
Inferring stochastic dynamics with growth from cross-sectional data

Stephen Zhang[†]
School of Mathematics and Statistics,
University of Melbourne

Suryanarayana Maddu
Center for Computational Biology,
Flatiron Institute

Xiaojie Qiu
Department of Genetics,
Stanford University School of Medicine

Victor Chardès[†]
Center for Computational Biology,
Flatiron Institute

Abstract

Time-resolved single-cell omics data offers high-throughput, genome-wide measurements of cellular states, which are instrumental to reverse-engineer the processes underpinning cell fate. Such technologies are inherently destructive, allowing only cross-sectional measurements of the underlying stochastic dynamical system. Furthermore, cells may divide or die in addition to changing their molecular state. Collectively these present a major challenge to inferring realistic biophysical models. We present a novel approach, *unbalanced* probability flow inference, that addresses this challenge for biological processes modelled as stochastic dynamics with growth. By leveraging a Lagrangian formulation of the Fokker-Planck equation, our method accurately disentangles drift from intrinsic noise and growth. We showcase the applicability of our approach through evaluation on a range of simulated and real single-cell RNA-seq datasets. Comparing to several existing methods, we find our method achieves higher accuracy while enjoying a simple two-step training scheme.

1 Introduction

Single-cell measurement technologies offer an unbiased, single-cell resolution view of the molecular processes dictating cell fate. These methods predominantly rely on microfluidic devices to capture and read molecules in individual cells, destroying the cell in the measurement process. Therefore the state of any cell can be measured only at a single time. Experimental study of temporal biological processes must therefore proceed by time-course studies, in which a time-series of population snapshots is obtained. At each time-point, a measurement is obtained from a distinct population of cells, assumed to be identically prepared at the initial time. In practice these data can be obtained by serial sampling from a larger population, or from biological replicates measured at different times [1, 2]. The challenge with time-resolved population snapshot data is that direct observation of dynamics in terms of longitudinal information is lost, and must be reconstructed from static population profiles. This motivates the inverse problem of reconstructing an underlying dynamical system which not only interpolates the data, but also captures the key biophysical phenomenon at play in the regulation of gene expression, notably intrinsic noise and cellular proliferation. On the other hand, ignoring these aspects can result in incorrect identification of regulatory mechanism [3, 4].

While the task of learning stochastic dynamics interpolating high-dimensional distributions is now routinely solved in generative modelling tasks [5, 6], doing so while ensuring faithfulness to biophysically relevant features like intrinsic molecular noise as well as division and death remains a

[†]Corresponding authors: syz@syz.id.au, vchardes@flatironinstitute.org.

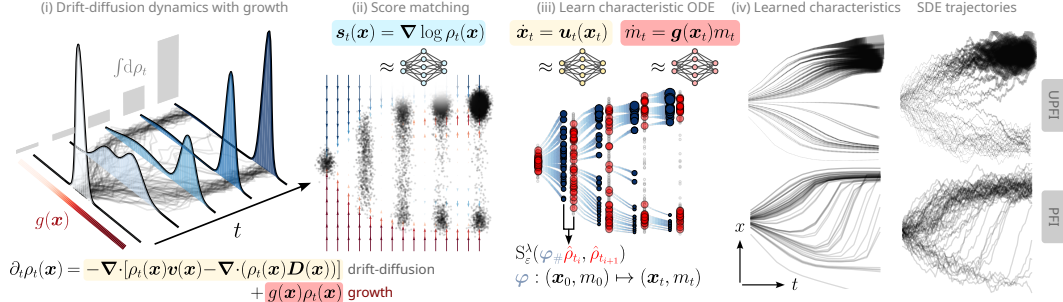


Figure 1: **Overview.** (i) Stochastic population dynamics with growth, governed by a Fokker-Planck equation with source. (ii) Score matching trains a neural network to model the contribution of noise. (iii) Neural-ODE based learning of the Fokker-Planck characteristics. (iv) Learned characteristics and corresponding SDE trajectories.

challenge. Similarly, while the dynamical inference problem has seen significant interest from the single cell analysis community, the issues of the noise and growth remain open for the most part [3, 4, 7, 8]. For instance, most existing methods assume either no noise [9] or a constant isotropic diffusivity [2, 10], and either no growth [11] or the availability of prior information such as growth rate estimates [2] or lineage tracing [4, 12].

With this in mind, a flexible framework called *probability flow inference* (PFI) was recently introduced [13, 14] as a tool to infer stochastic dynamical models accounting for intrinsic noise. Leveraging the Lagrangian formulation of the Fokker-Planck equation, also known as the *probability flow* formulation [15], PFI infers the drift of an Itô diffusion interpolating between the distributions at successive times. A limitation of PFI is that it does not account for growth (i.e. proliferation and death), a key feature of realistic biological systems that must be appropriately modelled for the inference of accurate trajectories. This was notably observed in reprogramming experiments, where neglecting cellular death led to the inference of incorrect state transitions between apoptotic and pluripotent cells [2]. In this paper, we show how the PFI approach naturally extends to account for cellular proliferation, and we propose a new algorithm called *unbalanced probability flow inference* (UPFI), which allows to disentangle drift from intrinsic noise and growth in diffusion processes.

Stochastic population dynamics with growth We consider a population of individuals evolving following an Itô process of the form

$$d\mathbf{X}_t = \mathbf{v}_t(\mathbf{X}_t) dt + \boldsymbol{\sigma}_t(\mathbf{X}_t) d\mathbf{B}_t, \quad (1)$$

with $d\mathbf{B}_t$ the increments of a d -dimensional Brownian motion, $\mathbf{v}_t(\mathbf{X}_t)$ the drift encoding deterministic aspects of the system and $\boldsymbol{\sigma}_t(\mathbf{X}_t)$ the strength of the noise, which can depend on the state \mathbf{X}_t , as well as the drift.

To include growth in the model (1), we model individuals as undergoing *division* and *death* at rates $b_t(\mathbf{X}_t)$ and $d_t(\mathbf{X}_t)$, respectively. More precisely, during a short time interval $\tau \ll 1$, an individual with state \mathbf{X}_t divides with probability $b_t(\mathbf{X}_t)\tau + o(\tau)$ and dies with probability $d_t(\mathbf{X}_t)\tau + o(\tau)$. Upon a division event, both descendants inherit their parent's state \mathbf{X}_t . With these prescriptions, the Itô process (1), along with the division and death mechanism, defines a branching diffusion process [16]. The density of individuals with state \mathbf{x} at time t , denoted $\rho_t(\mathbf{x})$, follows a Fokker-Planck equation (FPE) with a source term [12]:

$$\partial_t \rho_t(\mathbf{x}) = -\nabla \cdot [\rho_t(\mathbf{x})\mathbf{v}_t(\mathbf{x}) - \nabla \cdot (\rho_t(\mathbf{x})\mathbf{D}_t(\mathbf{x}))] + g_t(\mathbf{x})\rho_t(\mathbf{x}), \quad (2)$$

where $\mathbf{D}_t(\mathbf{x}) = \frac{1}{2}\boldsymbol{\sigma}_t(\mathbf{x})\boldsymbol{\sigma}_t(\mathbf{x})^\top \in \mathbb{S}_+^d$ is the diffusivity matrix and $g_t(\mathbf{x}) = b_t(\mathbf{x}) - d_t(\mathbf{x})$ is the net fitness associated with the gene expression \mathbf{x} . Regions where $g_t(\mathbf{x}) < 0$ correspond to regions of net death, while regions $g_t(\mathbf{x}) > 0$ correspond to net growth. The total number of cells $|\rho_t| = \int \rho_t(\mathbf{x}) d\mathbf{x}$ is thus not conserved over time, and ρ_t is not a probability density. In Fig. 1(i) we illustrate the evolution of ρ_t for a bistable process in two dimensions, where growth has the effect of accumulating the mass on one of the two branches, even though both branches are equally favourable in terms of the energetic landscape. In Section 3, we will analyse in more detail a high-dimensional version of this bifurcating process.

To motivate the relevance of (1) for cell dynamics, we point out that a SDE of this form with coupled drift and noise terms naturally arises from the Chemical Langevin Equation, obtained by coarse-graining of the biophysically principled Chemical Master Equation [17]. The importance of multiplicative noise in biological systems is also pointed out by [3] which highlights the nontrivial impact that complex noise models can have on dynamics. Finally, multiplicative noise models also arise in non-biological settings such as the Cox-Ingersoll-Ross model in finance [18].

On the other hand, the importance of modelling division and death is readily apparent for applications such as cell dynamics, ecology, and disease modelling [19, 20]. In biological systems, organised cell division and death are a key control mechanism that work alongside regulatory dynamics, and are thus an essential component that must be modelled to ensure accurate inference results.

Problem statement Population snapshot measurements arise in settings where tracking of individuals is either impossible or expensive, such as in single-cell RNA sequencing (scRNA-seq) studies. Longitudinal information on the stochastic trajectories of individuals and their progenitors is lost. Instead, one has access only to *statistically independent*, cross-sectional measurements of the population density $\rho_t(\mathbf{x})$. In other words, we observe the *temporal marginals* of the process (1) with birth and death. Given the knowledge of $\rho_t(\mathbf{x})$, our goal is to infer the drift \mathbf{v}_t , the fitness g_t and the diffusivity $\mathbf{D}_t(\mathbf{x})$ of the underlying branching diffusion process, or in other words, to invert (2). While our approach applies to the simultaneous inference of these three quantities, we assume that $\mathbf{D}_t(\mathbf{x})$, or the form of its functional dependency on the drift in the case of a Chemical Langevin Equation [17], is known. In what follows, we restrict ourselves to inferring the drift and the fitness.

Consider $K + 1$ cross-sectional measurements taken from (2) at successive times $t_0 = 0 < \dots < t_K = T$. Each measurement i consists of N_i sampled states $\{\mathbf{x}_{k,t_i}, 1 \leq k \leq N_i\}$. We assume that N_i/N_0 estimates the ratio of individuals $|\rho_{t_i}|/|\rho_0|$ at all times t_i , $1 \leq i \leq K$. With this assumption, we can estimate ρ_{t_i} up to a multiplicative constant $|\rho_0|/N_0$ using the samples available: $\rho_{t_i} \simeq \frac{|\rho_0|}{N_0} \sum_{j=1}^{N_i} \delta(\mathbf{x} - \mathbf{x}_{j,t_i})$. Practically, the knowledge of this constant term is unnecessary because the rescaled density $N_0\rho_t/|\rho_0|$ satisfies the same equation as ρ_t so we set it to one.

Related work Approaches based on optimal transport for reconstructing dynamics typically either ignore cellular birth and death entirely [11, 13, 14, 21], or handle growth using an unbalanced relaxation of entropic optimal transport (EOT) and prior knowledge on growth [2, 9, 10, 12, 22]. In particular, while unbalanced EOT can easily be solved computationally using a modified Sinkhorn algorithm [23], it proceeds via a relaxation of the static transport problem and does not readily offer a natural dynamical interpretation in the sense of the continuous-time dynamics of (2). Recently a significant amount of theoretical progress was made on the closely related Schrödinger bridge problem for branching Brownian motion [16], however designing computational approaches for its solution in high dimensional settings remains an open problem. Several recent works aim to learn neural approximations to dynamics with growth, but either assume deterministic dynamics [24], a global fitness linking drift to growth [25] (see also the discussion in Appendix C.3), or rely in the end on unbalanced EOT for conditioning flow matching [26]. These methods are therefore subject to limitations rendering them inapplicable for the problem we describe. One work that addresses our problem setting is [27] which proposes a deep learning based method, DeepRUOT. While the algorithm is designed with the same aims as ours, it requires a multi-stage training procedure prone to instability, first relying on flow matching and then on simulation-based training.

2 Methodology

Unbalanced Probability Flow Inference (UPFI) To solve this inverse problem we build upon the Probability Flow Inference (PFI) approach of [13, 14], in which the Fokker-Planck equation is fitted in the Lagrangian frame of reference. This formulation leverages the fact that the drift-diffusion term in (2) can be re-written as a transport term. In our setting, such a rewriting reads

$$\partial_t \rho_t(\mathbf{x}) = -\nabla \cdot [\rho_t(\mathbf{x}) (\mathbf{v}_t(\mathbf{x}) - \nabla \cdot \mathbf{D}_t(\mathbf{x}) - \mathbf{D}_t(\mathbf{x}) \nabla \log \rho_t(\mathbf{x}))] + g_t(\mathbf{x}) \rho_t(\mathbf{x}), \quad (3)$$

The phase-space velocity in the transport term now depends on the score $\nabla \log \rho_t(\mathbf{x})$, which is independent of the total mass of the measure since $\nabla \log \rho_t = \nabla \log (\rho_t/|\rho_t|)$. Interestingly, (3) can be solved in the same Lagrangian frame of reference as without growth $g_t = 0$. Indeed, the

solution (\mathbf{x}_t, m_t) of the following system of ODEs

$$\begin{aligned} \frac{d\mathbf{x}_t}{dt} &= \mathbf{v}_t(\mathbf{x}_t) - \nabla \cdot \mathbf{D}_t(\mathbf{x}_t) - \mathbf{D}_t(\mathbf{x}_t) \nabla \log \rho_t(\mathbf{x}_t) =: \mathbf{u}_t(\mathbf{x}_t), & \mathbf{x}_0 &\sim \rho_0, \\ \frac{dm_t}{dt} &= g_t(\mathbf{x}_t) m_t, & m_0 &= \rho_0(\mathbf{x}_0), \end{aligned} \quad (4)$$

produces a weak solution to (2) [28, Proposition 2.1], where we have written \mathbf{u}_t to denote the phase-space velocity field. In other terms, by going to the Lagrangian frame of reference, we can trade solving a PDE in d dimensions with solving an ODE system in $d + 1$ dimensions. This gain comes at a cost, since it requires learning the score function $\nabla \log \rho_t(\mathbf{x})$ beforehand. However, this can efficiently be done offline for high-dimensional datasets because the score is independent of the parameters of the dynamical model in (4).

We remark on the importance that (4) yields a *weak* solution to (2) – solving this system for finitely many particles yields a weighted sum of point masses $\hat{\rho}_t$ approximating the measure $\rho_t d\mathbf{x}$ but does not provide the *density* ρ_t . The density evolution is governed by $d(\log \rho_t(\mathbf{x}_t))/dt = g_t(\mathbf{x}_t) - (\nabla \cdot \mathbf{u}_t)(\mathbf{x}_t)$, i.e. an additional term arises from the divergence of the phase-space velocity. While this is the setting arising in the continuous normalising flows literature [29] and is also used by [24] in the context of growth, in practice computing the divergence term is well-known to be computationally expensive as it relies on computing the trace of a Jacobian.

Our approach, UPFI, consists of two steps: (i) estimating a time-dependent score function $\mathbf{s}_t(\mathbf{x}) \simeq \nabla \log \rho_t(\mathbf{x})$ from available snapshots, and (ii) learning the coefficients of the ODE system (4) that fit the observed snapshots $\{\rho_{t_i}, 1 \leq i \leq K\}$. For the first step of the algorithm, we estimate the score with denoising score matching [5, 30], as it has shown best performance in terms of accuracy and scalability on similar problems [14]. For the second step, we push the observed samples and their associated mass from time t_i to time t_{i+1} following (4). The explicit update equations read

$$\hat{\mathbf{x}}_{k,t} = \mathbf{x}_{k,t_i} + \int_{t_i}^t (\mathbf{v}_u(\hat{\mathbf{x}}_{k,u}) - \nabla \cdot \mathbf{D}_u(\hat{\mathbf{x}}_{k,u}) - \mathbf{D}_u(\hat{\mathbf{x}}_{k,u}) \mathbf{s}_u(\hat{\mathbf{x}}_{k,u})) du \quad (5)$$

$$m_{k,t} = \exp \left(\int_{t_i}^t g_u(\hat{\mathbf{x}}_{k,u}) du \right), \quad (6)$$

with $t \in [t_i, t_{i+1}]$. This allows us to construct an inferred marginal distribution $\hat{\rho}_{t_{i+1}}$, which reads

$$\hat{\rho}_{t_{i+1}}(\mathbf{x}) = \sum_{k=1}^{N_i} m_{k,t_{i+1}} \delta(\hat{\mathbf{x}}_{k,t_{i+1}} - \mathbf{x}). \quad (7)$$

We can then optimise jointly over the drift and the growth by minimising a discrepancy $\mathcal{D}(\hat{\rho}_{t_{i+1}}, \rho_{t_{i+1}})$ between the inferred and the true measures. In practice, we minimise the total discrepancy across all snapshots, $(\mathbf{v}_t^*(\mathbf{x}), g_t^*(\mathbf{x})) = \arg \min_{(\mathbf{v}_t(\mathbf{x}), g_t(\mathbf{x}))} \sum_{i=1}^K \mathcal{D}(\hat{\rho}_{t_i}, \rho_{t_i})$. While in principle various choices of \mathcal{D} could be appropriate, we opt to use the unbalanced Sinkhorn divergence [31] which possesses desirable geometric and computational properties [23, 32]: $\mathcal{D} = S_{\varepsilon, \gamma}$, where $\varepsilon, \gamma > 0$ specify the entropic regularisation level and soft mass constraint respectively. Most importantly, this choice works directly with discrete measures, allowing us to bypass entirely computation of densities.

We summarise the UPFI algorithm in Alg. 1. These successive steps are illustrated for a two-dimensional bifurcating process in Fig. 1(ii)-(iii). In Fig. 1(iv) we show the difference between the resulting probability flow lines inferred with PFI and with UPFI. Because PFI doesn't explicitly account for proliferation, it infers flow lines which correct for the mass imbalance between the two bifurcating branches by incorrectly connecting them. On the other hand, UPFI recovers the appropriate flow lines which do not connect the two bifurcating branches. The inference of a drift biased by proliferation, as exemplified with PFI in Fig. 1D, is an instance of a wider issue regarding the ability to infer jointly drift of proliferation.

The problem of identifying simultaneously drift and fitness Even in the absence of cellular proliferation, the drift term $\mathbf{v}_t(\mathbf{x})$ is in general not uniquely identifiable [14, 33, 34]. Naturally, these same arguments extend to the simultaneous inference of the drift and growth, and furthermore we can expect to mistake drift for growth and vice versa. While the approach outlined in Algorithm 1 applies to any nonlinear force field, here we aim to gain theoretical insight into the identifiability problem by analytically studying an Ornstein-Uhlenbeck (OU) process with quadratic fitness, in which the population density remains a Gaussian measure. Although OU processes can not capture bifurcations, they have been successfully used to infer non-bifurcating processes in single-cell RNA-seq data [35]. For these processes, the drift is a linear function $\mathbf{v}_t(\mathbf{x}) = \mathbf{A}_t \mathbf{x} + \mathbf{e}_t$ and the matrix

Algorithm 1 Unbalanced Probability Flow Inference (UPFI)

Input: $K + 1$ statistically independent snapshots $\{\mathbf{x}_{k,t_i}\}_{k=1}^{N_i}$ sampled from $\{\rho_{t_i}\}_{i=0}^K$, i.e. the solution to (2) at successive times t_0, t_1, \dots, t_K .
Estimate score: $\mathbf{s}_\phi(t, \mathbf{x}) \approx \nabla \log \rho_t(\mathbf{x})$ using score matching
Initialise: Force $\mathbf{v}_\theta(t, \mathbf{x})$, growth rate $\mathbf{g}_\theta(t, \mathbf{x})$, diffusivity $\mathbf{D}(t, \mathbf{x})$.
while not converged **do**
 for $k \in \{0, 1, \dots, K - 1\}$ **do**
 $(\mathbf{x}_{\ell,t_{k+1}}, m_{\ell,t_{k+1}})_{\ell=1}^{N_k} \leftarrow \text{odesolve}(\text{Eq. (4)}, \mathbf{x}_0 = (\mathbf{x}_{\ell,t_k}, m_{\ell,t_k})_{\ell=1}^{N_k}, \mathbf{ts} = [t_k, t_{k+1}])$
 $\hat{\rho}_{t_{k+1}} \leftarrow \sum_{\ell=1}^{N_k} m_{\ell,t_{k+1}} \delta(\mathbf{x} - \mathbf{x}_{\ell,t_{k+1}})$ *Predicted marginal at t_{k+1}*
 $\ell_k \leftarrow S_{\varepsilon, \gamma}(\rho_{t_{k+1}}, \hat{\rho}_{t_{k+1}})$ *Sinkhorn loss on marginals*
 $\mathcal{L} = K^{-1} \sum_{k=1}^{K-1} \ell_k$ *Compute total loss*
 $\theta \leftarrow \text{Update}(\theta, \nabla_\theta \mathcal{L})$ *Update θ*
 return \mathbf{v}_θ

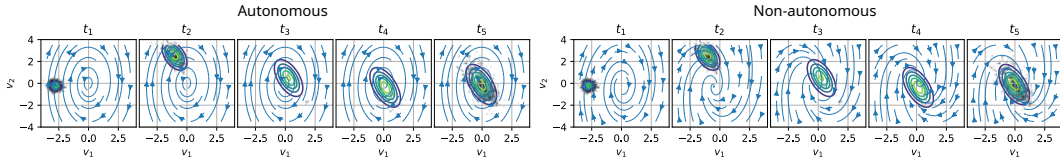


Figure 2: **Non-identifiability of growth and drift in the Gaussian case.** 8-dimensional OU process with quadratic growth. Left: autonomous drift and growth produces the same marginals as a non-autonomous system.

\mathbf{A}_t is directly interpretable as the gene regulatory network driving the process. We prove in the appendix the following result for OU processes with quadratic fitness.

Proposition 2.1 (OU process with quadratic fitness). *Consider an OU process with quadratic fitness, whose density satisfies*

$$\partial_t \rho_t(\mathbf{x}) = -\nabla \cdot ((\mathbf{A}_t \mathbf{x} + \mathbf{e}_t) \rho_t(\mathbf{x}) - \mathbf{D}_t \nabla \rho_t(\mathbf{x})) + (b_t + \mathbf{c}_t^\top \mathbf{x} + \frac{1}{2} \mathbf{x}^\top \mathbf{\Gamma}_t \mathbf{x}) \rho_t(\mathbf{x}), \quad (8)$$

where $\mathbf{A}_t \in \mathbb{R}^{d \times d}$, $\mathbf{e}_t, \mathbf{c}_t \in \mathbb{R}^d$, and $b_t \in \mathbb{R}$ are generic, $\mathbf{D}_t \in \mathbb{R}^{d \times d}$ is symmetric positive definite, and $\mathbf{\Gamma}_t \in \mathbb{R}^{d \times d}$ is symmetric negative semi-definite. If $\rho_0 = m_0 \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ with $m_0 > 0$, then $\rho_t = m_t \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$ for all $t \geq 0$, where

$$\dot{\boldsymbol{\Sigma}}_t = \mathbf{A}_t \boldsymbol{\Sigma}_t + \boldsymbol{\Sigma}_t \mathbf{A}_t^\top + 2\mathbf{D}_t + \boldsymbol{\Sigma}_t \mathbf{\Gamma}_t \boldsymbol{\Sigma}_t, \quad (9)$$

$$\dot{\boldsymbol{\mu}}_t = (\mathbf{A}_t + \boldsymbol{\Sigma}_t \mathbf{\Gamma}_t) \boldsymbol{\mu}_t + \boldsymbol{\Sigma}_t \mathbf{c}_t + \mathbf{e}_t, \quad (10)$$

$$\frac{\dot{m}_t}{m_t} = \frac{1}{2} \boldsymbol{\mu}_t^\top \mathbf{\Gamma}_t \boldsymbol{\mu}_t + \mathbf{c}_t^\top \boldsymbol{\mu}_t + b_t + \frac{1}{2} \text{tr}(\mathbf{\Gamma}_t \boldsymbol{\Sigma}_t). \quad (11)$$

Using this result, we can observe that it is impossible to identify the gene regulatory network uniquely, and that many solutions fit equally well the data. This is made concrete in the following result.

Corollary 2.2. *Consider an OU process with drift $\mathbf{v}_t(\mathbf{x}) = \mathbf{A}_t \mathbf{x}$ and a time-dependent fitness as in (8). Let $\mathbf{K}_t \in \mathbb{R}^{n \times n}$ be an arbitrary matrix. Then there exists $\epsilon > 0$ such that the system is indistinguishable from another OU process with drift $\mathbf{v}_t(\mathbf{x}) = \mathbf{A}_t + \mathbf{I} + \epsilon \mathbf{K}_t$ and time-dependent quadratic fitness.*

Importantly, this result holds even if we enforce the drift to be autonomous: both symmetric and asymmetric parts of the drift matrix can be confounded with cellular growth, and vice-versa. We show in Fig. 2 a simple illustration of this fact – the same sequence of distributions can arise from autonomous linear drift with quadratic growth, but also non-autonomous linear drift alone.

Loss function and uniqueness of the minimum The identifiability issue discussed previously calls for the use of a regularisation when optimising for the drift $\mathbf{v}_t(\mathbf{x})$. This regularisation ensures that, even if the method is not consistent in general, it leads at least to a unique solution. While different regularisers could be used, here we opt for the Wasserstein-Fisher-Rao energy function

[32] as a natural choice. The loss function reads, with $\alpha > 0, \lambda \geq 0$:

$$L = \sum_{i=1}^K S_{\varepsilon, \gamma}(\hat{\rho}_{t_i}, \rho_{t_i}) + \lambda(t_i - t_{i-1}) \int_{t_{i-1}}^{t_i} \int (\|\mathbf{v}_t(\mathbf{x})\|^2 + \alpha |g_t(\mathbf{x})|^2) d\rho_t(\mathbf{x}) dt. \quad (12)$$

Once again, we can gain some insight into the role of the regularisation by studying the linear-quadratic case from Prop. 2.1. Assuming $t_i - t_{i-1} = \Delta t$, we show in the appendix that in the non-entropic limit $\varepsilon \rightarrow 0$ (in which case the Sinkhorn divergence becomes the Gaussian Hellinger-Kantorovich distance [36]) and when $\Delta t \rightarrow 0$ with $T = K\Delta t$ fixed, the loss tends to a functional which has a unique minimum as a function of the parameters defining the fitness and the drift.

Theorem 2.3 (Loss function for OU processes with quadratic fitness). *Consider the true process as well as the inferred process to both be OU processes with quadratic fitness, i.e. (8). Denoting $q = 2/\gamma$, with $K\Delta t$ fixed, when $\Delta t \rightarrow 0$ we have $\Delta t^{-1}L \rightarrow \mathcal{L}$, where \mathcal{L} is the continuous time loss function:*

$$\mathcal{L} = q^{-1} \int_0^T dt m_t \left(\|\mathbf{v}_t - \hat{\mathbf{v}}_t\|_{\mathbf{X}_t^{-1}}^2 + \frac{1}{2} (h_t - \hat{h}_t)^2 \right) \quad (13)$$

$$+ q \sum_{i,j} \frac{\sigma_{i,t}^2 \sigma_{j,q,t}^2}{(\sigma_{j,q,t}^2 \sigma_{i,t}^2 + \sigma_{i,q,t}^2 \sigma_{j,t}^2)^2} \left(\mathbf{w}_{i,t}^\top (\mathbf{B}_t - \hat{\mathbf{B}}_t) \mathbf{w}_{j,t} \right)^2 + \lambda \int_0^T R_t dt \quad (14)$$

where $\Sigma_t = \sum_i \sigma_{i,t}^2 \mathbf{w}_{i,t} \mathbf{w}_{i,t}^\top$ is the eigendecomposition of the covariance Σ_t , $\sigma_{i,q,t}^2 = 1 + q\sigma_{i,t}^2$ for all i , $\mathbf{X}_t = 2\Sigma_t + q^{-1}$. We also have

$$\begin{aligned} \mathbf{B}_t - \hat{\mathbf{B}}_t &= \Sigma_t (\mathbf{A}_t - \hat{\mathbf{A}}_t)^\top + (\mathbf{A}_t - \hat{\mathbf{A}}_t) \Sigma_t + \Sigma_t (\mathbf{\Gamma}_t - \hat{\mathbf{\Gamma}}_t) \Sigma_t + 2(\mathbf{D}_t - \hat{\mathbf{D}}_t), \\ \mathbf{v}_t - \hat{\mathbf{v}}_t &= ((\mathbf{A}_t - \hat{\mathbf{A}}_t) + \Sigma_t (\mathbf{\Gamma}_t - \hat{\mathbf{\Gamma}}_t)) \boldsymbol{\mu}_t + \Sigma_t (\mathbf{c}_t - \hat{\mathbf{c}}_t) + (\mathbf{e}_t - \hat{\mathbf{e}}_t), \\ h_t - \hat{h}_t &= \frac{1}{2} \boldsymbol{\mu}_t^\top (\mathbf{\Gamma}_t - \hat{\mathbf{\Gamma}}_t) \boldsymbol{\mu}_t + (\mathbf{c}_t - \hat{\mathbf{c}}_t)^\top \boldsymbol{\mu}_t + (b_t - \hat{b}_t) + \frac{1}{2} \text{tr}((\mathbf{\Gamma}_t - \hat{\mathbf{\Gamma}}_t) \Sigma_t), \end{aligned} \quad (15)$$

and R_t is a strongly convex function of the parameters defining the fitness and the drift. For $\lambda > 0$, \mathcal{L} has a unique minimiser $t \mapsto (\hat{\mathbf{A}}_t^*, \hat{\mathbf{e}}_t^*, \hat{b}_t^*, \hat{\mathbf{c}}_t^*, \hat{\mathbf{\Gamma}}_t^*)$.

Scalability The unbalanced Sinkhorn divergence $S_{\varepsilon, \gamma}$ can be computed with a per-iteration complexity of $\mathcal{O}(B^2)$ and a dimension-independent sample complexity of $\mathcal{O}(B^{-1/2})$ [31], provided the batch size B is sufficiently large. In contrast, denoising score matching has a per-iteration complexity of $\mathcal{O}(Bd)$, where d denotes the dimensionality. The computational cost of ODE integration can be reduced, at the expense of some accuracy, by taking fewer Euler steps between time points. In practice, we find that two to three steps are typically sufficient. This imposes a mild limitation on the scalability to high dimensions, and we find the UPFI algorithm can handle up to moderately high dimensions as illustrated in the results section below. This moderately high-dimensional regime is particularly relevant for modelling cellular processes such as haematopoietic stem cell differentiation, where only a few dozen key transcription factors drive cell fate decisions [1]. Additionally, PCA dimensionality reduction is a routine preprocessing step in single cell analysis workflows [37].

3 Results

High-dimensional bifurcating system We first consider a bistable system in \mathbb{R}^d where the dynamics are driven by a potential landscape $\mathbf{v} = -\nabla V$, $V(\mathbf{x}) = 0.9\|\mathbf{x} - \mathbf{a}\|_2^2 \|\mathbf{x} - \mathbf{b}\|_2^2 + 10 \sum_{i=3}^d x_i^2$, where attractors are located at $\mathbf{a}, \mathbf{b} = \pm(\mathbf{e}_1 + \mathbf{e}_2)$. We impose a birth rate $b(\mathbf{x}) = \frac{5}{2}(1 + \tanh(2x_0))$ and $d(\mathbf{x}) = 0$ so that individuals closer to the positive attractor divide at a high rate (see schematic in Fig. 1(i)). For $d = \{2, 5, 10, 25, 50\}$ we simulate the system and sample population snapshots at $T = 5$ timepoints (see Fig. 3(a) for $d = 10$). To reconstruct the dynamics, we apply UPFI (Alg. 1), as well as PFI [13, 14]. For additional comparisons, we consider a principled deterministic fitness-based method (fitness-ODE, see Section C.3), a method inspired by TIGON [24], DeepRUOT [27], optimal transport flow matching (OTFM) [38] and its unbalanced variant, UOTFM [26]. The implementation of TIGON as described and provided by [24] is difficult to use in practice, we therefore re-implemented the method with several simplifications which we believe improves its performance,

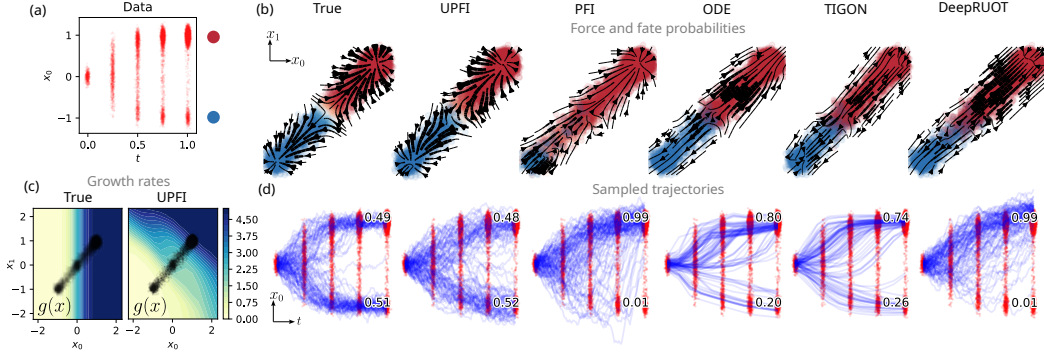


Figure 3: **10-dimensional bistable system.** (a) Population snapshots over time, shown in the first coordinate x_0 . (b) True and inferred force fields shown in (x_0, x_1) , coloured by fate probabilities. (c) True and inferred growth rates shown in (x_0, x_1) . (d) Sampled trajectories from true and learned dynamics with growth suppressed (pure drift-diffusion process). The fraction of trajectories terminating in the upper (resp. lower) regions are indicated. Without growth both branches are equiprobable.

d	UPFI	PFI	FITNESS-ODE	TIGON++	DEEPRUOT	OTFM	UOTFM
Path energy distance (\downarrow)							
2	0.14 \pm 0.09	1.41 \pm 0.16	0.30 \pm 0.18	0.46 \pm 0.12	2.15 \pm 0.01	1.16 \pm 0.13	0.42 \pm 0.13
5	0.04 \pm 0.03	1.34 \pm 0.06	0.30 \pm 0.14	0.63 \pm 0.16	0.47 \pm 0.04	1.07 \pm 0.11	0.36 \pm 0.10
10	0.05 \pm 0.04	1.03 \pm 0.18	0.29 \pm 0.15	0.61 \pm 0.06	1.32 \pm 0.05	1.09 \pm 0.19	0.38 \pm 0.08
25	0.22 \pm 0.20	1.51 \pm 0.10	0.20 \pm 0.04	0.74 \pm 0.04	0.14 \pm 0.00	1.39 \pm 0.09	0.57 \pm 0.22
50	0.15 \pm 0.02	1.89 \pm 0.21	0.20 \pm 0.08	0.62 \pm 0.04	0.31 \pm 0.02	1.38 \pm 0.06	0.69 \pm 0.14
Force error (cosine \downarrow)							
2	0.07 \pm 0.02	0.08 \pm 0.01	0.34 \pm 0.04	0.26 \pm 0.04	0.44 \pm 0.00	0.41 \pm 0.03	0.44 \pm 0.06
5	0.15 \pm 0.04	0.09 \pm 0.01	0.36 \pm 0.02	0.37 \pm 0.03	0.44 \pm 0.00	0.44 \pm 0.01	0.46 \pm 0.03
10	0.10 \pm 0.00	0.12 \pm 0.00	0.37 \pm 0.05	0.35 \pm 0.02	0.45 \pm 0.00	0.45 \pm 0.01	0.45 \pm 0.01
25	0.06 \pm 0.00	0.09 \pm 0.00	0.19 \pm 0.04	0.27 \pm 0.01	0.37 \pm 0.00	0.60 \pm 0.01	0.63 \pm 0.02
50	0.06 \pm 0.00	0.07 \pm 0.00	0.26 \pm 0.03	0.25 \pm 0.01	0.44 \pm 0.00	0.55 \pm 0.01	0.52 \pm 0.01
Force error (L^2, \downarrow)							
2	1.17 \pm 0.03	1.81 \pm 0.09	1.70 \pm 0.04	1.62 \pm 0.03	2.84 \pm 0.00	1.77 \pm 0.06	1.68 \pm 0.04
5	2.89 \pm 0.11	3.53 \pm 0.18	3.78 \pm 0.02	3.83 \pm 0.04	4.07 \pm 0.00	3.86 \pm 0.01	3.80 \pm 0.03
10	4.31 \pm 0.08	4.76 \pm 0.14	5.87 \pm 0.13	5.89 \pm 0.04	6.17 \pm 0.00	6.09 \pm 0.04	5.96 \pm 0.01
25	6.74 \pm 0.05	6.95 \pm 0.05	8.94 \pm 0.20	9.58 \pm 0.03	9.98 \pm 0.00	10.32 \pm 0.02	10.28 \pm 0.04
50	9.45 \pm 0.06	9.41 \pm 0.03	13.37 \pm 0.22	13.43 \pm 0.04	14.45 \pm 0.00	14.60 \pm 0.04	14.49 \pm 0.01
Pearson fate correlation, \uparrow)							
2	0.98 \pm 0.01	0.57 \pm 0.04	0.94 \pm 0.01	0.91 \pm 0.03	0.51 \pm 0.00	0.79 \pm 0.01	0.96 \pm 0.01
5	0.99 \pm 0.00	0.59 \pm 0.02	0.93 \pm 0.01	0.82 \pm 0.04	0.84 \pm 0.00	0.76 \pm 0.01	0.95 \pm 0.00
10	0.99 \pm 0.00	0.65 \pm 0.02	0.93 \pm 0.01	0.84 \pm 0.03	0.77 \pm 0.00	0.76 \pm 0.02	0.95 \pm 0.00
25	0.97 \pm 0.01	0.65 \pm 0.02	0.90 \pm 0.02	0.77 \pm 0.01	0.95 \pm 0.00	0.74 \pm 0.02	0.93 \pm 0.01
50	0.98 \pm 0.00	0.63 \pm 0.03	0.91 \pm 0.01	0.80 \pm 0.02	0.92 \pm 0.00	0.73 \pm 0.01	0.93 \pm 0.01

Table 1: Numerical evaluation results for d -dimensional bistable system, $d \in \{2, 5, 10, 25, 50\}$.

and we thus refer to it as TIGON++ (see Section C.4 for in-depth discussion). We provide details on flow matching in Section C.6.

Fig. 3(b) shows in the (x_0, x_1) plane: (i) the vector field v_t as a streamplot, and (ii) the *fate probabilities* by colour, obtained from the true and reconstructed dynamics. We define the fate probability of a state (t_i, x) towards an attractor \mathcal{A} as the frequency of sampled trajectories starting from (t_i, x) that terminate closest to that attractor. (see Section C for full experiment details)

We observe that UPFI recovers vector fields and fate probabilities that almost perfectly resemble the ground truth. On the other hand PFI, unable to account for the presence of growth, misattributes the shift in distribution to a spurious drift towards the faster dividing branch. On the other hand, fitness-ODE and TIGON++ are able to pinpoint the bifurcation but produce binary fate probabilities since they are deterministic. Furthermore, the inferred vector fields are qualitatively different from the ground truth for the same reason. DeepRUOT, which in theory models both growth and stochasticity, is unable to recover good dynamics. This is potentially owing to instabilities in their training scheme,

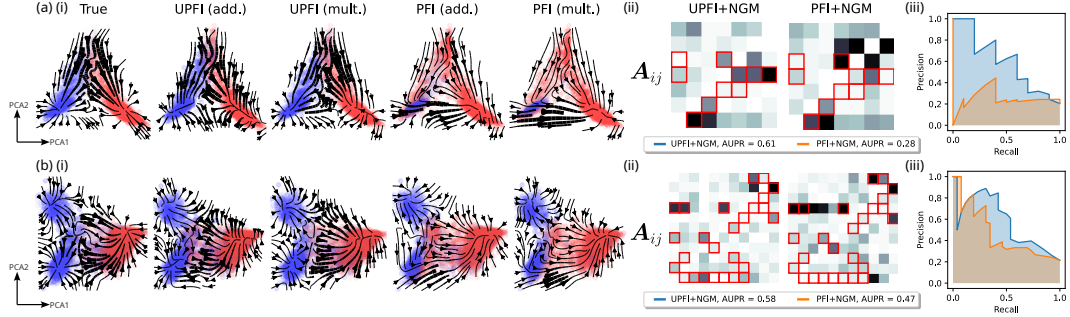


Figure 4: **Simulated regulatory networks.** (a) (i) True and inferred vector fields for 7-dimensional bifurcating system, coloured by fate probabilities. (ii) Learned causal graphs using neural graphical model within UPFI (PFI) frameworks with true interactions shown in red. (iii) Precision-recall curve quantification of prediction accuracy. (b)(i-iii) Same as (a) for 11-dimensional HSC system.

	UPFI (add.)	UPFI (mult.)	PFI (add.)	PFI (mult.)
Bifurcating				
Path energy distance (\downarrow)	1.22 \pm 0.38	1.69 \pm 0.14	1.91 \pm 1.04	4.62 \pm 1.14
Force error (L^2 , \downarrow)	12.59 \pm 0.39	11.78 \pm 0.30	18.96 \pm 3.25	16.49 \pm 1.62
Pearson fate correlation (\uparrow)	0.97 \pm 0.00	0.98 \pm 0.00	0.66 \pm 0.17	0.62 \pm 0.06
Haematopoietic				
Path energy distance (\downarrow)	0.99 \pm 0.26	0.98 \pm 0.15	1.42 \pm 0.42	1.86 \pm 0.30
Force error (L^2 , \downarrow)	19.58 \pm 0.33	18.13 \pm 0.25	26.80 \pm 3.71	26.84 \pm 1.48
Fate prediction error (TV \downarrow)	0.08 \pm 0.00	0.07 \pm 0.00	0.23 \pm 0.03	0.27 \pm 0.04

Table 2: Numerical evaluation results for CLE systems.

which involves multiple distinct stages and losses [27, Algorithm 1] compared to the two-step UPFI approach (Alg. 1). Visualising the true and inferred growth rates (Fig. 3(c)) we find good agreement. Finally in Fig. 3(d) we show sampled trajectories over time for the true system and each method, and these further illustrate our conclusions from before. Because of space constraints, we show sampled trajectories for OTFM and UOTFM in Fig. 6 in the Appendix.

In Table 1 we show numerical evaluation metrics for each method as d varies; here we also compare to results for OTFM and UOTFM. At the level of trajectories, we quantify the *energy distance* [39] between sampled paths from the ground truth and inferred processes treated as empirical distributions in $L^2([0, 1], \mathbb{R}^d)$. At the level of the force, we measure the reconstruction error on v_t in terms of the L^2 and cosine distances. We also report the Pearson correlation between the true and estimated fate probabilities. In all cases, we find that UPFI recovers dynamics with high accuracy, while remaining competitive in terms of runtime, as show in Table 7. Finally, in Table 8 we also present regularisation ablation experiments which show limited effect of the growth regularisation term α . We interpret this as the evidence of implicit regularisation stemming from relatively small neural network sizes.

Simulated gene regulatory networks Next we study more complex systems where dynamics arise from chemical reaction networks via the Chemical Langevin Equation (CLE) [14, 40]: we consider a 7-gene bifurcating system (Fig. 4(a)) and a 11-gene haematopoietic stem cell (HSC) system (Fig. 4(b)) in which the simulated dynamics reflect known biology [41]. Cells in the bifurcating system branch and randomly proceed to one of two stable states. The HSC system is multifurcating and cell states evolve towards one of four stable states, {Monocyte (Mo), Granulocyte (G), Megakaryocyte (Mk), Erythrocyte (E)}. Dynamics in both sys-

	UPFI	UPFI (Jac)	PFI	PFI (Jac)
Bifurcating				
AUPR (\uparrow)	0.64 \pm 0.03	0.52 \pm 0.07	0.33 \pm 0.09	0.34 \pm 0.10
AUROC (\uparrow)	0.79 \pm 0.02	0.74 \pm 0.03	0.65 \pm 0.07	0.63 \pm 0.08
Haematopoietic				
AUPR (\uparrow)	0.59 \pm 0.04	0.58 \pm 0.02	0.53 \pm 0.05	0.51 \pm 0.04
AUROC (\uparrow)	0.80 \pm 0.02	0.79 \pm 0.01	0.73 \pm 0.02	0.70 \pm 0.02

Table 3: Accuracy of directed graph prediction using neural graphical model for CLE systems.

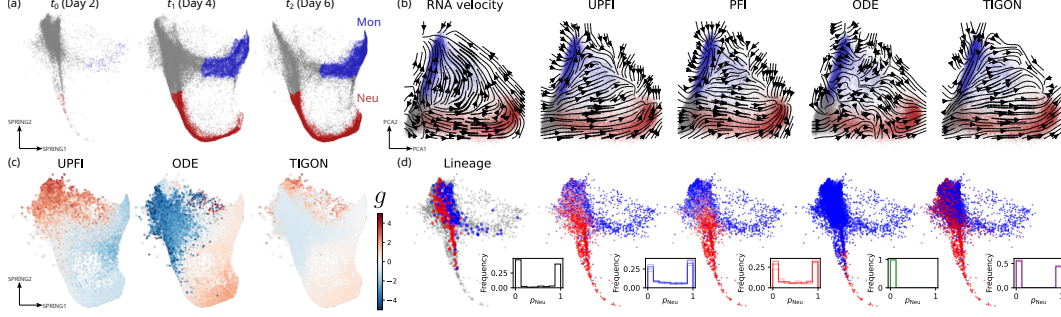


Figure 5: **Monocyte-neutrophil development.** (a) Temporal snapshots of monocyte-neutrophil fate determination, shown using SPRING coordinates and celltype annotations from the original publication. (b) Learned vector fields: RNA velocity vector field learned from spliced-unspliced data, all others from temporal snapshots. Cosine distance (relative to RNA velocity field) for Mon, Neu cells shown. (c) Learned growth rates. (d) Fate probabilities empirically estimated from lineage tracing data and predicted from learned dynamics. Pearson correlation (relative to lineage tracing data) shown.

tems are of the form

$$d\mathbf{X}_t = (\mathbf{u}(\mathbf{X}_t) - \mathbf{v}(\mathbf{X}_t)) dt + \frac{1}{\sqrt{V}} \sqrt{\mathbf{u}(\mathbf{X}_t) + \mathbf{v}(\mathbf{X}_t)} d\mathbf{B}_t. \quad (16)$$

where \mathbf{u}, \mathbf{v} specify production and degradation rates and $V > 0$ is the system volume. In particular, the CLE involves a *multiplicative* noise model where the diffusivity varies with \mathbf{x} . This is in contrast to the additive noise setting of Fig. 3 where the diffusivity is fixed. For the bifurcating system, we specify a birth rate $b_t(\mathbf{x})$ causing cells in one branch to divide faster than others. In the HSC system, we specify b_t similarly so that cells in the Erythrocyte branch divide faster. We then apply UPFI and PFI: for each inference method, we consider both *additive* and *multiplicative* noise models

$$d\mathbf{X}_t = \mathbf{v}_\theta(\mathbf{x}) dt + \sigma d\mathbf{B}_t, \quad (\text{add.})$$

$$d\mathbf{X}_t = (\mathbf{f}_\theta(\mathbf{x}) - \mathbf{g}_\theta(\mathbf{x})) dt + \sigma \sqrt{\mathbf{f}_\theta(\mathbf{x}) + \mathbf{g}_\theta(\mathbf{x})} d\mathbf{B}_t. \quad (\text{mult.})$$

We show in Fig. 4(a, b)(i) the learned vector field for the bifurcating and HSC systems for each method and model, and highlight fate probabilities towards the fast growing branch. We find PFI in general infers a spurious force towards the fast-growing branch as in the previous example and consequently incorrect fate probabilities. On the other hand, UPFI with both additive and multiplicative noise models produce qualitatively similar results that resemble the ground truth. Quantitatively we confirm this in Table 2, and also find that UPFI with the multiplicative noise model mostly outperforms the additive noise model by a small but consistent margin.

Since UPFI and PFI work with a neural parameterisation of the force \mathbf{v} , this allows for plug-in use of interpretable architectures. To illustrate this, we make use of the Neural Graphical Model (NGM) architecture [42] from which a sparse and unsigned “causal graph” can be extracted and interpreted as the learned regulatory network. For simplicity we use the additive noise model, although use of NGM in the multiplicative noise case is also straightforward. We show in Fig. 4(a,b)(ii) the learned adjacency matrices along with the true interactions highlighted in red, and (iii) the precision-recall curves quantifying the network inference accuracy. In both cases UPFI recovers a more accurate network than PFI – see Table 3 for full evaluation results. This highlights that modelling of growth is essential to making accurate inferences on dynamics and thus gene interactions [2, 19], as well as the possibility for incorporating custom architectures within the UPFI approach.

Lineage-tracing single-cell RNA-seq dataset We now apply UPFI to an experimental time-course of monocyte-neutrophil development in vitro [1] across 3 timepoints over 4 days (Fig. 5). Additional information on the dynamics are available in two forms: (i) lineage tracing, where descendants of a common ancestor cell are distinguished by unique molecular labels (“barcodes”), and (ii) “RNA velocity” information [43], which provides partial information on the gene expression dynamics from relative abundances of spliced and unspliced RNA transcripts. This information, although limited, is independent from the population snapshots and so provides a point of comparison for the dynamics inferred from snapshots. We preprocess the data following standard pipelines

t	Celltype	UPFI	PFI	FITNESS-ODE	TIGON++
		Cosine distance (\downarrow)			
0	Monocyte	0.26 ± 0.01	0.33 ± 0.02	0.42 ± 0.04	0.30 ± 0.02
0	Neutrophil	0.27 ± 0.02	0.31 ± 0.01	0.44 ± 0.02	0.26 ± 0.01
1	Monocyte	0.26 ± 0.01	0.32 ± 0.01	0.46 ± 0.04	0.28 ± 0.02
1	Neutrophil	0.24 ± 0.01	0.28 ± 0.01	0.43 ± 0.01	0.22 ± 0.01
2	Monocyte	0.27 ± 0.01	0.36 ± 0.01	0.41 ± 0.03	0.33 ± 0.02
2	Neutrophil	0.31 ± 0.01	0.33 ± 0.01	0.38 ± 0.01	0.27 ± 0.01
Metric		Fate correlation (\uparrow)			
Kendall's τ		0.22 ± 0.03	0.07 ± 0.03	-0.02 ± 0.01	0.19 ± 0.04
Pearson's r		0.26 ± 0.04	0.09 ± 0.04	-0.02 ± 0.01	0.19 ± 0.05
Spearman's ρ		0.27 ± 0.04	0.08 ± 0.04	-0.03 ± 0.01	0.20 ± 0.05

Table 4: Evaluation results on lineage tracing dataset [1].

[43, 44] and use the leading 10 principal components to represent cell state. We apply UPFI with an additive noise model as well as several other methods to this data and show results for the learned drift alongside the RNA velocity field in Fig. 5(b). Since the magnitude of RNA velocity estimates are unreliable [45] we measure the cosine similarity between the RNA velocity field and the inferred force from snapshots (Table 4). We restrict this analysis to the monocyte and neutrophil cell clusters, corresponding to cells that are already committed to their respective lineages. We find that UPFI and TIGON++ perform comparably.

Inferred growth rates (Fig. 5(c)) show that UPFI predicts that cells in the earlier progenitor states have a higher division rate, which is consistent with the expected biology [1, 24]. On the other hand, the fitness-ODE yields the opposite while TIGON++ predicts a very low growth rate for most cells. The availability of lineage tracing data across timepoints provide us with a ground truth for measuring accuracy of fate predictions (Fig. 5): fate probabilities predicted by UPFI agree with the lineage tracing data more closely than other methods. In particular, PFI and fitness-ODE exhibit a bias towards monocyte lineage. While TIGON++ is able to predict a mixture of both fates, as in the bistable system it produces only deterministic outcomes. Similarly, quantitative results (Table 4) show that UPFI performs best in several different fate correlation metrics.

Finally, we note that this dataset was also studied in [24, 27] which introduced TIGON and DeepRUOT respectively. However, in these papers all analyses of this dataset were carried out using two-dimensional embeddings computed by a force-directed layout algorithm [1]. The setting we consider (10-dimensional PCA embeddings) is more challenging, and also a more realistic scenario since nonlinear dimensionality reduction methods are known to introduce distortions and are thus unsuitable for downstream quantitative analysis [46].

4 Discussion

We study the problem of inferring stochastic dynamics from cross-sectional snapshots of a population subject to drift, diffusion as well as birth and death. To this end we propose UPFI, a method based on the probability flow characterisation of the Fokker-Planck equation. Our approach is flexible enough to handle generic noise models as well as growth, and crucially applies to the practically important scenario where growth rates are unknown and must be learned together with the drift. We provide a theoretical treatment in the linear-quadratic case, demonstrating the nature of the non-identifiability that arises from growth. We conclude by showcasing our method's efficacy using both simulated and experimental single cell data.

There are overall several lessons to be summarised from our work, which may be of relevance more generally to the problems of reconstructing dynamics from snapshots. As shown by our theoretical analysis and supported by results, in the fully general setting where the drift and growth components are non-autonomous, there is little reason to expect to accurately separate drift from growth effects. In the absence of additional information to aid with inference, we advocate to impose an autonomous drift and possibly also autonomous growth field. This is consistent with cell-autonomous models of biological dynamics, i.e. ignoring cell-cell interactions or temporally varying environments. Subject to the modelling limitations incurred by these assumptions, we argue that this serves as a strong inductive bias that can help inference.

References

- [1] C Weinreb, A Rodriguez-Fraticelli, FD Camargo, AM Klein, Lineage tracing on transcriptional landscapes links state to fate during differentiation. *Science* 367, eaaw3381 (2020).
- [2] G Schiebinger, et al., Optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming. *Cell* 176, 928–943 (2019).
- [3] MA Coomer, L Ham, MP Stumpf, Noise distorts the epigenetic landscape and shapes cell-fate decisions. *Cell Systems* 13, 83–102 (2022).
- [4] B Bonham-Carter, G Schiebinger, Cellular proliferation biases clonal lineage tracing and trajectory inference. *Bioinformatics* 40, btae483 (2024).
- [5] Y Song, S Ermon, Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems* 32 (2019).
- [6] Y Song, et al., Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456* (2020).
- [7] EB Gunnarsson, J Foo, K Leder, Statistical inference of the rates of cell proliferation and phenotypic switching in cancer. *Journal of Theoretical Biology* 568, 111497 (2023).
- [8] DR Sisan, M Halter, JB Hubbard, AL Plant, Predicting rates of cell state change caused by stochastic fluctuations using a data-driven landscape model. *Proceedings of the National Academy of Sciences* 109, 19262–19267 (2012).
- [9] A Tong, J Huang, G Wolf, D Van Dijk, S Krishnaswamy, Trajectorynet: A dynamic optimal transport network for modeling cellular dynamics in *International conference on machine learning*. (PMLR), pp. 9526–9536 (2020).
- [10] GHT Yeo, SD Saksena, DK Gifford, Generative modeling of single-cell time series with pre-scient enables prediction of cell trajectories with interventions. *Nature communications* 12, 3222 (2021).
- [11] T Hashimoto, D Gifford, T Jaakkola, Learning population-level diffusions with generative rnns in *International Conference on Machine Learning*. (PMLR), pp. 2417–2426 (2016).
- [12] E Ventre, et al., Trajectory inference for a branching sde model of cell differentiation. *arXiv preprint arXiv:2307.07687* (2023).
- [13] V Chardès, S Maddu, MJ Shelley, Stochastic force inference via density estimation. *arXiv preprint arXiv:2310.02366* (2023).
- [14] S Maddu, V Chardès, MJ Shelley, Inferring biological processes with intrinsic noise from cross-sectional data. *arXiv preprint arXiv:2410.07501* (2025).
- [15] MS Albergo, NM Boffi, E Vanden-Eijnden, Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797* (2023).
- [16] A Baradat, H Lavenant, Regularized unbalanced optimal transport as entropy minimization with respect to branching brownian motion. *arXiv preprint arXiv:2111.01666* (2021).
- [17] R Grima, P Thomas, AV Straube, How accurate are the nonlinear chemical fokker-planck and chemical langevin equations? *The Journal of chemical physics* 135 (2011).
- [18] JC Cox, JE Ingersoll, SA Ross, , et al., A theory of the term structure of interest rates. *Econometrica* 53, 385–407 (1985).
- [19] DS Fischer, et al., Inferring population dynamics from single-cell rna-sequencing time series data. *Nature biotechnology* 37, 461–468 (2019).
- [20] M Pariset, YP Hsieh, C Bunne, A Krause, V De Bortoli, Unbalanced diffusion schrödinger bridge. *arXiv preprint arXiv:2306.09099* (2023).

- [21] C Bunne, L Papaxanthos, A Krause, M Cuturi, Proximal optimal transport modeling of population dynamics in *International Conference on Artificial Intelligence and Statistics*. (PMLR), pp. 6511–6528 (2022).
- [22] L Chizat, S Zhang, M Heitz, G Schiebinger, Trajectory inference via mean-field langevin in path space. *Advances in Neural Information Processing Systems* 35, 16731–16742 (2022).
- [23] L Chizat, G Peyré, B Schmitzer, FX Vialard, Scaling algorithms for unbalanced optimal transport problems. *Mathematics of computation* 87, 2563–2609 (2018).
- [24] Y Sha, Y Qiu, P Zhou, Q Nie, Reconstructing growth and dynamic trajectories from single-cell transcriptomics data. *Nature Machine Intelligence* 6, 25–39 (2024).
- [25] K Neklyudov, R Brekelmans, D Severo, A Makhzani, Action matching: Learning stochastic dynamics from samples in *International conference on machine learning*. (PMLR), pp. 25858–25889 (2023).
- [26] L Eyring, et al., Unbalancedness in neural monge maps improves unpaired domain translation. *arXiv preprint arXiv:2311.15100* (2023).
- [27] Z Zhang, T Li, P Zhou, Learning stochastic dynamics from snapshots through regularized unbalanced optimal transport. *arXiv preprint arXiv:2410.00844* (2024).
- [28] A Chertock, A practical guide to deterministic particle methods in *Handbook of numerical analysis*. (Elsevier) Vol. 18, pp. 177–202 (2017).
- [29] W Grathwohl, RT Chen, J Bettencourt, I Sutskever, D Duvenaud, Ffjord: Free-form continuous dynamics for scalable reversible generative models. *arXiv preprint arXiv:1810.01367* (2018).
- [30] P Vincent, A connection between score matching and denoising autoencoders. *Neural computation* 23, 1661–1674 (2011).
- [31] T Séjourné, J Feydy, FX Vialard, A Trounev, G Peyré, Sinkhorn divergences for unbalanced optimal transport. *arXiv preprint arXiv:1910.12958* (2019).
- [32] L Chizat, G Peyré, B Schmitzer, FX Vialard, An interpolating distance between optimal transport and fisher-rao metrics. *Foundations of Computational Mathematics* 18, 1–44 (2018).
- [33] G Schiebinger, Reconstructing developmental landscapes and trajectories from single-cell data. *Current Opinion in Systems Biology* 27, 100351 (2021).
- [34] C Weinreb, S Wolock, BK Tusi, M Socolovsky, AM Klein, Fundamental limits on dynamic inference from single-cell snapshots. *Proceedings of the National Academy of Sciences* 115, E2467–E2476 (2018).
- [35] L Wang, et al., Dictys: dynamic gene regulatory network dissects developmental continuum with single-cell multiomics. *Nature Methods* 20, 1368–1378 (2023).
- [36] H Janati, Ph.D. thesis (Institut Polytechnique de Paris) (2021).
- [37] MD Luecken, FJ Theis, Current best practices in single-cell rna-seq analysis: a tutorial. *Molecular systems biology* 15, e8746 (2019).
- [38] A Tong, et al., Improving and generalizing flow-based generative models with minibatch optimal transport. *arXiv preprint arXiv:2302.00482* (2023).
- [39] ML Rizzo, GJ Székely, Energy distance. *Wiley interdisciplinary reviews: Computational statistics* 8, 27–38 (2016).
- [40] A Pratapa, AP Jalihal, JN Law, A Bharadwaj, T Murali, Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. *Nature methods* 17, 147–154 (2020).
- [41] J Krumsiek, C Marr, T Schroeder, FJ Theis, Hierarchical differentiation of myeloid progenitors is encoded in the transcription factor network. *PloS one* 6, e22649 (2011).

- [42] A Bellot, K Branson, M van der Schaar, Neural graphical modelling in continuous-time: consistency guarantees and algorithms. *arXiv preprint arXiv:2105.02522* (2021).
- [43] X Qiu, et al., Mapping transcriptomic vector fields of single cells. *Cell* 185, 690–711 (2022).
- [44] FA Wolf, P Angerer, FJ Theis, Scanpy: large-scale single-cell gene expression data analysis. *Genome biology* 19, 1–5 (2018).
- [45] SC Zheng, G Stein-O'Brien, L Boukas, LA Goff, KD Hansen, Pumping the brakes on rna velocity by understanding and interpreting rna velocity estimates. *Genome biology* 24, 246 (2023).
- [46] T Chari, L Pachter, The specious art of single-cell genomics. *PLOS Computational Biology* 19, e1011288 (2023).
- [47] DL Lukes, DL Lukes, *Differential equations: classical to controlled*, Mathematics in science and engineering. (Academic Press, New York), (1982).
- [48] S Särkkä, A Solin, *Applied Stochastic Differential Equations*, Institute of Mathematical Statistics Textbooks. (Cambridge University Press, Cambridge), (2019).
- [49] JE Potter, Massachusetts Institute of Technology, United States, *A matrix equation arising in statistical filter theory*, NASA CR- 7270. (National Aeronautics and Space Administration, Washington, D.C.), (1965).
- [50] GA Seber, AJ Lee, *Linear Regression Analysis*. (John Wiley & Sons), (2003).
- [51] B Dacorogna, *Direct Methods in the Calculus of Variations*, Applied Mathematical Sciences eds. F John, JE Marsden, L Sirovich. (Springer, Berlin, Heidelberg) Vol. 78, (1989).
- [52] J Feydy, et al., Interpolating between optimal transport and mmd using sinkhorn divergences in *The 22nd International Conference on Artificial Intelligence and Statistics*. pp. 2681–2690 (2019).
- [53] TO Gallouët, L Monsaingeon, A jko splitting scheme for kantorovich–fisher–rao gradient flows. *SIAM Journal on Mathematical Analysis* 49, 1100–1130 (2017).

A General results for Ornstein-Uhlenbeck processes

A.1 OU process with affine fitness

Proposition A.1. *Consider an OU process with affine fitness, whose density satisfies*

$$\partial_t p_t = -\nabla \cdot ((\mathbf{A}_t \mathbf{x} + \mathbf{e}_t) p_t - \mathbf{D}_t \nabla p_t) + (b_t + \mathbf{c}_t^\top \mathbf{x}) p_t. \quad (17)$$

where $\mathbf{A}_t \in \mathbb{R}^{d \times d}$, $\mathbf{e}_t, \mathbf{c}_t \in \mathbb{R}^d$, and $b_t \in \mathbb{R}$ are general, $\mathbf{D}_t \in \mathbb{R}^{d \times d}$ is symmetric positive definite. If $\rho_0 = m_0 \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$, then $\rho_t = m_t \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$ for all $t \geq 0$, where

$$\dot{\boldsymbol{\Sigma}}_t = \mathbf{A}_t \boldsymbol{\Sigma}_t + \boldsymbol{\Sigma}_t \mathbf{A}_t^\top + 2\mathbf{D}_t, \quad (18)$$

$$\dot{\boldsymbol{\mu}}_t = \mathbf{A}_t \boldsymbol{\mu}_t + \boldsymbol{\Sigma}_t \mathbf{c}_t + \mathbf{e}_t, \quad (19)$$

$$\frac{\dot{m}_t}{m_t} = \mathbf{c}_t^\top \boldsymbol{\mu}_t + b_t. \quad (20)$$

Proof. We denote $\hat{p}_t(\mathbf{k})$ the Fourier transform of $p_t(\mathbf{x})$. We have the following identities

$$\mathcal{F}(\nabla \cdot (\mathbf{A}_t \mathbf{x} + \mathbf{e}_t) p_t) = -\mathbf{k}^\top \mathbf{A}_t \nabla_{\mathbf{k}} \hat{p}_t + i \mathbf{e}_t^\top \mathbf{k} \hat{p}_t, \quad (21)$$

$$\mathcal{F}(\nabla \cdot (\mathbf{D}_t \nabla p_t)) = -\mathbf{k}^\top \mathbf{D}_t \mathbf{k} \hat{p}_t. \quad (22)$$

Such that the Fourier transform of the PDE reads

$$\partial_t \hat{p}_t = (\mathbf{k}^\top \mathbf{A}_t + i \mathbf{c}_t^\top) \nabla_{\mathbf{k}} \hat{p}_t - \mathbf{k}^\top \mathbf{D}_t \mathbf{k} \hat{p}_t + (b_t - i \mathbf{e}_t^\top \mathbf{k}) \hat{p}_t. \quad (23)$$

We define the characteristics as the solution of

$$\frac{d\mathbf{k}_s}{ds} = -\mathbf{A}_s^\top \mathbf{k}_s - i \mathbf{c}_s, \quad (24)$$

such that

$$\frac{d\hat{p}_s(\mathbf{k}_s)}{ds} = \left(\frac{d\mathbf{k}_s}{ds} \right)^\top \nabla_{\mathbf{k}} \hat{p}_s(\mathbf{k}_s) + \partial_s \hat{p}_s(\mathbf{k}_s) \quad (25)$$

$$= -\mathbf{k}_s^\top \mathbf{D}_s \mathbf{k}_s \hat{p}_s + (b_s - i \mathbf{e}_s^\top \mathbf{k}_s) \hat{p}_s. \quad (26)$$

Denote $\Phi(t, t_0)$ the state transition matrix solution to the homogeneous ODE (24) with initial condition t_0 . It exists because the coefficient are locally integrable on \mathbb{R}^+ [47]. The state transition matrix has the following properties [48]:

$$\frac{d\Phi(s, u)}{ds} = -\mathbf{A}_s^\top \Phi(s, u), \quad \frac{d\Phi(s, u)}{du} = \Phi(s, u) \mathbf{A}_s^\top, \quad (27)$$

$$\Phi(t, u) \Phi(u, s) = \Phi(t, s), \quad \Phi(t, s)^{-1} = \Phi(s, t), \quad \Phi(s, s) = \mathbf{I}. \quad (28)$$

Denote $\Phi(s, 0) = \Phi_s$ for simplicity, we have, using the initial condition \mathbf{k}_0 :

$$\mathbf{k}_s = \Phi_s \mathbf{k}_0 - i \int_0^s \Phi(s, u) \mathbf{c}_u du. \quad (29)$$

We introduce $\tilde{\mathbf{v}}_s = -\int_0^s \Phi(s, u) \mathbf{c}_u du$, such that we have the Fourier transform along the characteristics reads, with an initial Gaussian distribution

$$\begin{aligned} \hat{p}_s(\mathbf{k}_s) = \exp \left[-i \mathbf{k}_0^\top \mathbf{x}_0 - \frac{1}{2} \mathbf{k}_0^\top \boldsymbol{\Sigma}_0 \mathbf{k}_0 - \mathbf{k}_0^\top \int_0^s \Phi_u^\top \mathbf{D}_u \Phi_u du \mathbf{k}_0 - 2i \mathbf{k}_0^\top \int_0^s \Phi_u^\top \mathbf{D}_u \tilde{\mathbf{v}}_u du \right. \\ \left. + \int_0^s \tilde{\mathbf{v}}_u^\top \mathbf{D}_u \tilde{\mathbf{v}}_u du + \int_0^s b_u du - i \int_0^s \mathbf{e}_u^\top \Phi_u \mathbf{k}_0 + \int_0^s \mathbf{e}_u^\top \tilde{\mathbf{v}}_u du \right]. \end{aligned} \quad (30)$$

We introduce the following quantities

$$\boldsymbol{\Psi}_s = \int_0^s \Phi_u^\top \mathbf{D}_u \Phi_u du, \quad \mathbf{q}_s^\top = \int_0^s \mathbf{e}_u^\top \Phi_u du, \quad \mathbf{u}_s = \int_0^s \Phi_u^\top \mathbf{D}_u \tilde{\mathbf{v}}_u du, \quad (31)$$

$$\gamma_s = \int_0^s \tilde{\mathbf{v}}_u^\top \mathbf{D}_u \tilde{\mathbf{v}}_u du, \quad w_s = \int_0^s \mathbf{e}_u^\top \tilde{\mathbf{v}}_u du. \quad (32)$$

Inverting the equation for characteristics we have $\mathbf{k}_0 = \Phi_s^{-1} \mathbf{k}_s + i\Phi_s^{-1} \int_0^s \Phi(s, u) \mathbf{c}_u du = \Phi_s^{-1} \mathbf{k}_s - i\Phi_s^{-1} \tilde{\mathbf{v}}_s$. We can directly see that the Fourier transform is quadratic in \mathbf{k} , such that it is the Fourier transform of a Gaussian measure. By considering one after the other the terms $O(k^2)$, $O(k)$ and $O(1)$, we can find analytical expressions for the covariance, the mean and the mass of the Gaussian measure as a function of the state transition matrix. They read:

$$\Sigma_s = \Phi_s^{-T} \Sigma_0 \Phi_s^{-1} + 2\Phi_s^{-T} \Psi_s \Phi_s^{-1}, \quad (33)$$

$$\mu_s = \Phi_s^{-T} (\mu_0 + 2\mathbf{u}_s + \mathbf{q}_s) - \Sigma_s \tilde{\mathbf{v}}_s, \quad (34)$$

$$\log m_s = -\tilde{\mathbf{v}}_s^\top \mu_s - \frac{1}{2} \tilde{\mathbf{v}}_s^\top \Sigma_s \tilde{\mathbf{v}}_s + w_s + \gamma_s + \int_0^s b_u du. \quad (35)$$

Taking derivatives of these quantities and using the properties of the state transition matrix we recover the expected results. \square

A.2 OU process with quadratic fitness

Proposition 2.1 (OU process with quadratic fitness). *Consider an OU process with quadratic fitness, whose density satisfies*

$$\partial_t \rho_t(\mathbf{x}) = -\nabla \cdot ((\mathbf{A}_t \mathbf{x} + \mathbf{e}_t) \rho_t(\mathbf{x}) - \mathbf{D}_t \nabla \rho_t(\mathbf{x})) + (b_t + \mathbf{c}_t^\top \mathbf{x} + \frac{1}{2} \mathbf{x}^\top \Gamma_t \mathbf{x}) \rho_t(\mathbf{x}), \quad (8)$$

where $\mathbf{A}_t \in \mathbb{R}^{d \times d}$, $\mathbf{e}_t, \mathbf{c}_t \in \mathbb{R}^d$, and $b_t \in \mathbb{R}$ are generic, $\mathbf{D}_t \in \mathbb{R}^{d \times d}$ is symmetric positive definite, and $\Gamma_t \in \mathbb{R}^{d \times d}$ is symmetric negative semi-definite. If $\rho_0 = m_0 \mathcal{N}(\mu_0, \Sigma_0)$ with $m_0 > 0$, then $\rho_t = m_t \mathcal{N}(\mu_t, \Sigma_t)$ for all $t \geq 0$, where

$$\dot{\Sigma}_t = \mathbf{A}_t \Sigma_t + \Sigma_t \mathbf{A}_t^\top + 2\mathbf{D}_t + \Sigma_t \Gamma_t \Sigma_t, \quad (9)$$

$$\dot{\mu}_t = (\mathbf{A}_t + \Sigma_t \Gamma_t) \mu_t + \Sigma_t \mathbf{c}_t + \mathbf{e}_t, \quad (10)$$

$$\frac{\dot{m}_t}{m_t} = \frac{1}{2} \mu_t^\top \Gamma_t \mu_t + \mathbf{c}_t^\top \mu_t + b_t + \frac{1}{2} \text{tr}(\Gamma_t \Sigma_t). \quad (11)$$

Proof. Let's use the ansatz $p_t(\mathbf{x}) = u_t(\mathbf{x}) \exp[\mathbf{x}^\top \mathbf{G}_t \mathbf{x}/2]$ where \mathbf{G}_t is symmetric. The equation for u_t now reads

$$\begin{aligned} \partial_t u_t + \frac{1}{2} u_t \mathbf{x}^\top \frac{d\mathbf{G}_t}{dt} \mathbf{x} \\ = -\nabla \cdot ((\mathbf{A}_t \mathbf{x} + \mathbf{e}_t - 2\mathbf{D}_t \mathbf{G}_t \mathbf{x}) u_t - \mathbf{D}_t \nabla u_t) \\ + \left(b_t - \text{tr}(\mathbf{G}_t \mathbf{D}_t) + \mathbf{c}_t^\top \mathbf{x} - \mathbf{e}_t^\top \mathbf{G}_t \mathbf{x} + \mathbf{x}^\top (\frac{1}{2} \Gamma_t + \mathbf{G}_t \mathbf{D}_t \mathbf{G}_t - \mathbf{A}_t^\top \mathbf{G}_t) \mathbf{x} \right) u_t. \end{aligned} \quad (36)$$

The quadratic term vanishes when \mathbf{G}_t verifies for all \mathbf{x} the equation $\mathbf{x}^\top (\Gamma_t/2 + \mathbf{G}_t \mathbf{D}_t \mathbf{G}_t - \mathbf{A}_t^\top \mathbf{G}_t) \mathbf{x} = \frac{1}{2} \mathbf{x}^\top \dot{\mathbf{G}}_t \mathbf{x}$. This is verified for \mathbf{G}_t satisfying the matrix Riccati equation

$$\Gamma_t + 2\mathbf{G}_t \mathbf{D}_t \mathbf{G}_t - \mathbf{A}_t^\top \mathbf{G}_t - \mathbf{G}_t \mathbf{A}_t = \frac{d\mathbf{G}_t}{dt}. \quad (37)$$

Provided the coefficients are locally integrable on \mathbb{R}^+ , that $\Gamma_t \leq 0$ and $\mathbf{D}_t \geq 0$, if $\mathbf{G}_0 \leq 0$, then there exists a unique solution \mathbf{G}_t on \mathbb{R}^+ for (37), and $\mathbf{G}_t \leq 0$ for all $t \geq 0$ [49]. We consider this unique solution with $\mathbf{G}_0 = 0$. Then, the equation for u_t becomes affine in growth

$$\partial_t u_t = -\nabla \cdot ((\tilde{\mathbf{A}}_t \mathbf{x} + \mathbf{e}_t - 2\mathbf{D}_t \mathbf{G}_t \mathbf{x}) u_t - \mathbf{D}_t \nabla u_t) + (\tilde{b}_t - \text{tr}(\mathbf{G}_t \mathbf{D}_t) + \tilde{\mathbf{c}}_t^\top \mathbf{x} - \mathbf{e}_t^\top \mathbf{G}_t \mathbf{x}) u_t. \quad (38)$$

We redefine $\tilde{\mathbf{A}}_t = \mathbf{A}_t - 2\mathbf{D}_t \mathbf{G}_t$, $\tilde{b}_t = b_t - \text{tr}(\mathbf{G}_t \mathbf{D}_t)$ and $\tilde{\mathbf{c}}_t = \mathbf{c}_t - \mathbf{G}_t \mathbf{e}_t$. The equation now reads

$$\partial_t u_t = -\nabla \cdot ((\tilde{\mathbf{A}}_t \mathbf{x} + \mathbf{e}_t) u_t - \mathbf{D}_t \nabla u_t) + (\tilde{b}_t + \tilde{\mathbf{c}}_t^\top \mathbf{x}) u_t. \quad (39)$$

We can apply the results of Prop. A.1. Since $u_0(\mathbf{x}) = p_0(\mathbf{x})$ is a Gaussian measure, it remains Gaussian, and we denote its covariance $\tilde{\Sigma}_t$, its mean $\tilde{\mu}_t$ and its mass \tilde{m}_t . It follows that $p_t(\mathbf{x})$ is

Gaussian measure for all $t \geq 0$ since $\tilde{\Sigma}_t^{-1} - \mathbf{G}_t$ is positive definite as the sum of positive definite and positive semi-definite terms. The covariance of this Gaussian measure is then $\Sigma_t = (\tilde{\Sigma}_t^{-1} - \mathbf{G}_t)^{-1}$. We have the following fact for any differentiable matrix valued function \mathbf{B}_t invertible for all t :

$$\frac{d\mathbf{B}_t^{-1}}{dt} = -\mathbf{B}_t^{-1} \frac{d\mathbf{B}_t}{dt} \mathbf{B}_t^{-1}. \quad (40)$$

Applying this to the covariance, we find:

$$\frac{d\Sigma_t}{dt} = -\Sigma_t \frac{d\tilde{\Sigma}_t^{-1}}{dt} \Sigma_t = \Sigma_t \tilde{\Sigma}_t^{-1} \left(\tilde{\mathbf{A}}_t \tilde{\Sigma}_t + \tilde{\Sigma}_t \tilde{\mathbf{A}}_t^\top + 2\mathbf{D}_t \right) \tilde{\Sigma}_t^{-1} \Sigma_t + \Sigma_t \frac{d\mathbf{G}_t}{dt} \Sigma_t. \quad (41)$$

Substituting $\tilde{\mathbf{A}}_t = \mathbf{A}_t - 2\mathbf{D}_t \mathbf{G}_t$ and $\dot{\mathbf{G}}_t = \mathbf{\Gamma}_t + 2\mathbf{G}_t \mathbf{D}_t \mathbf{G}_t - \mathbf{A}_t^\top \mathbf{G}_t - \mathbf{G}_t \mathbf{A}_t$ we have

$$\frac{d\Sigma_t}{dt} = \mathbf{A}_t \Sigma_t + \Sigma_t \mathbf{A}_t^\top + 2\mathbf{D}_t + \Sigma_t \mathbf{\Gamma}_t \Sigma_t. \quad (42)$$

Completing the square in high-dimensions allows us to write the mean of the overall process as

$$\boldsymbol{\mu}_t = \left(\tilde{\Sigma}_t^{-1} - \mathbf{G}_t \right)^{-1} \tilde{\Sigma}_t^{-1} \tilde{\boldsymbol{\mu}}_t = \Sigma_t \tilde{\Sigma}_t^{-1} \tilde{\boldsymbol{\mu}}_t = (\mathbf{I} + \Sigma_t \mathbf{G}_t) \tilde{\boldsymbol{\mu}}_t. \quad (43)$$

Using Prop. A.1 we have

$$\frac{d\mathbf{x}_t}{dt} = \left(\frac{\Sigma_t}{dt} \mathbf{G}_t + \Sigma_t \frac{d\mathbf{G}_t}{dt} \right) \tilde{\boldsymbol{\mu}}_t + (\mathbf{I} + \Sigma_t \mathbf{G}_t) (\tilde{\mathbf{A}}_t \tilde{\boldsymbol{\mu}}_t + \tilde{\Sigma}_t \tilde{\mathbf{c}}_t + \tilde{\mathbf{e}}_t) \quad (44)$$

We replace $\tilde{\mathbf{A}}_t = \mathbf{A}_t - 2\mathbf{D}_t \mathbf{G}_t$ and we use the Riccati equation in the terms multiplying $\tilde{\boldsymbol{\mu}}_t$. In the other terms we replace $\tilde{\mathbf{c}}_t = \mathbf{c}_t - \mathbf{G}_t \mathbf{e}_t$ and $\mathbf{e}_t = \tilde{\mathbf{e}}_t$. We find:

$$\frac{d\boldsymbol{\mu}_t}{dt} = (\mathbf{A}_t + \Sigma_t \mathbf{\Gamma}_t) \boldsymbol{\mu}_t + \Sigma_t \mathbf{c}_t + \mathbf{e}_t. \quad (45)$$

We can derive the fitness following the same approach and using lengthy simplifications. We only need to gather together the terms remaining after completing the square, as well as adjust for the change in covariance in the normalisation factor. However, this result is more easily obtained by using a result for the mean over a Gaussian probability distribution of quadratic form [50, Theorem 1.5]. We find that

$$\frac{\dot{m}}{m} = \mathbb{E}[b_t + \mathbf{c}_t^\top \mathbf{X} + \frac{1}{2} \mathbf{X}^\top \mathbf{\Gamma}_t \mathbf{X}] = b_t + \mathbf{c}_t^\top \boldsymbol{\mu}_t + \frac{1}{2} \text{tr}(\mathbf{\Gamma}_t \Sigma_t) + \frac{1}{2} \boldsymbol{\mu}_t^\top \mathbf{\Gamma}_t \boldsymbol{\mu}_t \quad (46)$$

where the expectation is understood to be taken with respect to $\mathcal{N}(\mathbf{x}_t, \Sigma_t)$. \square

Corollary 2.2. *Consider an OU process with drift $\mathbf{v}_t(\mathbf{x}) = \mathbf{A}_t \mathbf{x}$ and a time-dependent fitness as in (8). Let $\mathbf{K}_t \in \mathbb{R}^{n \times n}$ be an arbitrary matrix. Then there exists $\epsilon > 0$ such that the system is indistinguishable from another OU process with drift $\mathbf{v}_t(\mathbf{x}) = \mathbf{A}_t + \mathbf{I} + \epsilon \mathbf{K}_t$ and time-dependent quadratic fitness.*

Proof. Let's define the following growth parameters: $\tilde{b}_t = b_t - \text{tr}((\mathbf{I} + \epsilon \mathbf{K}_t) \Sigma_t)/2$, $\tilde{\mathbf{c}}_t = -(\mathbf{I} + \epsilon \mathbf{K}_t)^\top \Sigma_t^{-1} \boldsymbol{\mu}_t$ and $\tilde{\mathbf{\Gamma}}_t = \mathbf{\Gamma}_t - ((\mathbf{I} + \epsilon \mathbf{K}_t)^\top \Sigma_t^{-1} + \Sigma_t^{-1} (\mathbf{I} + \epsilon \mathbf{K}_t))$. With these and the drift $\mathbf{v}_t(\mathbf{x})$, the solution to the system of ODE above is unchanged. We take ϵ as the largest value such that $\tilde{\mathbf{\Gamma}}_t$ is negative definite. This value is strictly larger than zero thanks to the identity. This derivation still holds if the drift is autonomous: if $\mathbf{v}_t(\mathbf{x}) = \mathbf{A} \mathbf{x}$, then there for any \mathbf{K} , there exists $\epsilon > 0$ such that the system is indistinguishable from another OU process with $\tilde{\mathbf{v}}_t(\mathbf{x}) = \mathbf{A} + \mathbf{I} + \epsilon \mathbf{K}$ and time-dependent quadratic fitness. \square

B Continuous-time loss for OU processes with quadratic fitness

Theorem 2.3 (Loss function for OU processes with quadratic fitness). *Consider the true process as well as the inferred processed to both be OU processes with quadratic fitness, i.e. (8). Denoting*

$q = 2/\gamma$, with $K\Delta t$ fixed, when $\Delta t \rightarrow 0$ we have $\Delta t^{-1}L \rightarrow \mathcal{L}$, where \mathcal{L} is the continuous time loss function:

$$\mathcal{L} = q^{-1} \int_0^T dt m_t \left(\|\mathbf{v}_t - \hat{\mathbf{v}}_t\|_{\mathbf{X}_t^{-1}}^2 + \frac{1}{2}(h_t - \hat{h}_t)^2 \right) \quad (13)$$

$$+ q \sum_{i,j} \frac{\sigma_{i,t}^2 \sigma_{j,q,t}^2}{(\sigma_{j,q,t}^2 \sigma_{i,t}^2 + \sigma_{i,q,t}^2 \sigma_{j,t}^2)^2} \left(\mathbf{w}_{i,t}^\top (\mathbf{B}_t - \hat{\mathbf{B}}_t) \mathbf{w}_{j,t} \right)^2 \Big) + \lambda \int_0^T R_t dt \quad (14)$$

where $\Sigma_t = \sum_i \sigma_{i,t}^2 \mathbf{w}_{i,t} \mathbf{w}_{i,t}^\top$ is the eigendecomposition of the covariance Σ_t , $\sigma_{i,q,t}^2 = 1 + q\sigma_{i,t}^2$ for all i , $\mathbf{X}_t = 2\Sigma_t + q^{-1}$. We also have

$$\begin{aligned} \mathbf{B}_t - \hat{\mathbf{B}}_t &= \Sigma_t (\mathbf{A}_t - \hat{\mathbf{A}}_t)^\top + (\mathbf{A}_t - \hat{\mathbf{A}}_t) \Sigma_t + \Sigma_t (\Gamma_t - \hat{\Gamma}_t) \Sigma_t + 2(\mathbf{D}_t - \hat{\mathbf{D}}_t), \\ \mathbf{v}_t - \hat{\mathbf{v}}_t &= ((\mathbf{A}_t - \hat{\mathbf{A}}_t) + \Sigma_t (\Gamma_t - \hat{\Gamma}_t)) \boldsymbol{\mu}_t + \Sigma_t (\mathbf{c}_t - \hat{\mathbf{c}}_t) + (\mathbf{e}_t - \hat{\mathbf{e}}_t), \\ h_t - \hat{h}_t &= \frac{1}{2} \boldsymbol{\mu}_t^\top (\Gamma_t - \hat{\Gamma}_t) \boldsymbol{\mu}_t + (\mathbf{c}_t - \hat{\mathbf{c}}_t)^\top \boldsymbol{\mu}_t + (b_t - \hat{b}_t) + \frac{1}{2} \text{tr}((\Gamma_t - \hat{\Gamma}_t) \Sigma_t), \end{aligned} \quad (15)$$

and R_t is a strongly convex function of the parameters defining the fitness and the drift. For $\lambda > 0$, \mathcal{L} has a unique minimiser $t \mapsto (\hat{\mathbf{A}}_t^*, \hat{\mathbf{e}}_t^*, \hat{b}_t^*, \hat{\mathbf{c}}_t^*, \hat{\Gamma}_t^*)$.

Proof. When the entropic regularisation $\epsilon = 0$, the unbalanced Sinkhorn divergence between two Gaussian measures reduces to the Gaussian-Hellinger-Kantorovich distance $S_{0,\gamma} = \text{GHK}_\gamma$. Between the inferred and the true process at time t_i it reads:

$$\begin{aligned} \text{GHK} \left(m_{t_i} \mathcal{N}(\boldsymbol{\mu}_{t_i}, \Sigma_{t_i}), \hat{m}_{t_i} \mathcal{N}(\hat{\boldsymbol{\mu}}_{t_i}, \hat{\Sigma}_{t_i}) \right) \\ = 2q^{-1} \left(m_{t_i} + \hat{m}_{t_i} - 2 \sqrt{\frac{m_{t_i} \hat{m}_{t_i}}{\det \mathbf{J}} \exp \left[-\frac{1}{2} (\boldsymbol{\mu}_{t_i} - \hat{\boldsymbol{\mu}}_{t_i})^\top \mathbf{X}_{t_i}^{-1} (\boldsymbol{\mu}_{t_i} - \hat{\boldsymbol{\mu}}_{t_i}) \right]} \right), \end{aligned} \quad (47)$$

where we have

$$\mathbf{X}_{t_i} = \Sigma_{t_i} + \Sigma_{t_i} + q^{-1} \mathbf{I}, \quad \mathbf{J} = (\Sigma_{t_i,q} \Sigma_{t_i,q})^{1/2} (\mathbf{I} - q(\Sigma_{t_i} \Sigma_{t_i,q}^{-1} \Sigma_{t_i,q}^{-1} \Sigma_{t_i})^{1/2}), \quad (48)$$

with

$$\Sigma_{t_i,q} = q\Sigma_{t_i} + \mathbf{I}, \quad \Sigma_{t_i,q} = q\Sigma_{t_i} + \mathbf{I}. \quad (49)$$

Without loss of generality we consider the case $t_1 = \Delta t$, and we perform Taylor expansion in Δt of the GHK loss. Because the final expansion will be of order Δt^2 , only the terms of order Δt of the covariances play a role.

The hard part in this expansion is the expansion of $\det \mathbf{J}$. We denote $\Sigma_{\Delta t} = \Sigma_0 + \Delta t \mathbf{A}$ and $\tilde{\Sigma}_{\Delta t} = \Sigma_0 + \Delta t \tilde{\mathbf{A}}$ and $\Sigma_{0,q} = \mathbf{I} + q\Sigma_0$. For this, we need to compute $\Sigma_{\Delta t} \Sigma_{\Delta t,q}^{-1} \tilde{\Sigma}_{\Delta t} \tilde{\Sigma}_{\Delta t,q}^{-1}$, up to second order in Δt . We have that

$$\Sigma_{\Delta t} \Sigma_{\Delta t,q}^{-1} = (\Sigma_0 + \Delta t \mathbf{A}) (\Sigma_{0,q}^{-1} - q\Delta t \Sigma_{0,q}^{-1} \mathbf{A} \Sigma_{0,q}^{-1} + q^2 \Delta t^2 \Sigma_{0,q}^{-1} \mathbf{A} \Sigma_{0,q}^{-1} \mathbf{A} \Sigma_{0,q}^{-1}) \quad (50)$$

$$= \Sigma_0 \Sigma_{0,q}^{-1} + \Delta t (\mathbf{A} \Sigma_{0,q}^{-1} - q\Sigma_0 \Sigma_{0,q}^{-1} \mathbf{A} \Sigma_{0,q}^{-1}) \quad (51)$$

$$+ \Delta t^2 q (q\Sigma_0 \Sigma_{0,q}^{-1} \mathbf{A} \Sigma_{0,q}^{-1} \mathbf{A} \Sigma_{0,q}^{-1} - \mathbf{A} \Sigma_{0,q}^{-1} \mathbf{A} \Sigma_{0,q}^{-1}). \quad (52)$$

We also have

$$\tilde{\Sigma}_{\Delta t,q}^{-1} \tilde{\Sigma}_{\Delta t} = (\Sigma_{0,q}^{-1} - q\Delta t \Sigma_{0,q}^{-1} \tilde{\mathbf{A}} \Sigma_{0,q}^{-1} + q^2 \Delta t^2 \Sigma_{0,q}^{-1} \tilde{\mathbf{A}} \Sigma_{0,q}^{-1} \tilde{\mathbf{A}} \Sigma_{0,q}^{-1}) (\Sigma_0 + \Delta t \tilde{\mathbf{A}}) \quad (53)$$

$$= \Sigma_{0,q}^{-1} \Sigma_0 + \Delta t (\Sigma_{0,q}^{-1} \tilde{\mathbf{A}} - q\Sigma_{0,q}^{-1} \tilde{\mathbf{A}} \Sigma_{0,q}^{-1} \Sigma_0) \quad (54)$$

$$+ \Delta t^2 q (q\Sigma_{0,q}^{-1} \tilde{\mathbf{A}} \Sigma_{0,q}^{-1} \tilde{\mathbf{A}} \Sigma_{0,q}^{-1} \Sigma_0 - \Sigma_{0,q}^{-1} \tilde{\mathbf{A}} \Sigma_{0,q}^{-1} \tilde{\mathbf{A}}). \quad (55)$$

We can now gather the $O(1)$, $O(\Delta t)$, $O(\Delta t^2)$ separately. We denote them respectively $\mathbf{M}, \mathbf{H}_1, \mathbf{H}_{2345}$. We define the operation 'tt.' as the 'tilde transpose' operation, which is applied

to the term directly to its left. We therefore have

$$M = \Sigma_0 \Sigma_{0,q}^{-2} \Sigma_0 \quad (56)$$

$$H_1 = \Sigma_0 (\Sigma_{0,q}^{-2} \tilde{A} - q \Sigma_{0,q}^{-2} \tilde{A} \Sigma_{0,q}^{-1} \Sigma_0) + \text{tt}. \quad (57)$$

$$H_{2345} = q \Sigma_0 (q \Sigma_{0,q}^{-2} \tilde{A} \Sigma_{0,q}^{-1} \tilde{A} \Sigma_{0,q}^{-1} \Sigma_0 - \Sigma_{0,q}^{-2} \tilde{A} \Sigma_{0,q}^{-1} \tilde{A}) + \text{tt}. \quad (58)$$

$$+ A \Sigma_{0,q}^{-2} \tilde{A} + q^2 \Sigma_0 \Sigma_{0,q}^{-1} A \Sigma_{0,q}^{-2} \tilde{A} \Sigma_{0,q}^{-1} \Sigma_0 \quad (59)$$

$$- q A \Sigma_{0,q}^{-2} \tilde{A} \Sigma_{0,q}^{-1} \Sigma_0 + \text{tt}. \quad (60)$$

Using the Woodbury formula we also have $\Sigma_0 \Sigma_{0,q}^{-1} = q^{-1} (I - \Sigma_{0,q}^{-1})$. We have that Σ_0 and $\Sigma_{0,q}^{-1}$ commute, and that $M^{1/2} = \Sigma_0 \Sigma_{0,q}^{-1} = q^{-1} (I - \Sigma_{0,q}^{-1})$. We introduce $S(Y)$ the unique solution X to the following Lyapunov equation

$$M^{1/2} X + X M^{1/2} = Y \quad (61)$$

This solution is expressed in terms of an integral (and is linear in Y), ie.

$$S(Y) = \int_0^\infty e^{-M^{1/2}t} Y e^{-M^{1/2}t} dt \quad (62)$$

As a result, we have

$$\left(\Sigma_{\Delta t} \Sigma_{\Delta t,q}^{-1} \tilde{\Sigma}_{\Delta t,q}^{-1} \tilde{\Sigma}_{\Delta t} \right)^{1/2} = M^{1/2} + \Delta t S(H_1) + \Delta t^2 (S(H_{2345}) - S(S(H_1)^2)) \quad (63)$$

We have $I - q M^{1/2} = \Sigma_{0,q}^{-1}$, such that

$$\det \left(I - q \left(\Sigma_{\Delta t} \Sigma_{\Delta t,q}^{-1} \tilde{\Sigma}_{\Delta t,q}^{-1} \tilde{\Sigma}_{\Delta t} \right)^{1/2} \right) = \det \Sigma_{0,q}^{-1} \det (I + \Delta t L + \Delta t^2 G), \quad (64)$$

where $L = -q \Sigma_{0,q} S(H_1)$ and $G = -q \Sigma_{0,q} (S(H_{2346}) - S(S(H_1)^2))$. We then have the following expansion at second order in Δt

$$\det (I + \Delta t L + \Delta t^2 G) = 1 + \Delta t \text{tr} L + \frac{\Delta t^2}{2} (\text{tr}^2 L - \text{tr} L^2 + 2 \text{tr} G). \quad (65)$$

So we need to compute $\text{tr} L$, $\text{tr} G$, $\text{tr} L^2$. We have, using the fact that $\Sigma_{0,q}$ commutes with $M^{1/2}$,

$$\text{tr} L = -q \text{tr} (\Sigma_{0,q} \int_0^\infty e^{-M^{1/2}t} H_1 e^{-M^{1/2}t} dt) = -\frac{q}{2} \text{tr} (M^{-1/2} \Sigma_{0,q} H_1) \quad (66)$$

$$= -\frac{q}{2} \text{tr} (\Sigma_{0,q}^2 \Sigma_0^{-1} H_1) = -\frac{q}{2} (\text{tr} (A + \tilde{A}) - q \text{tr} ((A + \tilde{A}) \Sigma_{0,q}^{-1} \Sigma_0)) \quad (67)$$

$$= -\frac{q}{2} \text{tr} (\Sigma_{0,q}^{-1} (A + \tilde{A})). \quad (68)$$

Computation of $\text{tr} G$

Using the same trick as for $\text{tr} L$ and standard properties of the trace we have

$$\text{tr} (-q \Sigma_{0,q} S(H_{2345})) = \frac{1}{2} \left(q^2 \text{tr} (A \Sigma_{0,q}^{-1} A \Sigma_{0,q}^{-1} + \text{tt}.) - q \text{tr} (\Sigma_0^{-1} A \Sigma_{0,q}^{-2} \tilde{A}) \right) \quad (69)$$

To compute $\text{tr} (\Sigma_{0,q} S(S(H_1)^2))$ we denote $M^{1/2} = \sum_i u_i w_i w_i^\top$ the eigendecomposition of $M^{1/2}$. We also denote $W_i = w_i w_i^\top$. Therefore we have

$$S(H_1) = \sum_{i,j} \frac{1}{u_i + u_j} W_i H_1 W_j \quad (70)$$

such that

$$\Sigma_{0,q} S(S(H_1)^2) = \sum_{i,j,k} \frac{(1 - u_k q)^2 (1 - u_j q)}{(u_i + u_k)(u_j + u_k)(u_i + u_j)} W_i (u_i \tilde{A} + u_k A) W_k (u_k \tilde{A} + u_j A) W_j. \quad (71)$$

Taking the trace we have

$$\text{tr}(\Sigma_{0,q} S(S(H_1)^2)) = \sum_{i,j} \frac{(1-u_j q)^2 (1-u_i q)}{2u_i(u_i + u_j)^2} \text{tr} \left(W_i(u_i \tilde{A} + u_j A) W_j(u_j \tilde{A} + u_i A) \right). \quad (72)$$

Additionally, we have that

$$\text{tr} \left(W_i(u_i \tilde{A} + u_j A) W_j(u_j \tilde{A} + u_i A) \right) \quad (73)$$

$$= u_i u_j \text{tr} \left(W_i(A - \tilde{A}) W_j(A - \tilde{A}) \right) + (u_i + u_j)^2 \text{tr} \left(W_i A W_j \tilde{A} \right) \quad (74)$$

As a result we have

$$\text{tr} G = \frac{1}{2} q^2 \text{tr}(A \Sigma_{0,q}^{-1} A \Sigma_{0,q}^{-1} + \text{tt.}) + \frac{q}{2} \sum_{i,j} \frac{(1-u_j q)^2 (1-u_i q) u_j}{(u_i + u_j)^2} \text{tr} \left(W_i(A - \tilde{A}) W_j(A - \tilde{A}) \right) \quad (75)$$

Computation of $\text{tr} L^2$

Similarly we have

$$\Sigma_{0,q} S(H_1) = \sum_{i,j} \frac{(1-u_j q)}{(u_i + u_j)} W_i \left(u_i \tilde{A} + u_j A \right) W_j. \quad (76)$$

Using the same approach for $\text{tr} G$ we find

$$\text{tr}((\Sigma_{0,q} S(H_1))^2) = \sum_{i,j} \frac{(1-u_i q)(1-u_j q) u_i u_j}{(u_i + u_j)^2} \text{tr} \left(W_i(A - \tilde{A}) W_j(A - \tilde{A}) \right) \quad (77)$$

$$+ \text{tr}(\Sigma_{0,q}^{-1} A \Sigma_{0,q}^{-1} \tilde{A}). \quad (78)$$

As a result we have

$$\text{tr} L^2 = q^2 \text{tr}(\Sigma_{0,q}^{-1} A \Sigma_{0,q}^{-1} \tilde{A}) + q^2 \sum_{i,j} \frac{(1-u_i q)(1-u_j q) u_i u_j}{(u_i + u_j)^2} \text{tr} \left(W_i(A - \tilde{A}) W_j(A - \tilde{A}) \right) \quad (79)$$

Computation of $\det(\Sigma_{\Delta t,q} \tilde{\Sigma}_{\Delta t,q})^{1/2}$

We have

$$\Sigma_{\Delta t,q} \tilde{\Sigma}_{\Delta t,q} = \Sigma_{0,q}^2 \left(I + q \Delta t (\Sigma_{0,q}^{-2} A \Sigma_{0,q} + \Sigma_{0,q}^{-1} \tilde{A}) + q^2 \Delta t^2 \Sigma_{0,q}^{-2} A \tilde{A} \right). \quad (80)$$

Taking the determinant and Taylor expanding we have

$$\det(\Sigma_{\Delta t,q} \tilde{\Sigma}_{\Delta t,q}) = \det(\Sigma_{0,q})^2 \quad (81)$$

$$\times \left(1 + q \Delta t \text{tr}(\Sigma_{0,q}^{-1} (A + \tilde{A})) + q^2 \frac{\Delta t^2}{2} \left((\text{tr}^2(\Sigma_{0,q}^{-1} (A + \tilde{A})) - \text{tr}(\Sigma_{0,q}^{-1} A \Sigma_{0,q}^{-1} A + \text{tt.})) \right) \right). \quad (82)$$

Expanding the square root we find

$$\det(\Sigma_{\Delta t,q} \tilde{\Sigma}_{\Delta t,q})^{1/2} = \det(\Sigma_{0,q}) \quad (83)$$

$$\times \left(1 + q \frac{\Delta t}{2} \text{tr}(\Sigma_{0,q}^{-1} (A + \tilde{A})) + q^2 \frac{\Delta t^2}{8} \left(\text{tr}^2(\Sigma_{0,q}^{-1} (A + \tilde{A})) - 2 \text{tr}(\Sigma_{0,q}^{-1} A \Sigma_{0,q}^{-1} A + \text{tt.}) \right) \right) \quad (84)$$

Computation of $\det \mathbf{J}$

Going back to $\det \mathbf{J}$, we are left with

$$\det \mathbf{J} = \left(1 + q \frac{\Delta t}{2} \text{tr}(\Sigma_{0,q}^{-1}(\mathbf{A} + \tilde{\mathbf{A}})) + q^2 \frac{\Delta t^2}{8} \left(\text{tr}^2(\Sigma_{0,q}^{-1}(\mathbf{A} + \tilde{\mathbf{A}})) - 2\text{tr}(\Sigma_{0,q}^{-1}\mathbf{A}\Sigma_{0,q}^{-1}\mathbf{A} + \text{tt.}) \right) \right) \quad (85)$$

$$\times \left(1 + \Delta t \text{tr} \mathbf{L} + \frac{\Delta t^2}{2} (\text{tr}^2 \mathbf{L} - \text{tr} \mathbf{L}^2 + 2\text{tr} \mathbf{G}) \right). \quad (86)$$

Using the result for $\text{tr} \mathbf{L}$ we see that the terms in Δt cancel, and the final expansion is of order Δt^2 . As a result, after simplifications we are left with

$$\det \mathbf{J} = \left(1 - q^2 \frac{\Delta t^2}{4} \text{tr}(\Sigma_{0,q}^{-1}\mathbf{A}\Sigma_{0,q}^{-1}\mathbf{A} + \text{tt.}) + \frac{\Delta t^2}{2} (-\text{tr} \mathbf{L}^2 + 2\text{tr} \mathbf{G}) \right). \quad (87)$$

Simplifying further we find that

$$\det \mathbf{J} = 1 + \frac{\Delta t^2 q}{2} \left(\frac{q}{2} \text{tr}(\Sigma_{0,q}^{-1}(\mathbf{A} - \tilde{\mathbf{A}})\Sigma_{0,q}^{-1}(\mathbf{A} - \tilde{\mathbf{A}})) \right) \quad (88)$$

$$+ \sum_{i,j} \frac{u_j(1-u_iq)(1-u_jq)(1-q(u_j+u_i))}{(u_i+u_j)^2} \text{tr} \left(\mathbf{W}_i(\mathbf{A} - \tilde{\mathbf{A}})\mathbf{W}_j(\mathbf{A} - \tilde{\mathbf{A}}) \right) \quad (89)$$

This can be simplified

$$\left(\frac{q}{2} \text{tr}(\Sigma_{0,q}^{-1}(\mathbf{A} - \tilde{\mathbf{A}})\Sigma_{0,q}^{-1}(\mathbf{A} - \tilde{\mathbf{A}})) \right) \quad (90)$$

$$+ \sum_{i,j} \frac{u_j(1-u_iq)(1-u_jq)(1-q(u_j+u_i))}{(u_i+u_j)^2} \text{tr} \left(\mathbf{W}_i(\mathbf{A} - \tilde{\mathbf{A}})\mathbf{W}_j(\mathbf{A} - \tilde{\mathbf{A}}) \right) \quad (91)$$

$$= \sum_{i,j} \frac{(1-qu_i)(1-qu_j)}{(u_i+u_j)^2} \left(\frac{q}{2} u_i^2 - \frac{q}{2} u_j^2 + u_j \right) \text{tr} \left(\mathbf{W}_i(\mathbf{A} - \tilde{\mathbf{A}})\mathbf{W}_j(\mathbf{A} - \tilde{\mathbf{A}}) \right). \quad (92)$$

which leaves us with the following simplification

$$\det \mathbf{J} = 1 + \frac{\Delta t^2 q}{2} \sum_{i,j} \frac{u_i(1-u_iq)(1-u_jq)}{(u_i+u_j)^2} \text{tr} \left(\mathbf{W}_i(\mathbf{A} - \tilde{\mathbf{A}})\mathbf{W}_j(\mathbf{A} - \tilde{\mathbf{A}}) \right). \quad (93)$$

Using $\Sigma_0 = \sum_i \sigma_i^2 \mathbf{W}_i$, we can compute the eigenvalue decomposition as a function of σ_i . Using the notation $\sigma_{i,q}^2 = 1 + q\sigma_i^2$ we have

$$\frac{u_i(1-u_iq)(1-u_jq)}{(u_i+u_j)^2} = \frac{\sigma_i^2 \sigma_{j,q}^2}{(\sigma_{j,q}^2 \sigma_i^2 + \sigma_{i,q}^2 \sigma_j^2)^2}. \quad (94)$$

Finally, at first non-zero order in Δt the determinant reads

$$\det \mathbf{J} = 1 + \frac{\Delta t^2 q}{2} \sum_{i,j} \frac{\sigma_i^2 \sigma_{j,q}^2}{(\sigma_{j,q}^2 \sigma_i^2 + \sigma_{i,q}^2 \sigma_j^2)^2} \text{tr} \left(\mathbf{W}_i(\mathbf{A} - \tilde{\mathbf{A}})\mathbf{W}_j(\mathbf{A} - \tilde{\mathbf{A}}) \right). \quad (95)$$

Taylor expansion of the GHK term

We now expand the masses and mean at first order in Δt , $m_{\Delta t} = m_0 + \Delta t m_0 h$, $\tilde{m}_{\Delta t} = m_0 + \Delta t m_0 \tilde{h}$, $\boldsymbol{\mu}_{\Delta t} = \boldsymbol{\mu}_0 + \Delta t \mathbf{v}$, $\tilde{\boldsymbol{\mu}}_{\Delta t} = \boldsymbol{\mu}_0 + \Delta t \tilde{\mathbf{v}}$. We have at first non zero order in Δt

$$\exp \left[-\frac{1}{2} (\boldsymbol{\mu}_{\Delta t} - \tilde{\boldsymbol{\mu}}_{\Delta t})^\top \mathbf{X}_{\Delta t}^{-1} (\boldsymbol{\mu}_{\Delta t} - \tilde{\boldsymbol{\mu}}_{\Delta t}) \right] = 1 - \frac{\Delta t^2}{2} (\mathbf{v} - \tilde{\mathbf{v}})^\top \mathbf{X}_0^{-1} (\mathbf{v} - \tilde{\mathbf{v}}). \quad (96)$$

We denote $\mathbf{X}_0 = 2\Sigma_0 + q^{-1}$. Expanding the remaining terms, the zeroth and first order terms cancel, leading to the expansion

$$\text{GHK} \left(m_{\Delta t} \mathcal{N}(\boldsymbol{\mu}_{\Delta t}, \boldsymbol{\Sigma}_{\Delta t}), \hat{m}_{\Delta t} \mathcal{N}(\hat{\boldsymbol{\mu}}_{\Delta t}, \hat{\boldsymbol{\Sigma}}_{\Delta t}) \right) \quad (97)$$

$$= m_0 q^{-1} \Delta t^2 \left(\left((\mathbf{v} - \tilde{\mathbf{v}})^\top \mathbf{X}_0^{-1} (\mathbf{v} - \tilde{\mathbf{v}}) + \frac{1}{2} (h - \tilde{h})^2 \right) \right) \quad (98)$$

$$+ q \sum_{i,j} \frac{\sigma_i^2 \sigma_{j,q}^2}{(\sigma_{j,q}^2 \sigma_i^2 + \sigma_{i,q}^2 \sigma_j^2)^2} \text{tr} \left(\mathbf{W}_i (\mathbf{A} - \tilde{\mathbf{A}}) \mathbf{W}_j (\mathbf{A} - \tilde{\mathbf{A}}) \right). \quad (99)$$

Integrating over all snapshots, the continuous time loss reads

$$\begin{aligned} \mathcal{L} = & \int_0^T m_t q^{-1} \left(\left((\mathbf{v} - \tilde{\mathbf{v}})^\top \mathbf{X}_0^{-1} (\mathbf{v} - \tilde{\mathbf{v}}) + \frac{1}{2} (h - \tilde{h})^2 \right) \right. \\ & + q \sum_{i,j} \frac{\sigma_i^2 \sigma_{j,q}^2}{(\sigma_{j,q}^2 \sigma_i^2 + \sigma_{i,q}^2 \sigma_j^2)^2} \text{tr} \left(\mathbf{W}_i (\mathbf{A} - \tilde{\mathbf{A}}) \mathbf{W}_j (\mathbf{A} - \tilde{\mathbf{A}}) \right) \left. \right) dt \\ & + \lambda \int_0^T \int \rho_t(\mathbf{x}) (\|\mathbf{v}_t(\mathbf{x})\|_2^2 + \alpha |g_t(\mathbf{x})|^2) d\mathbf{x} dt. \end{aligned} \quad (100)$$

The first order expansions \mathbf{A} , h , and \mathbf{v} are obtained directly using the ODEs in Prop. 2.1, giving the final result.

Let's denote $\theta_t \in \mathbb{R}^N$ the vector of the N parameters defining the growth and the drift at time t . We study the functional $\mathcal{L}[\theta_t]$ which reads

$$\mathcal{L}[\theta_t] = \int_0^T (F_t(\theta_t) + \lambda R_t(\theta_t)) dt. \quad (101)$$

where F is the part of the integrand coming from the expansion of the Gaussian-Hellinger-Kantorovich, and R is the regularisation.

Let's take $t \in [0, T]$. Because $\theta \mapsto F_t(\theta)$ is the composition of convex functions and of affine maps, $\theta \mapsto F_t(\theta)$ is also convex. Let's show that $\theta \mapsto R_t(\theta)$ is a strongly convex function.

Lemma B.1. *Let $w : (\theta, x) \in \mathbb{R}^N \times \mathbb{R}^n \mapsto \mathbb{R}^d$ be a function continuous in x and linear in θ . Let $f : \mathbb{R}^d \mapsto \mathbb{R}$ be a continuous, strictly convex function and ρ a continuous function from \mathbb{R}^n to $]0, +\infty[$. We then have that*

$$\ell : \theta \in \mathbb{R}^N \mapsto \int \rho(x) f(w(\theta, x)) dx \quad (102)$$

is strictly convex.

Proof. Let's take $\theta_1 \neq \theta_2$ and $\alpha \in]0, 1[$. By linearity, we have $\forall x$:

$$w(\alpha\theta_1 + (1-\alpha)\theta_2, x) = \alpha w(\theta_1, x) + (1-\alpha)w(\theta_2, x). \quad (103)$$

Because $\theta \mapsto f(w(\theta, x))$ is the composition of a convex function and of a linear map, it is convex for all x . Then, $\forall x$,

$$f(w(\alpha\theta_1 + (1-\alpha)\theta_2, x)) = f(\alpha w(\theta_1, x) + (1-\alpha)w(\theta_2, x)) \leq \alpha f(w(\theta_1, x)) + (1-\alpha)f(w(\theta_2, x)). \quad (104)$$

Because w is linear in θ , each of its component in \mathbb{R}^d is a multivariate polynomial function of θ of degree one. By uniqueness of the coefficient of polynomial functions, because $\theta_1 \neq \theta_2$ there exists x_0 such that $w(\theta_1, x_0) \neq w(\theta_2, x_0)$. By continuity, this is also true on an open set $U \subset \mathbb{R}^n$ centred in x_0 . Therefore, since f is strictly convex, the inequality is strict for all $x \in U$, i.e.:

$$f(w(\alpha\theta_1 + (1-\alpha)\theta_2, x)) = f(\alpha w(\theta_1, x) + (1-\alpha)w(\theta_2, x)) < \alpha f(w(\theta_1, x)) + (1-\alpha)f(w(\theta_2, x)). \quad (105)$$

By multiplying by $\rho(x)$, which is strictly positive, and by integrating, we keep the inequality strict and we find

$$\ell(\alpha\theta_1 + (1-\alpha)\theta_2) < \alpha\ell(\theta_1) + (1-\alpha)\ell(\theta_2), \quad (106)$$

proving that ℓ is strictly convex. \square

	Hidden layers	Batch size	Learning rate	Iterations
Bistable ($d \leq 10$)	64, 64, 64	256	0.003	25,000
Bistable ($d > 10$)	256, 256, 256	256	0.001	100,000
Bifurcating CLE	64, 64, 64	256	0.01	10,000
Haematopoietic CLE	128, 128, 128	256	0.01	10,000
Lineage tracing	128, 128, 128	256	0.01	10,000

Table 5: Hyperparameter settings: score networks

		Hidden (force)	Hidden (growth)	Batch	LR	Iterations	λ	α	γ
UPFI	Bistable ($d \leq 10$)	64, 64, 64	64, 64, 64	256	0.003	5,000	see details	–	5
	Bistable ($d > 10$)	256, 256, 256	256, 256, 256	256	0.001	25,000	see details	–	5
	Bif CLE	64, 64, 64	64	256	0.003	10,000	0.001	1	5
	Haem CLE	128, 128, 128	128	256	0.003	10,000	0.001	1	5
	Lineage	128, 128, 128	128, 128, 128	256	0.001	10,000	0.001	1	25
PFI	Bistable ($d \leq 10$)	64, 64, 64	–	256	0.003	5,000	see details	–	–
	Bistable ($d > 10$)	256, 256, 256	–	256	0.001	25,000	see details	–	–
	Bif CLE	64, 64, 64	–	256	0.003	10,000	0.001	–	–
	Haem CLE	128, 128, 128	–	256	0.003	10,000	0.001	–	–
	Lineage	128, 128, 128	–	256	0.001	10,000	0.001	–	–
ODE	Bistable ($d \leq 10$)	Coupled to growth	64, 64, 64	256	0.003	5,000	see details	–	5
	Bistable ($d > 10$)	Coupled to growth	256, 256, 256	256	0.001	25,000	see details	–	5
	Lineage	Coupled to growth	128, 128, 128	256	0.001	10,000	0.001	0.1	25
TIGON++	Bistable ($d \leq 10$)	64, 64, 64	64, 64, 64	256	0.003	5,000	see details	–	5
	Bistable ($d > 10$)	256, 256, 256	256, 256, 256	256	0.001	25,000	see details	–	5
	Lineage tracing	128, 128, 128	128, 128, 128	256	0.001	10,000	0.001	0.1	25

Table 6: Hyperparameter settings: dynamics. For bistable system certain hyperparameters chosen as a function of d , see Section C.2 for details.

We have $\rho_t(x) > 0$ for all x because it is a Gaussian density. The drift and the fitness being linear in θ , this shows that

$$\theta \mapsto R_t(\theta) = \int \rho_t(x) (\|v_t(x)\|_2^2 + \alpha |g_t(x)|^2) dx \quad (107)$$

is a strictly convex function of θ . Additionally, it is a multivariate polynomial function of θ of degree two, so its Hessian is a definite positive and constant, proving that $\theta \mapsto R_t(\theta)$ is strongly convex, and so is $F_t + R_t$. Along with the fact that both $(t, \theta) \mapsto F_t(\theta)$ and $(t, \theta) \mapsto R_t(\theta)$ are continuous in $[0, T] \times \mathbb{R}^N$, this is enough to ensure the existence and uniqueness of a minimum $t \mapsto \theta_t^*$ for the functional $\mathcal{L}[\theta_t]$ [51]. \square

C Implementation and experiment details

We implement Alg. 1 using PyTorch and employ the GeomLoss package [52] for computation of the unbalanced Sinkhorn divergence. All model training was carried out using a NVIDIA L40S GPU. Code is available at <https://github.com/zsteve/UPFI>.

C.1 Score matching

While in principle any score matching approach to learn $s_t(x) = \nabla \log p_t(x)$ can be used within Alg. 1, in practice we employ denoising score matching [30] within the noise-conditional score network framework introduced in [5]. For $K + 1$ snapshots taken at times $(t_i)_{i=0}^K$, we parameterise the time-dependent score using a multilayer perceptron (MLP) $s_\phi(t, x, \eta) = \text{NN}_\phi(d + 2, d)$ where d is the dimension and η is the noise level, and train using the algorithm described in [5, Section 4.2]. While a range of noise levels $\eta_0 < \dots < \eta_L$ are used for training the score, subsequently for training the probability flow we use the smallest noise scale $\eta = \eta_0$, representing our final estimate of the score.

In what follows, we use a default of $L = 5$ noise levels logarithmically spaced between $(\exp(-2), 1)$. Score networks $s_t(x)$ are parameterised using MLPs with ReLU activations. In all case, score training was carried out using noise-conditional denoising score matching [5, 30] using the AdamW optimiser with the hyperparameter choices listed in Table 5.

C.2 Training: UPFI and PFI

Additive noise models For UPFI, in all cases we parameterise an *autonomous* force $v_\theta(\mathbf{x})$ using a MLP. This is because, in all simulated systems, the true force is also autonomous. In the experimental lineage tracing dataset we reason that an autonomous force would be consistent with the biologically motivated model of a Waddington’s landscape [34]. We parameterise separately the force $v_\theta(\mathbf{x})$ and growth $g_\theta(\mathbf{x})$ using MLPs with ReLU activations. We train UPFI models using the AdamW optimiser, our architecture and hyperparameter choices are listed in Table 6.

For PFI, motivated by the observations in the Gaussian case we reason that an autonomous force is insufficient to fit the data if growth is not accounted for. We therefore employ a *non-autonomous* force, parameterising $v_\theta(t, \mathbf{x}) = \text{NN}_\theta(d + 1, d)(t, \mathbf{x})$ with ReLU activations. We train PFI models using the AdamW optimiser, our architecture and hyperparameter choices are listed in Table 6.

Multiplicative noise model For the multiplicative noise models we considered in Fig. 5, we parameterise an autonomous force in two components, corresponding to production and degradation terms in the model (16). Specifically, we let

$$\mathbf{f}_\theta(\mathbf{x}) = \text{NN}_\theta(d, d)(\mathbf{x}), \quad \mathbf{g}_\theta(\mathbf{x}) = \text{NN}_\theta(d, d)(\mathbf{x}),$$

and the resulting force is $(\mathbf{f}_\theta - \mathbf{g}_\theta)(\mathbf{x})$. To constrain the output of both networks to be non-negative, we opt for a Softplus activation on the final layer of outputs. For all other layers we use the ReLU activation as a default choice. Architectures for (\mathbf{f}, \mathbf{g}) and all other hyperparameter choices are as given in Table 6.

Scaling regularisation with dimension for bifurcating system For the bifurcating system of Figure 3 and Table 1, it is necessary to scale the regularisation parameters (λ, α) as d ranges in $\{2, 5, 10, 25, 50\}$. We reason that $\|\mathbf{v}\|^2 = \sum_{i=1}^d v_i^2$ grows roughly linearly with increasing dimension d , while the growth rate $|g|$ does not depend on d since it is a scalar-valued field. This motivates the rescaled regularisation term

$$\lambda(d^{-1}\|\mathbf{v}\|^2 + \alpha|g|^2)$$

In practice we use $\lambda = 0.001$ and $\alpha = 0.1$ for all the values of d considered.

Additionally, we alter aspects of training such as network size, learning rate, and number of training iterations depending on d . For $d > 10$ we used a wider network, paired with a smaller learning rate and more training iterations. These choices were consistently applied to all methods under consideration except for DeepRUOT due to difficulties with modifying the implementation.

C.3 Training: fitness-ODE

Motivated by issues pertaining to the identifiability of dynamics involving both drift and growth, we propose a well-known dynamical model as a baseline model for inference. Let $\{\rho_t\}_t$ be a continuous distributional path satisfying mild conditions in the space of measures describing some population evolution. Then there exists a unique scalar field $U_t(\mathbf{x}_t)$ such that ρ_t satisfies

$$\partial_t \rho_t(\mathbf{x}) = -\nabla \cdot (\rho_t(\mathbf{x}) \nabla U_t(\mathbf{x})) + U_t(\mathbf{x}) \rho(\mathbf{x}). \quad (108)$$

That this is the case can be read from [25, Section A.3] or [53, Proposition 2.2]. This can be interpreted as a continuous dynamics where $U_t(\mathbf{x})$ is the *fitness* of state \mathbf{x} . The rate at which agents reproduce is prescribed by $U_t(\mathbf{x})$, and agents migrate to regions of higher fitness following $\nabla U_t(\mathbf{x})$. Theoretically, for a regular enough sequence of population snapshots $\{\rho_t\}_t$, a single time-dependent fitness function U_t is sufficient to generate the path $t \mapsto \rho_t$ via these dynamics. While it is perhaps not obvious, a *single* quantity, the fitness U_t , is enough to generate the full path ρ_t in the space of measures. As a baseline, we therefore propose a neural parameterisation of U_t and to learn U_t following the TIGON++ setup, but with $\mathbf{v}_t(\mathbf{x}) = \nabla U_t(\mathbf{x})$ and $g_t(\mathbf{x}) = U_t(\mathbf{x})$. We parameterise $U_t(\mathbf{x}) = \text{NN}_\theta(d + 1, 1)(t, \mathbf{x})$ using ReLU activations. All hyperparameter choices are listed in Table 6.

C.4 Training: TIGON++

The problem of dynamical transport for systems with mass imbalance was previously studied in the work [24]. In this work, the authors consider *deterministic* systems only and allow both the force

$v_t(\mathbf{x})$ and growth $g_t(\mathbf{x})$ to be time dependent. However, the TIGON algorithm relies on kernel density estimation (KDE) from the input data [24, Methods] which is sensitive to the choice of kernel bandwidth and suffers from the curse of dimensionality as the dimension increases. The choice of data-fitting loss, in the form of minimising squared discrepancies between the predicted and KDE densities, adds to these difficulties: this loss relies point-wise on estimated densities and is thus not “geometry-aware” in the sense of optimal transport based losses [52]. Finally, propagating densities under flow models e.g. (4) are well known to be computationally costly. For these reasons, training the TIGON algorithm was infeasible for most of our numerical experiments

Because we needed a robust comparison baseline, we decided to implement the same model used by TIGON (deterministic transport with growth), but train it with the UPFI training procedure. With UPFI, we circumvent the need for density estimation by using the probability flow formulation of the Fokker-Planck equation. Together with the use of the unbalanced Sinkhorn divergence as the data fitting loss, we believe that this substantially improves the TIGON method and also makes for a more rigorous baseline to test against UPFI. We call TIGON++ this re-implementation of TIGON.

Specifically, we parameterise a drift $v_\theta(t, \mathbf{x}) = \text{NN}_\theta(d+1, d)(t, \mathbf{x})$ and growth $g_\theta(t, \mathbf{x}) = \text{NN}_\theta(d+1, 1)(t, \mathbf{x})$ with ReLU activations. For a sampled data point $(\mathbf{x}_0, m_0 = 1)$ at $t = 0$, its state $(\mathbf{x}_{t_i}, m_{t_i})$ at each timepoint t_i is simulated by forward integration of the system

$$\dot{\mathbf{x}}_t = v_\theta(t, \mathbf{x}_t), \quad \dot{m}_t = g_\theta(t, \mathbf{x}_t)m_t, \quad (109)$$

and we form the empirical distribution $\hat{\rho}_{t_i}(\mathbf{x}) = \sum_{k=1}^{N_i} m_{k,t_i} \delta(\hat{\mathbf{x}}_{k,t_i} - \mathbf{x})$ for each timepoint t_i . For the rest of the training procedure we use the same data-fitting loss as UPFI, i.e. (12).

C.5 DeepRUOT

We use the existing DeepRUOT implementation provided by [27], which parameterises the force, growth rate and score function. Different to our method, however, the individual components of the dynamics are coupled via a physics-informed neural network (PINN)-type loss ([27, Section 5.3]) that aims to incorporate information from the governing Fokker-Planck equation. The training procedure DeepRUOT consists of multiple stages and involves first neural ODE training, followed by flow matching, and again neural ODE training. For the bistable system example (Fig. 3) we use their PyTorch implementation of [27, Algorithm 1]. For each of the drift, growth and score networks, three hidden layers of size 128 were used. For further details on DeepRUOT implementation and training we refer the reader to [27] and accompanying code.

C.6 Flow matching

We consider optimal transport-conditioned flow matching (OTFM), a class of *simulation-free* methods for training flows that approximate dynamical OT [38]. Compared to the other methods, OTFM training does not require numerical integration during training. This comes at the cost of requiring the transport plan $\pi_{i,i+1}$, or coupling, between successive snapshots $(\rho_{t_i}, \rho_{t_{i+1}})$ to be computed in advance of training the flow model. Given a pair of distributions (ρ, ρ') with a OT coupling π on $t \in [0, 1]$, OTFM trains a flow network $v_\theta(t, \mathbf{x})$ by minimising a nonlinear least squares objective:

$$\min_{\theta} \mathbb{E}_{t \in U[0,1]} \mathbb{E}_{(\mathbf{x}, \mathbf{x}') \sim \pi} \|(\mathbf{x}' - \mathbf{x}) - v_\theta(t, (1-t)\mathbf{x} + t\mathbf{x}')\|_2^2, \quad (110)$$

where π is the optimal coupling obtained by solving the entropic OT problem:

$$\min_{\pi: \pi \mathbf{1} = \rho, \pi^\top \mathbf{1} = \rho'} \frac{1}{2} \mathbb{E}_{(\mathbf{x}, \mathbf{x}') \sim \pi} \|\mathbf{x} - \mathbf{x}'\|_2^2 + \varepsilon \text{KL}(\pi | \rho \otimes \rho'). \quad (111)$$

We refer the reader to [38] for an in-depth discussion. We implement OTFM for dynamics inference by applying (110) across each pair of snapshots (t_i, t_{i+1}) to learn a single network $v_\theta(t, \mathbf{x})$. To generate results shown in Table 1 we compute entropic OT couplings between pairs of snapshots using a squared Euclidean cost and pick $\varepsilon = \sigma^2(t_{i+1} - t_i)$.

By default, OTFM treats the balanced case of transport and is not appropriate for modelling dynamics with growth. As an additional baseline, we try an unbalanced variant, UOTFM, where the coupling π is obtained by solving the *unbalanced* relaxation [23] of (111), where the hard marginal constraint is replaced with a soft penalty with weight $\lambda > 0$:

$$\min_{\pi} \frac{1}{2} \mathbb{E}_{(\mathbf{x}, \mathbf{x}') \sim \pi} \|\mathbf{x} - \mathbf{x}'\|_2^2 + \varepsilon \text{KL}(\pi | \rho \otimes \rho') + \lambda (\text{KL}(\pi \mathbf{1} | \rho) + \text{KL}(\pi^\top \mathbf{1} | \rho')). \quad (112)$$

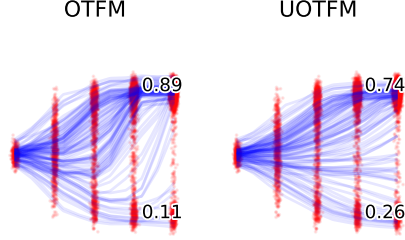


Figure 6: Sampled trajectories for trained flow matching models for bifurcating example, $d = 10$ (see Figure 4).

In practice we take $\lambda = N^{-2} \sum_{i,j} \|x_i - x'_j\|^2$, i.e. the mean of the cost matrix. This is essentially the approach advocated for in [26]. In Figure 6 we show sampled trajectories from trained OTFM and UOTFM models for the bifurcating system of Section 3. As expected, OTFM results in trajectories biased towards the upper branch, while this is partially mitigated using UOTFM.

C.7 Forward simulation

Bistable system We consider a potential-driven dynamics in dimension $d \in \{2, 5, 10\}$ specified by

$$\begin{aligned} \mathbf{v} &= -\nabla V, & V(\mathbf{x}) &= 0.9\|\mathbf{x} - \mathbf{a}\|_2^2 \|\mathbf{x} - \mathbf{b}\|_2^2 + 10 \sum_{i=3}^d x_i^2 \\ b(\mathbf{x}) &= \frac{5}{2}(1 + \tanh(2x_0)), \\ d(\mathbf{x}) &= 0. \end{aligned}$$

For $t \in [0, 1]$ we simulate particles following $d\mathbf{X}_t = \mathbf{v}(\mathbf{X}_t) dt + \sigma d\mathbf{B}_t$ using the Euler-Maruyama method. We set the noise level to $\sigma = 1/2$, and use the initial condition $\mathbf{X}_0 \sim \mathcal{N}(\mathbf{0}, 0.01\mathbf{I})$. At each Euler step, simulated particles divide with probability $b(\mathbf{X})\Delta t$. We simulate starting from $N_0 = 500$ particles, and the total population size grows over time following the prescribed dynamics. Population snapshots are taken from *independent* realisations of the process at $K + 1 = 5$ timepoints uniformly spaced between $[0, 1]$.

Reaction network systems The bifurcating and HSC reaction networks were taken from previous literature [40, 41], corresponding to the networks BF and HSC in the collection of BoolODE benchmarking problems [40]. The original implementation, however, modelled both gene and protein expression levels and as a result does not strictly fall in the modelling framework we consider. This is because protein levels are not observed and thus are hidden variables. We re-implemented each of these systems to involve only gene expression dynamics, and also change the noise model: in the original implementation an ad-hoc square-root noise model was used, i.e. $\sigma(\mathbf{x}) = \alpha\sqrt{\mathbf{x}}$. We choose to use a more biophysically motivated noise model (16), and take

$$\sigma_i(\mathbf{x}) = \sqrt{f_i(\mathbf{x}) + \lambda_i x_i},$$

where $\mathbf{f}(\mathbf{x})$ is a vector-valued function of state-dependent production rates for each gene x_i , and each λ_i is the corresponding degradation rate. For the growth rates, we consider a scenario where cells in one branch of the system trajectory divide at a faster rate than the others:

- In the bifurcating network, we set

$$\beta(x) = 5 \left(\frac{\tanh(5(x_7 - 0.7)) + 1}{2} \right) \left(1 - \frac{\tanh(5(x_1 - 0.7)) + 1}{2} \right)$$

- In the HSC network, we set

$$\beta(x) = 5.5 \left(\frac{\tanh(x_E - 1.5) + 1}{2} \right).$$

We simulate both systems using the same code as for the bistable system, with the volume parameter $1/\sqrt{V} = 0.5$, starting with a population of 500 cells and capturing $K + 1 = 10$ timepoints. For bifurcating and HSC network we use simulation time intervals of $0 \leq t \leq 1.25$ and $0 \leq t \leq 1$ respectively.

Fate probability computation We provide a straightforward definition of fate probability in what follows. Let \mathbf{x}_t be the state of an observed cell or individual at time t . Let $\sqcup_i \Omega_i$ be a partitioning of the state space (e.g. some subset of \mathbb{R}^d) where each Ω_i is understood to correspond to well-defined, stable states of the system at some final time, say $t = 1$. In the biological setting, this is typically thought of as a mature cell “type” [1, 34]. Then the fate probability of \mathbf{x}_t towards Ω_i is defined as the conditional probability:

$$\mathbb{P}(\mathbf{X}_1 \in \Omega_i | \mathbf{X}_t = \mathbf{x}_t).$$

In practice, such as for the bistable system of Fig. 3, we form a partitioning of the state space into two regions by running k -means with $k = 2$ on the final snapshot from the system. Given any query state \mathbf{x}_t at time t , we empirically estimated true and inferred fate probabilities by forward simulation of either the ground truth SDE or inferred dynamics:

$$\mathbb{P}(\mathbf{X}_1 \in \Omega_i | \mathbf{X}_t = \mathbf{x}_t) \approx M^{-1} \sum_{i=1}^M \mathbf{1}_{\Omega_i}(\mathbf{X}_1 | \mathbf{X}_t = \mathbf{x}_t),$$

where M is the number of trials to sample.

C.8 Neural graphical model

For the Neural Graphical Model (NGM) example of Fig. 4, we use the architecture introduced in [42] as a drop-in parameterisation of the autonomous force $\mathbf{v}_\theta(\mathbf{x})$. For each output variable, we use two hidden layers with sizes [64, 64]. We use a group lasso regularisation strength $\lambda_{\text{GL}} = 0.03$ and employ the proximal update scheme outlined in [42, Section C.4.1] with a learning rate of 0.003 and train for 5,000 iterations. We use the score networks that were already pre-trained for the additive and multiplicative UPFI models. All other training details are taken to be the same as for the earlier UPFI training.

C.9 Single cell lineage tracing data

Preprocessing Data for the study of [1] are available from the original publication using the GEO database with accession number GSE140802. Starting from raw counts, expression data is normalised using `dyn.pp.recipe_monocle` function from the Dynamo package [43]. In brief, raw gene expression values are per-cell normalised and then $\log(1 + x)$ -transformed. For all our experiments we use the 10-dimensional PCA embedding of cell gene expression profiles. Spliced and unspliced transcript counts were obtained from reanalysis of the raw sequencing data of [1] and RNA velocity estimates were subsequently obtained using the Dynamo package [43]. Scripts and datasets for this re-analysis are available upon request.

From the full dataset, 86,416 cells deemed to be contributing to the “Neutrophil-Monocyte” trajectory (as determined by the original publication [1]) were selected. Using the 10-dimensional PCA embedding for these data, we apply UPFI, PFI, fitness-ODE and TIGON. We do not include DeepRUOT in this analysis since it resulted in an out-of-memory error in the initial stages of training. Noting also that the original publication [27] considered only the 2D SPRING layout, we believe that further modification of its training pipeline may be necessary.

Training For UPFI, we train a time-dependent score model with hidden dimensions [128, 128, 128] for 10,000 iterations with a batch size of 256 and learning rate of 10^{-2} . We adopt an additive noise model and parameterise an autonomous force $\mathbf{v}_\theta(\mathbf{x})$ and growth $g_\theta(\mathbf{x})$ each with a MLP with hidden dimensions [128, 128, 128]. We set $\gamma = 25.0$, $\lambda = 0.001$, $\alpha = 1.0$ and we set $\sigma = 0.5$. Note that the choice of γ is not scale-invariant, and we found that the typical length scale in the lineage tracing data is larger than in the simulation data. We train for 10,000 iterations with a batch size of 256 and learning rate 10^{-3} .

We train PFI with the same hyperparameter choices as UPFI, except we use a non-autonomous force as done in earlier examples. Finally, we train TIGON and fitness-ODE following the training procedure outlined in Sections C.4 and C.3 and the same hyperparameters as for UPFI and PFI.

d	Score matching	UPFI	PFI	ODE	TIGON
25	249.73 ± 0.50	439.48 ± 1.51	333.51 ± 2.02	861.41 ± 11.43	313.79 ± 3.71
50	250.71 ± 1.52	443.52 ± 2.23	335.94 ± 3.65	868.93 ± 11.86	316.18 ± 4.17
d	OTFM	UOTFM			
25	100.68 ± 0.84	52.64 ± 0.91			
50	99.61 ± 0.68	52.95 ± 0.75			

Table 7: Runtimes (s) for the different methods on the bistable system, for $d = 25, 50$.

Cosine similarity force field				Growth rate correlation			
	$\lambda = 0.0$	0.01	0.1		0.0	0.01	0.1
$\alpha = 0$	0.105	0.059	0.078	0	0.923	0.917	0.917
0.01	0.105	0.059	0.060	0.01	0.923	0.917	0.899
0.1	0.105	0.086	0.203	0.1	0.923	0.909	0.349

Table 8: Results for regularisation ablation experiments for the bistable system with $d = 10$.

C.10 Remark on UPFI in the case when only frequencies are available

When N_i/N_0 is not a good estimator for the ratio $|\rho_{t_i}|/|\rho_0|$, we can only build an estimator for the normalised density $\rho_{t_i}/|\rho_{t_i}|$:

$$\frac{\rho_{t_i}}{|\rho_{t_i}|} \simeq \frac{1}{N_i} \sum_{k=1}^{N_i} \delta(\mathbf{x} - \mathbf{x}_{k,t_i}) \quad (113)$$

This poses limitations on the fitness which can be inferred. Indeed, writing $\tilde{\rho}_t = \rho_t/|\rho_t|$, we have

$$\partial_t \tilde{\rho}_t = \partial_t \left(\frac{\rho_t}{|\rho_t|} \right) = \frac{1}{|\rho_t|} \partial_t \rho_t - (\partial_t \log |\rho_t|) \tilde{\rho}_t.$$

Substituting back the PDE governing ρ_t , we find that $\tilde{\rho}_t$ is also governed by a drift-diffusion PDE, but with a time-dependent bias in the source term:

$$\partial_t \tilde{\rho}_t = -\nabla \cdot [\tilde{\rho}_t (\mathbf{v}_t - \nabla \cdot \mathbf{D}_t - \mathbf{D}_t \nabla \log \tilde{\rho}_t)] + (g_t - \partial_t \log |\rho_t|) \tilde{\rho}_t.$$

Therefore, when we don't have access to the absolute number of individuals present in a population at a given time, we can only hope to infer the fitness g_t up to a time-dependent bias. In this case, the UPFI approach can also be applied using a large enough mass conservation strength q in the unbalanced Sinkhorn distance.

C.11 Ablation experiments

We performed ablation experiments for the regularisation on the bifurcating system with $d = 10$. In Table 8, we show for varying α and λ the cosine error for force recovery as well as Pearson correlation for growth rate recovery. When $\lambda = 0$, there is no regularisation in the loss function, and the resulting error is larger than for $\lambda > 0$. From the growth rate perspective, however, the unregularised model performs better, albeit at the cost of higher force field error.

These results suggest that the growth rate is less sensitive to regularisation, which probably stems from the implicit regularisation arising from the relatively small neural network sizes. Since the additional loss terms are included only to ensure uniqueness, as motivated by the theorem in the main text, this ablation study suggests a simple rule of thumb for UPFI: use little or no regularisation when employing moderately sized neural networks.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: The main claims made in the abstract and introduction accurately reflect our paper's contributions and scope.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: The paper discusses the theoretical limitations of the method, in particular issues of identifiability.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We provide in the supplementary materials a complete proof for each of the claims made in the main text.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide full details on architectures and hyperparameters necessary to reproduce the results presented in experiments. These details concern not only the method novel to this paper, but also other methods to which it is compared. The paper also fully discloses the algorithm used for training, and provides guidelines in appendix on how to train it. The code used for all experiments will be made public on a github repository after the revision process.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

- (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The lineage tracing data used in the experiments is publicly accessible on the GEO database with the query code GSE140802. The code used for all experiments will be made public on a github repository after the revision process.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so No is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The paper specifies in appendix all the architectures and hyperparameters used in the experiments. This information is sufficient to interpret and reproduce the results.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: The paper provides error bars for each of the comparison carried out. The error bars are also consistently defined as a the standard deviation of the observable, measured on repeated experiments using different random seeds.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: The neural networks used in this paper are small and can run on a single GPU at a time. This information is disclosed in the appendix related to the experimental settings.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: We have reviewed the code of ethics, and the research conducted in this paper conforms to its requirements.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: Our work addresses some fundamental inference problems for population dynamics models, and we see no potential for direct societal impacts of our works, either positive or negative.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Not applicable.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All assets used in our work are properly attributed in the main text or supplemental material.

Guidelines:

- The answer NA means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: A new algorithm is detailed, along with the necessary training details (in the appendix).

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: Not applicable.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: Not applicable.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: Not applicable.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.