally expensive. In contrast, we use per interval training scheme combined with a mixture-based or interpolation methods to sample the intermediates between intervals.

Algorithm 1 Multi-marginal training for VP–HJF

Input: time snapshots $\{x, t_k, v_k^{\text{prior}}\}_{k=0}^K$, model $s_\theta(t,x)$, weights $\lambda_{\text{hjb}}, \lambda_{\text{sw}}, \lambda_{\text{mass}}$, batch size B
while not converged **do**
  Sample a global batch across all intervals $(x, t, v_{\text{prior}})_{b=1}^B \sim \{x, t, v_{\text{prior}}\}_{k=0}^K$
  Compute boundary terms $s_0 \leftarrow s_\theta(t_0, x_0)$, $s_1 \leftarrow s_\theta(t_1, x_1)$
  for $k = 0$ **to** $K-1$ **do**
    Sample adjacent pairs $(x_k, v_k, x_{k+1}, v_{k+1}, r_m)^B \sim \{x, t, v_{\text{prior}}\}_{k=0}^K$
    HJB residual loss on interval $[t_k, t_{k+1}]$**:**
    for $m = 1$ **to** M_{HJB} **do**
      Sample $\tau^{(m)} \sim \mathcal{U}(0,1)$
      Sample $x_{t,k}$ from $\rho_{t,k} \approx (1-\tau)\rho_k + \tau\rho_{k+1}$
      Estimate $v_{\text{prior}}(t_k^{(m)}, x_{t,k})$ via KNN
      Compute $r_{t,k} \leftarrow \partial_t s_{t,k} + \frac{1}{2}\|\nabla_x s_{t,k}\|^2 + \nabla_x s_{t,k} \cdot v_{\text{prior}}(t_k^{(m)}, x_{t,k}) + \frac{1}{2}(s_{t,k})^2$
      $\mathcal{L}_{\text{HJB},k} \leftarrow \mathcal{L}_{\text{HJB},k} + \frac{1}{B}\sum_{b=1}^B (r_{t,k}^{(b)})^2$
    end for
    $\mathcal{L}_{\text{HJB},k} \leftarrow \frac{1}{M_{hjb}}\mathcal{L}_{\text{HJB},k}$
    Reconstruction loss:
    Define dynamics:
      $\dot{x} = \nabla_x s_\theta(t,x) + v_{\text{prior}}(t,x)$, $\dot{\log w} = s_\theta(t,x)$
    Use Strang Splitting to obtain $(x_{k+1}^{\text{pred}}, \log \hat{r})$
    $\mathcal{L}_{\text{recon}} \leftarrow \mathcal{L}_{\text{recon}} + \lambda_{\text{SW}} W_2^2(x_{k+1}^{\text{pred}}, x_{k+1}) + \lambda_{\text{mass}}(\log \hat{r} - \log r_m)^2$
  end for
  Total loss: $\mathcal{L}(\theta) \leftarrow \lambda_{hjb}\mathcal{L}_{\text{HJB}} + \mathcal{L}_{\text{recon}}$ Update θ
end while

5. Experiments

5.1. Synthetic dataset

Balanced case - Gaussian Translation In this experiment, we compare the transport accuracy using W_2 and kinetic energy under a controlled balanced setting. In Table 1, VP-HJF achieves Wasserstein distance W_2 accuracy comparable to FM (Lipman et al., 2022) while OT-FM (Tong et al., 2023a; Pooladian et al., 2023) attains the lowest W_2 error. However, when evaluated using a WFR-style kinetic energy metric, both FM and OT-FM incur substantially higher action costs. This is because these methods must learn the entire velocity field,including both gradient and rotational components, whereas VP-HJF pays kinetic cost only for the corrective component ∇s. This demonstrates that VP-HJF leverages the structured prior effectively, where the prior dynamics carry most of the transport, and the learned correction ∇s makes adjustments. To verify that the improvement is not solely due to a strong velocity prior v_{prior}, we also report a prior-only baseline. While the prior alone achieves moderate transport accuracy, it also shows high kinetic cost. In contrast, our method balances accuracy and energy efficiency by refining the prior with a learned correction. Additionally, Figure 1 demonstrates our approach can model rotation dynamics including rotating rings and

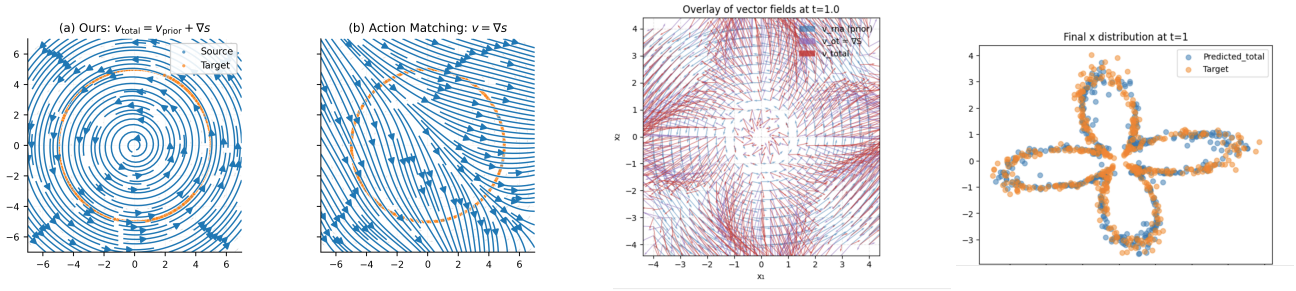


Figure 1. **Left** (first two): Ours correctly learned the rotating dynamic while AM failed. **Middle**: Vector field (red) bending away from v_{prior} (blue) to form petal shapes. **Right**: Predicted target distribution matches with ground truth

Table 2. Effect of adding $\mathcal{L}_{\text{mass}}$ for Lotka–Volterra. We report predicted mass ratio r_{pred}, absolute log error $|\Delta \log r|$, relative error (%), and velocity RMSE ($r_{\text{true}} = 1.418$).

Method	r_{pred}	$\lvert\Delta \log r\rvert$	Rel. err.	Vel. RMSE
VP–HJF	1.356	0.044	4.30	0.137
VP–HJF (w/o $\mathcal{L}_{\text{mass}}$)	0.899	0.455	36.56	0.158
Unbalanced AM	0.938	0.413	33.80	0.302
Prior-only	1.000	0.348	29.40	0.050

diverging petals structures. See Appendix E.

Lotka–Volterra with growth We consider a Lotka–Volterra system modeling interacting prey and predator populations $x_1(t), x_2(t)$ governed by a nonlinear dynamics, $\dot{x}_1(t) = \alpha\, x_1(t) - \beta\, x_1(t)\, x_2(t)$, $\dot{x}_2(t) = -\gamma\, x_2(t) + \delta\, x_1(t)\, x_2(t)$, where α,β, γ and δ are the prey and predator's growth rate and mortality rates (Goel et al., 1971). We introduce an explicit scalar growth field $g\big(x(t)\big) = \kappa\,\big(x_1(t) - x_2(t)\big)$, $\frac{d}{dt}\log w(t) = g\big(x(t)\big)$ which induces time-varying mass through the particle weights to model population growth and decay.

In Table 2, the explicit mass term $\mathcal{L}_{\text{mass}}$ enables VP-HJF to match r_{true} closely with 4% relative error. In contrast, both unbalanced AM and VP-HJF without the $\mathcal{L}_{mass}$ term exhibit errors exceeding 30%. Although AM aligns transport, it leaves the scale of the scalar potential s_θ unconstrained, leading to the miscalibrated integrated growth $\int_0^t g(x(t))\, dt$. Meanwhile, $\mathcal{L}_{\text{mass}}$ provides mass endpoint constraint, yielding mass dynamics align with the ground truth.

5.2. Real-world dataset

In this section, we evaluate our method on three single-cell RNA-seq datasets. All provide RNA velocity, which we use as the velocity prior v_{prior}. Such priors are common in biological and scientific applications beyond single-cell data. Incorporating them introduces an inductive bias that reduces learning complexity.

EB scRNA-Seq data We evaluate cell-trajectory inference on the Embryoid Body (EB) dataset of Moon et al. (2019), using the preprocessed release from Koshizuka & Sato (2022); Tong et al. (2020). The dataset comprises five snapshots over 27 days, grouped as $t_0 \in [0, 3]$, $t_1 \in [6, 9]$, $t_2 \in [12, 15]$, $t_3 \in [18, 21]$, $t_4 \in [24, 27]$. Leveraging RNA velocity as a prior v_{prior} at each snapshot, we adopt the multi-marginal training strategy by training a shorter trajectory: at each time interval, we sample an adjacent pair (t_k, t_{k+1}) and learn only the transport and growth to move $\rho_{t_k} \to \rho_{t_{k+1}}$. This yields more stable gradients and low target variance than enforcing all time points jointly. For details of the training algorithm, see Algorithm 1.

We train on 100-dim data and compare against recent multi-marginal methods: 3MSBM (Theodoropoulos et al., 2025), SBIRR (Shen et al., 2024), MMFM (Rohbeck et al., 2025) and DMSB (Chen et al., 2023) as well as other methods using global-in-time joint training or unbalanced optimal transport DeepRUOT (Zhang et al., 2024), Var-RUOT (Sun et al., 2025) and MIOFlow (Huguet et al., 2022). We follow the experiment setup from 3MSBM by having $t = 1, 3$ as the held-out sets. Table 3 (left) shows that our method outperforms most baselines and remains competitive with Var-RUOT and 3MSBM. Notably, our method outperforms SBIRR and MMFM, which solve piecewise Schrödinger bridges and OT couplings problem whereas methods like 3MSBM and DMSB solve a single global optimization with a joint coupling, highlighting the benefits of using RNA velocity as a local prior with per-interval supervision. Table 3 (right) reports runtime comparisons, where VP-HJF achieves orders-of-magnitude speedups over Var-RUOT. The runtime difference is primarily due to Var-RUOT's reliance on stochastic particle simulation across dense time discretizations and repeated optimal transport computations. In contrast, we use the structure-aware Strang splitting ODE integration with only four steps for the terminal reconstruction. Additional comparisons with Var-RUOT are provided in Appendix E.

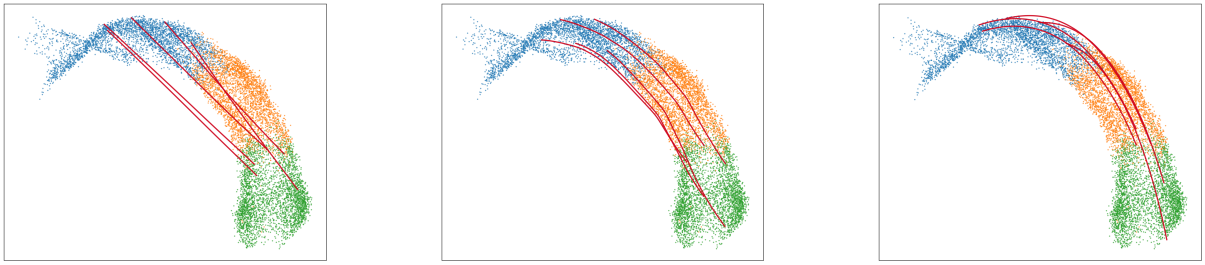
We also conducted robustness analysis on the misspecification of v_{prior} of our approach. Specifically, we perturb the reference field by (i) adding Gaussian noise, $v_{\text{prior}} = v_{\text{prior}} + \eta$, and (ii) rescaling its magnitude,

Table 3. Comparison on the EB dataset using SWD, MMD, and W_1 at held-out marginals (t_1, t_3). Baselines other than * are from (Theodoropoulos et al., 2025).

Method	SWD t_1	SWD t_3	MMD t_1	MMD t_3	W_1 t_1	W_1 t_3
DeepRUOT	0.73	0.67	0.43	0.36	13.45	14.90
Var-RUOT*	0.37	**0.24**	0.25	**0.06**	**10.28**	**11.92**
MIOFlow	0.84	0.94	1.01	0.92	13.20	13.57
SBIRR	0.80	0.91	0.71	0.73	15.09	20.39
MMFM	0.59	0.76	0.37	0.35	13.61	14.64
DMSB	0.58	0.54	0.38	0.36	14.08	15.22
3MSBM	0.48	0.38	**0.14**	0.18	13.89	13.11
VP–HJF (ODE)*	0.37	0.47	0.18	0.17	11.83	13.98
VP–HJF (Strang)*	**0.36**	0.25	0.24	**0.06**	10.52	12.46

Table 4. Training time comparison on the EB dataset ($d = 100$). Wall-clock time per iteration (batch size 256).

Method	time / iter (s)↓
Var-RUOT	136.19
VP–HJF (ODE)	2.43
VP–HJF (Strang)	0.22



(a) OT-FM (b) VP-HJF(ours) (c) CURLY-FM

Figure 2. Comparison of representative trajectories on the mouse erythroid dataset. **Left:** OT-FM produces nearly straight paths. **Right:** CURLY-FM learns strongly curved trajectories by modeling drift explicitly. **Middle:** Our VP-HJF integrates a velocity prior with a corrective potential, producing trajectories that follow the manifold geometry without enforcing excessive curvature.

Table 5. Robustness of VP–HJF to velocity-prior perturbations on EB (100D). Mean ± std over 5 seeds.

	Clean	Noise η		Scale c	
		0.25	0.75	0.5	1.5
$W_1(t_2)$	13.26±0.08	13.19±0.08	13.19±0.08	13.22±0.08	13.21±0.08
$W_1(t_4)$	14.59±0.09	14.58±0.10	14.57±0.09	14.56±0.10	14.62±0.09

$v_{\text{prior}} = c\, v_{\text{prior}}$. In Table 5, $\mathcal{W}_1$ at t_2 and t_4 changes only marginally under both noise levels ($\alpha \in \{0.25, 0.75\}$) and scaling factors ($c \in \{0.5, 1.5\}$). This indicates that VP–HJF is robust under mild to moderate prior perturbations, with the learned corrective field v_{corr} adapting to and compensating for mis-specification in v_{prior}.

Mouse erythroid scRNA-Seq data We evaluate VP-HJF on the mouse erythroid dataset (Pijuan-Sala et al., 2019) to assess its ability to model curved trajectories. Following the setup of CURLY-FM (Petrović et al., 2025), we use latent time to partition cells into three snapshots and adopt a simplified multi-marginal training strategy, where time points $t \in [0, 2]$ are used for training and $t = 1$ is the held out set for evaluation. To construct a continuous-time velocity prior, we estimate RNA velocity v_{prior} at sampled times using a KNN-based interpolation. In Figure 2, OT-FM favors near-straight transport paths due to its global action minimization, while CURLY-FM produces highly curved trajectories. Our approach resides in the middle, the velocity prior respects the biologically meaningful curvature while the learned corrective term v_{corr} enforces Hamilton–Jacobi consistency and discourages excessive deviation from low-action paths.This results in curved yet energy-aware trajectories that closely follow the erythroid differentiation manifold. Table 6 shows that our method achieves comparable results without requiring a separate drift network like CURLY-FM.

We also evaluate the velocity field directional alignment using cosine distance. CURLY-FM compares its learned drift directly against the KNN RNA velocity field. In VP-HJF, since the same KNN estimator constructs the velocity prior $v_{\text{prior}}(t, x) = v_{\text{knn}}(x)$, a direct comparison between v_{total} and $v_{\text{knn}}(x_t)$ would be trivial. Instead, to obtain a proxy directional target, we construct an endpoint-interpolated reference, where $v_0 = v_{\text{knn}}(x_0)$, $v_1 = v_{\text{knn}}(x_1)$, and interpolated with $v_{\text{target}} = (1 - t)v_0 + tv_1$. This metric evaluates directional consistency along inferred trajectories rather than pointwise velocity recovery, and is therefore not numerically identical to CURLY-FM's evaluation. Finally, we compare computational efficiency against flow matching-based models. VP-HJF demonstrates comparable runtime

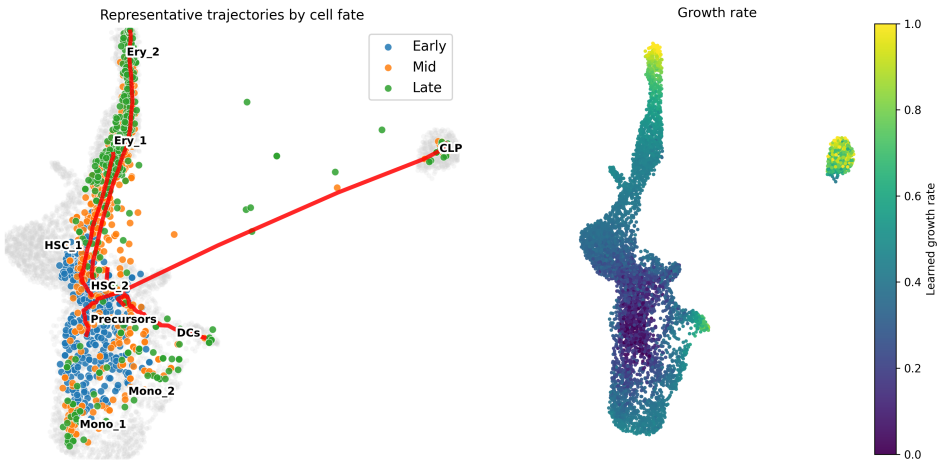


Figure 3. **Left**: Bone marrow trajectories. Colored points (ours) show inferred cell trajectories overlaid on the reference manifold (gray). **Right**: Learned growth field—yellow indicates high growth, purple low growth.

Table 6. Mouse erythroid results across dimensions (*baselines from (Petrović et al., 2025)*)

Algorithm ↓	$d = 2$		$d = 20$	
	W_2 ↓	Cos. Dist ↓	W_2 ↓	Cos. Dist ↓
OT-CFM	0.646 ± 0.006	0.146 ± 0.001	6.103 ± 0.074	0.489 ± 0.001
Metric FM	$\mathbf{0.269} \pm 0.004$	0.014 ± 0.001	$\mathbf{4.855} \pm 0.052$	0.495 ± 0.001
CURLY-FM	0.369 ± 0.090	$\mathbf{0.009} \pm 0.000$	6.124 ± 0.027	0.488 ± 0.001
VP–HJF (ODE)	0.454 ± 0.003	0.038 ± 0.001	4.887 ± 0.015	$\mathbf{0.424} \pm 0.036$
VP–HJF (Strang)	0.409 ± 0.006	0.036 ± 0.001	5.039 ± 0.012	0.460 ± 0.013

Table 7. Training time comparison on the Mouse erythroid dataset ($d = 2$). Wall-clock time per iteration (batch size 256).

Method	Time / iter (s)↓
OT-FM	0.008
CURLY-FM	0.069
VP–HJF (ODE)	0.086
VP–HJF (Strang)	0.073

to CURLY-FM, with the Strang splitting method providing a more efficient alternative to standard ODE integration.

Bone marrow scRNA-Seq data We evaluate our approach on the bone-marrow dataset with multiple fates from scVelo (Bergen et al., 2020). VP-HJF uses 30-dim data. Figure 3 (left) shows the representative medoid trajectories projected onto the reference UMAP embedding (gray). Trajectories originate from early progenitor regions, including HSCs and precursors (Early, blue), follow the major differentiation structure (Mid, orange), and reach terminal cell states (Late, green). The most frequent cell fate branches include Ery, CLP, and DC lineages. The red medoid trajectories show clear divergence toward these distinct branches, indicating that the model captures the dominant branching geometry of hematopoiesis.

Figure 3 (right) visualizes the learned effective growth field $g(t, x) = s_\theta(t, x)$, which governs local mass dynamics along the inferred trajectories. Early progenitor regions show low growth values, while growth increases progressively along differentiation paths, with strong signals along the Ery, CLP, and DC branches. This spatial pattern aligns with the representative trajectories on the left and reflects how mass is redistributed across lineages under the learned growth dynamics.

Importantly, the elevated growth observed near some terminal regions should not be interpreted as literal cell proliferation. In our framework, the growth term is learned jointly with transport as part of a prior-guided unbalanced transport objective and represents an *effective* mass correction rather than biological division. As a result, positive growth in terminal regions reflects a modeling correction that absorbs density mismatches rather than true biological proliferation. In the bone-marrow setting, fully mature or terminal cell states are expected to show limited growth rate. The observed terminal growth highlights a modeling trade-off rather than a biological claim. Incorporating additional structural biases on s_θ, such as temporal smoothness, terminal-state regularization, or lineage-specific boundary constraints, may further align the learned growth field with known biological behavior.

6. Conclusion

Our method uses domain knowledge to decompose the velocity field as prior and a single network to capture both growth and transport. This decomposition yields robust performance even under mild to moderate prior misspecification, indicating the flexibility of this framework to incorporate priors, making it a promising direction for modeling complex cellular dynamics.

Impact Statement

This work advances generative modeling by enabling the incorporation of domain knowledge into unbalanced optimal transport, which may improve interpretability and efficiency in scientific applications such as single-cell biology. By allowing practitioners to encode prior dynamics, the method has the potential to support modeling of complex dynamical systems and reduce reliance on purely data-driven approximations. As with other data-driven modeling approaches, careful consideration is required when applying this framework to sensitive domains to avoid misinterpretation of learned dynamics.

References

Albergo, M. S. and Vanden-Eijnden, E. Building normalizing flows with stochastic interpolants. *arXiv preprint arXiv:2209.15571*, 2022.

Ambrosio, L., Gigli, N., and Savaré, G. *Gradient flows: in metric spaces and in the space of probability measures*. Springer, 2005.

Atanackovic, L., Zhang, X., Amos, B., Blanchette, M., Lee, L. J., Bengio, Y., Tong, A., and Neklyudov, K. Meta flow matching: Integrating vector fields on the wasserstein manifold. *arXiv preprint arXiv:2408.14608*, 2024.

Benamou, J.-D. and Brenier, Y. A computational fluid mechanics solution to the monge-kantorovich mass transfer problem. *Numerische Mathematik*, 84(3):375–393, 2000.

Bergen, V., Lange, M., Peidli, S., Wolf, F. A., and Theis, F. J. Generalizing rna velocity to transient cell states through dynamical modeling. *Nature Biotechnology*, 38 (12):1408–1414, August 2020. ISSN 1546-1696. doi: 10.1038/s41587-020-0591-3. URL http://dx.doi.org/10.1038/s41587-020-0591-3.

Chen, T., Liu, G.-H., Tao, M., and Theodorou, E. Deep momentum multi-marginal schrödinger bridge. *Advances in Neural Information Processing Systems*, 36:57058–57086, 2023.

Chizat, L., Peyré, G., Schmitzer, B., and Vialard, F.-X. Unbalanced optimal transport: Dynamic and kantorovich formulations. *Journal of Functional Analysis*, 274(11): 3090–3123, 2018.

Eyring, L., Klein, D., Uscidda, T., Palla, G., Kilbertus, N., Akata, Z., and Theis, F. Unbalancedness in neural monge maps improves unpaired domain translation. *arXiv preprint arXiv:2311.15100*, 2023.

Goel, N. S., Maitra, S. C., and Montroll, E. W. On the volterra and other nonlinear models of interacting populations. *Reviews of modern physics*, 43(2):231, 1971.

Gu, A., Chien, E., and Greenewald, K. Partially observed trajectory inference using optimal transport and a dynamics prior. *arXiv preprint arXiv:2406.07475*, 2024.

Haviv, D., Pooladian, A.-A., Pe'er, D., and Amos, B. Wasserstein flow matching: Generative modeling over families of distributions. *arXiv preprint arXiv:2411.00698*, 2024.

Hong, W., Shi, Y., and Niles-Weed, J. Trajectory inference with smooth schr\" odinger bridges. *arXiv preprint arXiv:2503.00530*, 2025.

Huguet, G., Magruder, D. S., Tong, A., Fasina, O., Kuchroo, M., Wolf, G., and Krishnaswamy, S. Manifold interpolating optimal-transport flows for trajectory inference. *Advances in neural information processing systems*, 35: 29705–29718, 2022.

Kapusniak, K., Potaptchik, P., Reu, T., Zhang, L., Tong, A., Bronstein, M., Bose, J., and Di Giovanni, F. Metric flow matching for smooth interpolations on the data manifold. *Advances in Neural Information Processing Systems*, 37: 135011–135042, 2024.

Koshizuka, T. and Sato, I. Neural lagrangian schr\" odinger bridge: Diffusion modeling for population dynamics. *arXiv preprint arXiv:2204.04853*, 2022.

Lavenant, H., Zhang, S., Kim, Y.-H., Schiebinger, G., et al. Toward a mathematical theory of trajectory inference. *The Annals of Applied Probability*, 34(1A):428–500, 2024.

Lipman, Y., Chen, R. T., Ben-Hamu, H., Nickel, M., and Le, M. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Liu, X., Gong, C., and Liu, Q. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Moon, K. R., Van Dijk, D., Wang, Z., Gigante, S., Burkhardt, D. B., Chen, W. S., Yim, K., Elzen, A. v. d., Hirn, M. J., Coifman, R. R., et al. Visualizing structure and transitions in high-dimensional biological data. *Nature biotechnology*, 37(12):1482–1492, 2019.

Neklyudov, K., Brekelmans, R., Severo, D., and Makhzani, A. Action matching: Learning stochastic dynamics from samples. In *International conference on machine learning*, pp. 25858–25889. PMLR, 2023a.

Neklyudov, K., Brekelmans, R., Tong, A., Atanackovic, L., Liu, Q., and Makhzani, A. A computational framework for solving wasserstein lagrangian flows. *arXiv preprint arXiv:2310.10649*, 2023b.

Pariset, M., Hsieh, Y.-P., Bunne, C., Krause, A., and De Bortoli, V. Unbalanced diffusion schr\" odinger bridge. *arXiv preprint arXiv:2306.09099*, 2023.

Petrović, K., Atanackovic, L., Moro, V., Kapuśniak, K., Ceylan, I. I., Bronstein, M., Bose, A. J., and Tong, A. Curly flow matching for learning non-gradient field dynamics. *arXiv preprint arXiv:2510.26645*, 2025.

Pijuan-Sala, B., Griffiths, J. A., Guibentif, C., Hiscock, T. W., Jawaid, W., Calero-Nieto, F. J., Mulas, C., Ibarra-Soria, X., Tyser, R. C., Ho, D. L. L., et al. A single-cell molecular map of mouse gastrulation and early organogenesis. *Nature*, 566(7745):490–495, 2019.

Pooladian, A.-A., Ben-Hamu, H., Domingo-Enrich, C., Amos, B., Lipman, Y., and Chen, R. T. Multisample flow matching: Straightening flows with minibatch couplings. *arXiv preprint arXiv:2304.14772*, 2023.

Rohbeck, M., De Brouwer, E., Bunne, C., Huetter, J.-C., Biton, A., Chen, K. Y., Regev, A., and Lopez, R. Modeling complex system dynamics with flow matching across time and conditions. In *The Thirteenth International Conference on Learning Representations*, 2025.

Sha, Y., Qiu, Y., Zhou, P., and Nie, Q. Reconstructing growth and dynamic trajectories from single-cell transcriptomics data. *Nature Machine Intelligence*, 6(1):25–39, 2024.

Shen, Y., Berlinghieri, R., and Broderick, T. Multi-marginal schr\" odinger bridges with iterative reference refinement. *arXiv preprint arXiv:2408.06277*, 2024.

Strang, G. On the construction and comparison of difference schemes. *SIAM journal on numerical analysis*, 5(3):506–517, 1968.

Sun, Y., Zhang, Z., Wang, Z., Li, T., and Zhou, P. Variational regularized unbalanced optimal transport: Single network, least action. *arXiv preprint arXiv:2505.11823*, 2025.

Theodoropoulos, P., Saravanos, A. D., Theodorou, E. A., and Liu, G.-H. Momentum multi-marginal schr\" odinger bridge matching. *arXiv preprint arXiv:2506.10168*, 2025.

Tong, A., Huang, J., Wolf, G., Van Dijk, D., and Krishnaswamy, S. Trajectorynet: A dynamic optimal transport network for modeling cellular dynamics. In *International conference on machine learning*, pp. 9526–9536. PMLR, 2020.

Tong, A., Fatras, K., Malkin, N., Huguet, G., Zhang, Y., Rector-Brooks, J., Wolf, G., and Bengio, Y. Improving and generalizing flow-based generative models with minibatch optimal transport. *arXiv preprint arXiv:2302.00482*, 2023a.

Tong, A., Malkin, N., Fatras, K., Atanackovic, L., Zhang, Y., Huguet, G., Wolf, G., and Bengio, Y. Simulation-free schr\" odinger bridges via score and flow matching. *arXiv preprint arXiv:2307.03672*, 2023b.

Villani, C. et al. *Optimal transport: old and new*, volume 338. Springer, 2008.

Wang, D., Jiang, Y., Zhang, Z., Gu, X., Zhou, P., and Sun, J. Joint velocity-growth flow matching for single-cell dynamics modeling. *arXiv preprint arXiv:2505.13413*, 2025.

Zhang, Z., Li, T., and Zhou, P. Learning stochastic dynamics from snapshots through regularized unbalanced optimal transport. *arXiv preprint arXiv:2410.00844*, 2024.

Zhang, Z., Wang, Z., Sun, Y., Li, T., and Zhou, P. Modeling cell dynamics and interactions with unbalanced mean field schr\" odinger bridge. *arXiv preprint arXiv:2505.11197*, 2025.

A. Prior-guided HJB Inequality Derivation

We derive the velocity-prior guided HJB residual inequality in Eq. 7.

$$\begin{aligned}\mathcal{A}(\rho, v, g) \;&\geq \mathbb{E}_{x\sim\rho_1}[s_1(x)] - \mathbb{E}_{x\sim\rho_0}[s_0(x)] \\ &\quad - \int_0^1 \int \rho_t(x) \left(\partial_t s + \tfrac{1}{2}\|\nabla s\|^2 + \nabla s \cdot v_{\text{prior}} + \tfrac{1}{2}s^2\right) dx\, dt. \end{aligned} \tag{15}$$

We start by minimizing Definition 3.2. Since v_{prior} is constant and known, we only need to minimize the learned components $(\rho, v_{\text{corr}}, g)$. Then the velocity–prior guided WFR action becomes the following:

$$\begin{aligned} \min_{\rho,\, v_{\text{corr}},\, g} \quad & \mathcal{A}(\rho, v_{\text{corr}}, g) := \int_0^1 \int_\Omega \left(\tfrac{1}{2}\|v_{\text{corr}}\|^2 + \tfrac{1}{2}g^2\right)\rho\, dx\, dt \\ \text{s.t.} \quad & \partial_t \rho + \nabla\cdot\big(\rho(v_{\text{prior}} + v_{\text{corr}})\big) = g\,\rho, \qquad \rho_{|t=0} = \rho_0,\ \rho_{|t=1} = \rho_1. \end{aligned} \tag{16}$$

Step 1: Lagrangian formulation
First, we introduce a scalar multiplier $s(t, x)$, and define the Lagrangian by adding the continuity equation:

$$\begin{aligned}\mathcal{L} = &\int_0^1 \int_\Omega \left(\tfrac{1}{2}\|v_{\text{corr}}\|^2 + \tfrac{1}{2}g^2\right)\rho\, dx\, dt \\ &+ \int_0^1 \int_\Omega s\Big(\partial_t \rho + \nabla\cdot\left(\rho(v_{\text{prior}} + v_{\text{corr}})\right) - g\rho\Big)\, dx\, dt. \end{aligned} \tag{17}$$

Step 2: Integration by parts
Integration by parts for the time derivative term, $\int s\partial_t\rho$:

$$\begin{aligned}\int_0^1 \int_\Omega s\, \partial_t \rho\, dx\, dt &= \Big[\int_\Omega s\,\rho\, dx\Big]_{t=0}^{t=1} - \int_0^1 \int_\Omega \rho\, \partial_t s\, dx\, dt \\ &= \mathbb{E}_{x\sim\rho_1}[s_1(x)] - \mathbb{E}_{x\sim\rho_0}[s_0(x)] - \int_0^1 \int_\Omega \rho\, \partial_t s\, dx\, dt. \end{aligned} \tag{18}$$

For the divergence term $\int s\nabla\cdot(\rho w)$ with $w := v_{\text{prior}} + v_{\text{corr}}$:

$$\int_\Omega s\, \nabla\cdot(\rho w)\, dx = \int_{\partial\Omega} s\,\rho\, w\cdot n\, d\sigma - \int_\Omega \nabla s\cdot(\rho w)\, dx. \tag{19}$$

Assuming the boundary term vanishes under zero-flux or fast decay, the divergence term becomes:

$$\int_0^1 \int_\Omega s\, \nabla\cdot(\rho w)\, dx\, dt = -\int_0^1 \int_\Omega \rho\, w\cdot\nabla s\, dx\, dt. \tag{20}$$

Substituting these back and factoring out a minus sign for the integrals, the Lagrangian becomes:

$$\begin{aligned}\mathcal{L} &= \mathbb{E}_{x\sim\rho_1}[s_1(x)] - \mathbb{E}_{x\sim\rho_0}[s_0(x)] \\ &\quad - \int_0^1 \int_\Omega \rho_t(x) \left[\partial_t s + v_{\text{prior}}\cdot\nabla s - \left(\tfrac{1}{2}\|v_{\text{corr}}\|^2 - v_{\text{corr}}\cdot\nabla s\right) - \left(\tfrac{1}{2}g^2 - s\,g\right)\right] dx\, dt. \end{aligned} \tag{21}$$

Step 3: Fenchel–Young inequality
The Fenchel–Young inequality states that for any vectors a and p, we have:

$$\tfrac{1}{2}\|a\|^2 \geq p\cdot a - \tfrac{1}{2}\|p\|^2 \tag{22}$$

We set $a = v_{\text{corr}}$ and $p = \nabla s$, yielding:

$$\tfrac{1}{2}\|v_{\text{corr}}\|^2 - v_{\text{corr}}\cdot\nabla s \geq -\tfrac{1}{2}\|\nabla s\|^2 \tag{23}$$

Similarly, we set $a = g$ and $p = s$, yielding:

$$\tfrac{1}{2}g^2 - s\,g \geq -\tfrac{1}{2}s^2, \tag{24}$$

with equality iff $v_{\text{corr}} = \nabla s$ and $g = s$. Because these terms are subtracted inside the Lagrangian integral, replacing them with their lower bounds establishes a lower bound on the overall action. Thus, putting the pieces together, we have:

$$\begin{aligned}\mathcal{A}(\rho, v_{\text{corr}}, g) \;\geq\; & \mathbb{E}_{x\sim\rho_1}[s_1(x)] - \mathbb{E}_{x\sim\rho_0}[s_0(x)] \\ & - \int_0^1 \int_\Omega \rho_t(x) \left(\partial_t s \;+\; \tfrac{1}{2}\|\nabla s\|^2 \;+\; \nabla s \cdot v_{\text{prior}} \;+\; \tfrac{1}{2}s^2\right) dx\, dt.\end{aligned} \tag{25}$$

□

B. Proof of Theorem 3.4

Theorem B.1 (Prior-guided HJB optimality)**.** *Assume $(\rho_t, v_{\text{corr}}, g_\theta)$ is feasible for the prior-guided unbalanced transport action in Definition 3.2. If the HJB residual defined in Corollary 3.3 satisfies $r_\theta(t, x) = 0$ for ρ_t-a.e. on $[0, 1] \times \mathbb{R}^d$, then $(\rho_t, v_{\text{corr}}, g_\theta)$ satisfies the first-order (primal) optimality conditions of the prior-guided WFR action. In particular, the learned corrective field $v^*_{\text{corr}} = \nabla_x s_\theta$ and growth $g^* = s_\theta$ defines a minimizer of the prior-guided unbalanced transport objective.*

Proof sketch The proof of this theorem builds upon the derivations from the previous proof in Appendix A. Recall, from step 3 Fenchel–Young inequality above, we have:

$$\mathcal{A}(\rho, v_{\text{corr}}, g) \;\geq\; \mathbb{E}_{x\sim\rho_0}[s_0(x)] - \mathbb{E}_{x\sim\rho_1}[s_1(x)] - \int_0^1 \int_\Omega \rho_t(x)\, r_s(t, x)\, dx\, dt. \tag{26}$$

where the HJB residual is defined as:

$$r_s(t, x) := \partial_t s + \tfrac{1}{2}\|\nabla_x s\|^2 + \nabla_x s \cdot v_{\text{prior}} + \tfrac{1}{2}s^2. \tag{27}$$

This inequality provides a lower bound on the primal action for any choice of s and we do not assume strong duality here.

Residual optimality
Now suppose there exists a potential s_θ and a feasible triplet $(\rho_t, v_{\text{corr}}, g)$ such that

$$r_{s_\theta}(t, x) = 0 \quad \text{for } \rho_t\text{-a.e.}, \tag{28}$$

$$v_{\text{corr}} = \nabla_x s_\theta, \quad g = s_\theta, \tag{29}$$

and the boundary constraints $\rho_{|t=0} = \rho_0$, $\rho_{|t=1} = \rho_1$ hold.

Then the Fenchel–Young inequalities are equalities and equation 26 becomes:

$$\mathcal{A}(\rho, v_{\text{corr}}, g) = \mathbb{E}_{x\sim\rho_0}[s_\theta(0, x)] - \mathbb{E}_{x\sim\rho_1}[s_\theta(1, x)] - \int_0^1 \int_\Omega \rho_t(x)\, r_{s_\theta}(t, x)\, dx\, dt. \tag{30}$$

Since $r_{s_\theta} = 0$ ρ-a.e., then we have:

$$\mathcal{A}(\rho, v_{\text{corr}}, g) = \mathbb{E}_{x\sim\rho_0}[s_\theta(0, x)] - \mathbb{E}_{x\sim\rho_1}[s_\theta(1, x)]. \tag{31}$$

The vanishing of the residual r_{s_θ} together with feasibility of $(\rho_t, v_{\text{corr}}, g)$ implies stationarity of the Lagrangian. Therefore, $(\rho_t, v_{\text{corr}}, g)$ satisfies the first-order optimality conditions of the prior-guided WFR action.

In particular,

$$v^*_{\text{corr}} = \nabla_x s_\theta, \qquad g^* = s_\theta,$$

are optimal, and $(\rho_t, v^*_{\text{corr}}, g^*)$ satisfies as a primal optional solution of the prior-guided the WFR optimality conditions defined in Definition 3.2. □

C. Importance Reweighting

The squared HJB residual can be dominated by a few high-variance outliers (rare cells, sharp local flows), which destabilizes training. To ensure training stability and preventing these extreme high residual outliers, we adopt a simple batch-wise importance reweighting that down-weights large residuals. For a mini-batch $\{(t_i, x_i)\}_{i=1}^{B}$, let $r_i = \left| r_\theta(t_i, x_{t,i}) \right| + \varepsilon$ and with a temperature $\tau > 0$. Then for each sample, the weight is inversely proportional to a temperature-shaped residual as $\tilde{w}_i \propto r_i^{-\tau}$. Thus, larger residuals get smaller weight, which reduces variance while keeping the update focused and stable.

D. Further Discussions

On the velocity prior quality and assumptions The velocity prior v_{prior} indeed plays a constructive—but double-edged role in our method. A good prior captures coarse dynamics - reducing the learning complexity and improving on sample efficiency. A mis-specified prior can bias the learned corrective field $s_\theta(t, x)$ and slow or destabilize training. Hence, the *quality* of v_{prior} strongly influences both optimization and generalization. In practice, mild perturbation of the prior through noise, scale or mis-specification are corrected by s_θ, whereas severe mis-specification such as overly large or structurally wrong drifts can bias the learned corrective flow. Moreover, v_{prior} does *not* require divergence-free assumption. Our dual objective explicitly includes the cross term $\nabla_x s_\theta \cdot v_{\text{prior}}$ in the HJB residual avoiding hidden orthogonality requirements.

For single–cell datasets we currently use local supervision—training on adjacent pairs with a time–continuous shared network. This choice is simple and scalable and induces a globally smooth field, but it does not jointly enforce all marginals as in recent multi–marginal methods, which may limit long–range trajectory coherence. In future works, extending VP–HJF with global consistency could improve on long-range trajectory inference.

E. Additional Experiments and details

Balanced case - Gaussian Translation We define an affine prior $v_{\text{prior}}(x) = \mu_1 - \mu_0$ with the parameters $\mu_0 = (0, 0)$, $\mu_1 = (0.5, 6.0)$, $\Sigma_0 = ((2.5, 0.0), (0.0, 0.3))$, $\Sigma_1 = ((0.4, 0.0), (0.0, 2.2))$. This prior captures the dominant translation between the two Gaussian distributions.

Balanced case - Rotating Ring This experiment refers to Figure 1 (left two). First, we show a case when utilizing the velocity prior is crucial in learning the correct velocity field where curl-free methods like AM fails. We tested on a 2D rotating ring dataset where the points on the ring (source) are rotated by a fixed angle θ (target). The velocity prior is defined as $v_{\text{prior}} = \omega J x$, where J is the skew-symmetric rotation matrix and $x \in \mathbb{R}^2$, so this becomes $Jx = [0 \;\; 1;\; -1 \;\; 0][x_1 \;\; x_2]^\top = (x_2, -x_1)^\top$. The task for our model v_{ot} is to learn the residual correction after given the prior rotation knowledge, such as ensuring the boundary condition by aligning the mismatched source and target density or correcting the radial drift by push the points inwards or outwards,etc. In Figure 1 left (b), we show that since AM has the curl-free limitation, without a prior, its model $\nabla s_t(x)$ failed to represent pure rotation where the streamlines cut through the circle. In Figure 1 left (a), the streamlines from our method form a circular flow indicating that v_{prior} gives the model the correct inductive bias.

Diverging Petal This experiment refers to Figure 1 (right two). We created a curved and rotated petal-shape dataset to test our method on diverging multi-trajectory paths. The source is a gaussian distribution concentrated in the center and $v_{\text{prior}=\omega Jx}$ the rotation dynamic defined as before. Our task is to learn v_{petal}, a radial and angle-dependent term that push outward the points along the radius with different speed depend on the angle $\theta = \text{atan2}(y, x)$. We have

$$v_{\text{petal}}(x) \;=\; s(\theta)\,\hat{r}, \quad \hat{r} = \tfrac{x}{\|x\|}, \quad s(\theta) = \max\big(0,\; b + a\cos(k\theta)\big) \tag{32}$$

Compared with the petal shape appeared in AM and MIOFlow (Huguet et al., 2022), the underlying dynamic flow in our example is harder to learn, where the former one has a straight-axis aligned radial expansion as $r(x) = |x_1| + |x_2|$ with the gradient of $r(x)$ being a piece-wise constant and curl-free. Figure 1 middle shows that our petal shape matches with the target shape. Figure 1 right shows that the vector field (red arrows) are bending away from pure rotation (blue arrows) to align strongly with the petal shape by pushing the mass outward.

Lotka-Volterra with grwoth The total mass $M(t) = \mathbb{E}[w(t)]$ and the ground-truth mass ratio is calculated as $r_{\text{true}} = M(t)/M(0)$. We define the v_{prior} as prior oracle LV drift with added Gaussian noise with no growth term.

Table 8. Comparison on the EB dataset using W_2 at the held-out marginals (t_1, t_3). Baseline results other than * are taken from (Theodoropoulos et al., 2025).

Method	W_2 t_1	W_2 t_3
DeepRUOT	13.64	15.10
Var-RUOT*	10.34	12.02
MIOFlow	13.66	14.05
SBIRR	15.42	20.98
MMFM	14.68	14.83
DMSB	14.83	15.49
3MSBM	14.51	13.26
VP-HJF (ODE)*	11.94	12.28
VP-HJF (Strang)*	10.68	12.57

Table 9. Comparison on the EB dataset using weighted W_1 at marginals t_1–t_4 for Var-RUOT and VP-HJF.

	W_1^{weighted}			
Method	t_1	t_2	t_3	t_4
Var-RUOT*	10.28	11.58	11.90	13.28
VP-HJF (ODE)*	11.14	12.45	13.14	14.45
VP-HJF (Strang)*	11.37	12.43	14.36	14.14

EB scRNA-Seq data We conducted further comparison on the EB dataset with 100-dim using the $\mathcal{W}_2$ metric at the held-out marginals at t_1, t_3. Table 8 shows that our approach remains competitive and outperforms most methods. We then compare more directly with VAR-RUOT in Table 9 using the weighted $\mathcal{W}_1$, since both methods are in the unbalanced optimal transport setting. In this evaluation, we assign non-uniform particle weights by integrating the learned dynamics through ODE integration and use these predicted weights when computing $\mathcal{W}_1$, instead of uniform masses.

On this weighted $\mathcal{W}_1$ metric, Var-RUOT achieves slightly lower values than VP–HJF. This gap could partly due to the use of noisy RNA-velocity as v_{prior} in our framework, which can trade a small increase in transport cost for better agreement with the measured dynamics. In addition, Var-RUOT optimizes a single global-in-time trajectory via SDE simulations, whereas VP–HJF relies on deterministic ODE rollouts with local per-interval supervision.

E.1. Further training details on single-cell datasets

For all the single-cell datasets, we use a 4-layer MLP with Swish activation. The MLP outputs follows the Action Matching implementation, where the output $h_\theta(t, x)$ needs to multiply by the original data input x so the scaler output becomes $s = (h \times x).sum()$ Moreover, We optimize the model with Adam and set the learning rate to $1e-4$ for all datasets. We uses multi-marginal training in Algorithm 1. Training was conducted in a mixture of CPU and a single GPU. For computational cost comparisons, the experiments were conducted on the CPU for training time per iteration.

On Strang splitting implementation For training and evaluating VP-HJF, we integrate the learned particle dynamics using a fixed-step Strang splitting scheme governed by

$$\dot{x}(t) = v_{\text{prior}}(t, x) + \nabla_x s_\theta(t, x), \qquad \dot{\log w}(t) = s_\theta(t, x),$$

where v_{prior} is provided by RNA velocity estimation and $\nabla_x s_\theta$ is the learned correction.

Given an initial batch of particles x_0, we discretize the time interval $[0, 1]$ into K uniform steps of size $\Delta t = 1/K$. Each Strang step follows below subflows: (i) a half-step under the prior drift v_{prior}, (ii) a full step under the corrective gradient field $\nabla_x s_\theta$, and (iii) a second half-step under the prior drift. The corrective substep is integrated using a second-order Runge–Kutta (RK2) update evaluated at the midpoint in time. In practice, we use a small number of steps (e.g., $K = 4$) for terminal reconstruction during training, which is sufficient due to the smoothness induced by the prior-guided formulation.

KNN-based velocity prior implementation To obtain a continuous-time velocity prior $v_{\text{prior}}(t, x)$ from discrete snapshots

dataset, we use a local k-nearest neighbor (KNN) estimator. Let $X \in \mathbb{R}^{N\times d}$ denote the reference cell and $V \in \mathbb{R}^{N\times d}$ as the corresponding RNA velocity vectors. Given a sampled x_t, we compute pairwise Euclidean distances to reference points (instead of the entire dataset, we randomly choose large proportion of points) and select k nearest neighbors (we select top 30). We then estimate the velocity at x_t as a Gaussian-weighted average of the neighbors' velocities,

$$v_{\mathrm{knn}}(x_t) \;=\; \sum_{j\in\mathcal{N}_k(x_t)} w_j\, V_j, \qquad w_j \;\propto\; \exp\!\left(-\frac{\|x_t - X_j\|^2}{2h^2}\right),$$

where h is an adaptive bandwidth set to the distance of the k-th nearest neighbor.

During training, we sample intermediate points x_t by linear interpolation between OT-paired samples (x_0, x_1) at randomly sampled times $t \sim \mathcal{U}(0, 1)$,

$$x_t = (1-t)\, x_0 + t\, x_1,$$

and evaluate the KNN-based velocity $v_{\mathrm{knn}}(x_t)$ at these locations. Additionally, we also define a time-dependent decay for the velocity prior as

$$v_{\mathrm{prior}}(t, x) = (1-t)\, v_{\mathrm{knn}}(x),$$

This decay can gradually diminish the influence of the RNA velocity as trajectories approach terminal states, which helps to stabilize integration as well.

For EB, we trained 300 epoch and 256 batch size, using dopri5 with 16 steps for the ode integration. For Strang splitting, we used 4 steps. We set the HJB loss coefficients to $\lambda_{\mathrm{hjb}} = 0.01$, sliced Wasserstein loss coefficient $\lambda_{\mathrm{sw}} = 10$ and the mass loss coefficient to $\lambda_{\mathrm{mass}} = 0.01$. Additionally, since normalized RNA velocity is extremely small, we applied a scaler of 100 to v_{prior}.

For the Var-RUOT baseline on EB, we use the authors' publicly released implementation and configuration, changing the training epoch to match ours but kept the coefficient weights as is.

For Mouse erythroid, we trained for 3000 iterations and batch size is 256, using Euler with 32 steps for the ode integration. For Strang splitting, we used 4 steps. For 2-dim, we set the HJB loss coefficients to $\lambda_{\mathrm{hjb}} = 0.5$, sliced Wasserstein loss coefficient $\lambda_{\mathrm{sw}} = 1$ and the mass loss coefficient to $\lambda_{\mathrm{mass}} = 0.01$. Similarly to Curly-FM, we also applied a scaler to the RNA velocity. We set the scaler to 90 for the v_{prior}. For 20-dim, we set the HJB loss coefficients to $\lambda_{\mathrm{hjb}} = 0.01$, sliced Wasserstein loss coefficient $\lambda_{\mathrm{sw}} = 100$ and the mass loss coefficient to $\lambda_{\mathrm{mass}} = 0.01$. We set the scaler to 0.01 for the v_{prior}.

For comparisons with CURLY-FM and OT-FM, we used the author's publicly released repo to re-run the experiments and calculated the runtime.

For bone-marrow, we trained for 3000 iterations and batch size is 256, using Euler with 16 steps for the ode integration. For Strang splitting, we used 4 steps. We trained on 30-dim, where we used PCA to project the 30-dim onto 2D for the UMAP as shown in Figure 3. For 30-dim training, we set the HJB loss coefficients to $\lambda_{\mathrm{hjb}} = 1e-3$, sliced Wasserstein loss coefficient $\lambda_{\mathrm{sw}} = 50$ and the mass loss coefficient to $\lambda_{\mathrm{mass}} = 0.1$.