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*

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

where $\boldsymbol{A}_t \in \mathbb{R}^{d\times d}$, $\boldsymbol{e}_t, \boldsymbol{c}_t \in \mathbb{R}^d$, and $b_t \in \mathbb{R}$ are general, $\boldsymbol{D}_t \in \mathbb{R}^{d\times d}$ 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 $t \geqslant 0$, where

$$\dot{\boldsymbol{\Sigma}}_t = \boldsymbol{A}_t \boldsymbol{\Sigma}_t + \boldsymbol{\Sigma}_t \boldsymbol{A}_t^\top + 2\boldsymbol{D}_t, \tag{18}$$

$$\dot{\boldsymbol{\mu}}_t = \boldsymbol{A}_t \boldsymbol{\mu}_t + \boldsymbol{\Sigma}_t \boldsymbol{c}_t + \boldsymbol{e}_t, \tag{19}$$

$$\frac{\dot{m}_t}{m_t} = \boldsymbol{c}_t^\top \boldsymbol{\mu}_t + b_t. \tag{20}$$

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

$$\mathcal{F}\left(\nabla \cdot (\boldsymbol{A}_t \boldsymbol{x} + \boldsymbol{e}_t) p_t\right) = -\boldsymbol{k}^\top \boldsymbol{A}_t \nabla_{\boldsymbol{k}} \widehat{p}_t + i \boldsymbol{e}_t^\top \boldsymbol{k} \widehat{p}_t, \tag{21}$$

$$\mathcal{F}\left(\nabla \cdot (\boldsymbol{D}_t \nabla p_t)\right) = -\boldsymbol{k}^\top \boldsymbol{D}_t \boldsymbol{k} \widehat{p}_t. \tag{22}$$

Such that the Fourier transform of the PDE reads

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

We define the characteristics as the solution of

$$\frac{d\boldsymbol{k}_s}{ds} = -\boldsymbol{A}_s^\top \boldsymbol{k}_s - i \boldsymbol{c}_s, \tag{24}$$

such that

$$\frac{d\widehat{p}_s(\boldsymbol{k}_s)}{ds} = \left(\frac{d\boldsymbol{k}_s}{ds}\right)^\top \nabla_{\boldsymbol{k}} \widehat{p}_s(\boldsymbol{k}_s) + \partial_s \widehat{p}_s(\boldsymbol{k}_s) \tag{25}$$

$$= -\boldsymbol{k}_s^\top \boldsymbol{D}_s \boldsymbol{k}_s \widehat{p}_s + (b_s - i \boldsymbol{e}_s^\top \boldsymbol{k}_s) \widehat{p}_s. \tag{26}$$

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 $\mathbb{R}^+$ [41]. The state transition matrix has the following properties [42]:

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

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

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

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

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

$$\widehat{p}_s(\boldsymbol{k}_s) = \exp\Bigg[-i \boldsymbol{k}_0^\top \boldsymbol{x}_0 - \frac{1}{2} \boldsymbol{k}_0^\top \boldsymbol{\Sigma}_0 \boldsymbol{k}_0 - \boldsymbol{k}_0^\top \int_0^s \Phi_u^\top \boldsymbol{D}_u \Phi_u du \boldsymbol{k}_0 - 2i \boldsymbol{k}_0^\top \int_0^s \Phi_u^\top \boldsymbol{D}_u \widetilde{\boldsymbol{v}}_u du$$

$$+ \int_0^s \widetilde{\boldsymbol{v}}_u^\top \boldsymbol{D}_u \widetilde{\boldsymbol{v}}_u du + \int_0^s b_u du - i \int_0^s \boldsymbol{e}_u^\top \Phi_u \boldsymbol{k}_0 + \int_0^s \boldsymbol{e}_u^\top \widetilde{\boldsymbol{v}}_u du \Bigg]. \tag{30}$$

We introduce the following quantities

$$\boldsymbol{\Psi}_s = \int_0^s \Phi_u^\top \boldsymbol{D}_u \Phi_u du, \ \boldsymbol{q}_s^\top = \int_0^s \boldsymbol{e}_u^\top \Phi_u du, \ \boldsymbol{u}_s = \int_0^s \Phi_u^\top \boldsymbol{D}_u \widetilde{\boldsymbol{v}}_u du, \tag{31}$$

$$\gamma_s = \int_0^s \widetilde{\boldsymbol{v}}_u^\top \boldsymbol{D}_u \widetilde{\boldsymbol{v}}_u du, \ w_s = \int_0^s \boldsymbol{e}_u^\top \widetilde{\boldsymbol{v}}_u du. \tag{32}$$

Inverting the equation for characteristics we have $\boldsymbol{k}_0 = \Phi_s^{-1}\boldsymbol{k}_s + i\Phi_s^{-1}\int_0^s \Phi(s,u)\boldsymbol{c}_u du = \Phi_s^{-1}\boldsymbol{k}_s - i\Phi_s^{-1}\widetilde{\boldsymbol{v}}_s$. We can directly see that the Fourier transform is quadratic in $\boldsymbol{k}$, 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:

$$\boldsymbol{\Sigma}_s = \Phi_s^{-T}\boldsymbol{\Sigma}_0\Phi_s^{-1} + 2\Phi_s^{-T}\boldsymbol{\Psi}_s\Phi_s^{-1}, \tag{33}$$

$$\boldsymbol{\mu}_s = \Phi_s^{-T}(\boldsymbol{\mu}_0 + 2\boldsymbol{u}_s + \boldsymbol{q}_s) - \boldsymbol{\Sigma}_s\widetilde{\boldsymbol{v}}_s, \tag{34}$$

$$\log m_s = -\widetilde{\boldsymbol{v}}_s^\top\boldsymbol{\mu}_s - \frac{1}{2}\widetilde{\boldsymbol{v}}_s^\top\boldsymbol{\Sigma}_s\widetilde{\boldsymbol{v}}_s + w_s + \gamma_s + \int_0^s b_u du. \tag{35}$$

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

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*

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

where $\boldsymbol{A}_t \in \mathbb{R}^{d\times d}$, $\boldsymbol{e}_t, \boldsymbol{c}_t \in \mathbb{R}^d$, and $b_t \in \mathbb{R}$ are generic, $\boldsymbol{D}_t \in \mathbb{R}^{d\times d}$ is symmetric positive definite, and $\boldsymbol{\Gamma}_t \in \mathbb{R}^{d\times d}$ 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 $t \geqslant 0$, where

$$\dot{\boldsymbol{\Sigma}}_t = \boldsymbol{A}_t\boldsymbol{\Sigma}_t + \boldsymbol{\Sigma}_t\boldsymbol{A}_t^\top + 2\boldsymbol{D}_t + \boldsymbol{\Sigma}_t\boldsymbol{\Gamma}_t\boldsymbol{\Sigma}_t, \tag{9}$$

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

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

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

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

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

$$\boldsymbol{\Gamma}_t + 2\boldsymbol{G}_t\boldsymbol{D}_t\boldsymbol{G}_t - \boldsymbol{A}_t^\top\boldsymbol{G}_t - \boldsymbol{G}_t\boldsymbol{A}_t = \frac{d\boldsymbol{G}_t}{dt}. \tag{37}$$

Provided the coefficients are locally integrable on $\mathbb{R}^+$, that $\boldsymbol{\Gamma}_t \leqslant 0$ and $\boldsymbol{D}_t \geqslant 0$, if $\boldsymbol{G}_0 \leqslant 0$, then there exists a unique solution $\boldsymbol{G}_t$ on $\mathbb{R}^+$ for (37), and $\boldsymbol{G}_t \leqslant 0$ for all $t \geqslant 0$ [43]. We consider this unique solution with $\boldsymbol{G}_0 = 0$. Then, the equation for u_t becomes affine in growth

$$\partial_t u_t = -\nabla\cdot((\boldsymbol{A}_t\boldsymbol{x} + \boldsymbol{e}_t - 2\boldsymbol{D}_t\boldsymbol{G}_t\boldsymbol{x})u_t - \boldsymbol{D}_t\nabla u_t) + \left(b_t - \operatorname{tr}(\boldsymbol{G}_t\boldsymbol{D}_t) + \boldsymbol{c}_t^\top\boldsymbol{x} - \boldsymbol{e}_t^\top\boldsymbol{G}_t\boldsymbol{x}\right)u_t. \tag{38}$$

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

$$\partial_t u_t = -\nabla\cdot\left((\widetilde{\boldsymbol{A}}_t\boldsymbol{x} + \boldsymbol{e}_t)u_t - \boldsymbol{D}_t\nabla u_t\right) + \left(\widetilde{b}_t + \widetilde{\boldsymbol{c}}_t^\top\boldsymbol{x}\right)u_t. \tag{39}$$

We can apply the results of Prop. A.1. Since $u_0(\boldsymbol{x}) = p_0(\boldsymbol{x})$ is a Gaussian measure, it remains Gaussian, and we denote its covariance $\widetilde{\boldsymbol{\Sigma}}_t$, its mean $\widetilde{\boldsymbol{x}}_t$ and its mass $\widetilde{m}_t$. It follows that $p_t(\boldsymbol{x})$ is Gaussian measure for all $t \geqslant 0$ since $\widetilde{\boldsymbol{\Sigma}}_t^{-1} - \boldsymbol{G}_t$ is positive definite as the sum of positive definite and positive semi-definite terms. The covariance of this Gaussian measure is then $\boldsymbol{\Sigma}_t = (\widetilde{\boldsymbol{\Sigma}}_t^{-1} - \boldsymbol{G}_t)^{-1}$.

We have the following fact for any differentiable matrix valued function $\boldsymbol{B}_t$ invertible for all t:

$$\frac{d\boldsymbol{B}_t^{-1}}{dt} = -\boldsymbol{B}_t^{-1}\frac{d\boldsymbol{B}_t}{dt}\boldsymbol{B}_t^{-1}. \tag{40}$$

Applying this to the covariance, we find:

$$\frac{d\boldsymbol{\Sigma}_t}{dt} = -\boldsymbol{\Sigma}_t\frac{d\boldsymbol{\Sigma}_t^{-1}}{dt}\boldsymbol{\Sigma}_t = \boldsymbol{\Sigma}_t\widetilde{\boldsymbol{\Sigma}}_t^{-1}\left(\widetilde{\boldsymbol{A}}_t\widetilde{\boldsymbol{\Sigma}}_t + \widetilde{\boldsymbol{\Sigma}}_t\widetilde{\boldsymbol{A}}_t^\top + 2\boldsymbol{D}_t\right)\widetilde{\boldsymbol{\Sigma}}_t^{-1}\boldsymbol{\Sigma}_t + \boldsymbol{\Sigma}_t\frac{d\boldsymbol{G}_t}{dt}\boldsymbol{\Sigma}_t. \tag{41}$$

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

$$\frac{d\boldsymbol{\Sigma}_t}{dt} = \boldsymbol{A}_t\boldsymbol{\Sigma}_t + \boldsymbol{\Sigma}_t\boldsymbol{A}_t^\top + 2\boldsymbol{D}_t + \boldsymbol{\Sigma}_t\boldsymbol{\Gamma}_t\boldsymbol{\Sigma}_t. \tag{42}$$

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

$$\boldsymbol{\mu}_t = \left(\widetilde{\boldsymbol{\Sigma}}_t^{-1} - \boldsymbol{G}_t\right)^{-1}\widetilde{\boldsymbol{\Sigma}}_t^{-1}\widetilde{\boldsymbol{\mu}}_t = \boldsymbol{\Sigma}_t\widetilde{\boldsymbol{\Sigma}}_t^{-1}\widetilde{\boldsymbol{\mu}}_t = (\boldsymbol{I} + \boldsymbol{\Sigma}_t\boldsymbol{G}_t)\widetilde{\boldsymbol{\mu}}_t. \tag{43}$$

Using Prop. A.1 we have

$$\frac{d\boldsymbol{x}_t}{dt} = (\frac{\boldsymbol{\Sigma}_t}{dt}\boldsymbol{G}_t + \boldsymbol{\Sigma}_t\frac{d\boldsymbol{G}_t}{dt})\widetilde{\boldsymbol{\mu}}_t + (I + \boldsymbol{\Sigma}_t\boldsymbol{G}_t)(\widetilde{\boldsymbol{A}}_t\widetilde{\boldsymbol{\mu}}_t + \widetilde{\boldsymbol{\Sigma}}_t\widetilde{\boldsymbol{c}}_t + \widetilde{\boldsymbol{e}}_t) \tag{44}$$

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

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

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 [44, Theorem 1.5]. We find that

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

where the expectation is understood to be taken with respect to $\mathcal{N}(\boldsymbol{x}_t, \boldsymbol{\Sigma}_t)$. □

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

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

□

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 $\Delta t \to 0$ we have $\Delta t^{-1} L \to \mathcal{L}$, where $\mathcal{L}$ is the continuous time loss function:

$$\mathcal{L} = q^{-1} \int_0^T \mathrm{d}t \; m_t \Bigg(\|\boldsymbol{v}_t - \widehat{\boldsymbol{v}}_t\|^2_{\boldsymbol{X}_t^{-1}} + \frac{1}{2}(h_t - \widehat{h}_t)^2 \tag{13}$$

$$+ q \sum_{i,j} \frac{\sigma_{i,t}^2 \sigma_{j,q,t}^2}{(\sigma_{j,q,t}^2 \sigma_{i,t}^2 + \sigma_{i,q,t}^2 \sigma_{j,t}^2)^2} \left(\boldsymbol{w}_{i,t}^\top (\boldsymbol{B}_t - \widehat{\boldsymbol{B}}_t) \boldsymbol{w}_{j,t} \right)^2 \Bigg) + \lambda \int_0^T R_t \, \mathrm{d}t \tag{14}$$

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

$$\begin{aligned}
\boldsymbol{B}_t - \widehat{\boldsymbol{B}}_t &= \boldsymbol{\Sigma}_t (\boldsymbol{A}_t - \widehat{\boldsymbol{A}}_t)^\top + (\boldsymbol{A}_t - \widehat{\boldsymbol{A}}_t)\boldsymbol{\Sigma}_t + \boldsymbol{\Sigma}_t (\boldsymbol{\Gamma}_t - \widehat{\boldsymbol{\Gamma}}_t)\boldsymbol{\Sigma}_t + 2(\boldsymbol{D}_t - \widehat{\boldsymbol{D}}_t), \\
\boldsymbol{v}_t - \widehat{\boldsymbol{v}}_t &= ((\boldsymbol{A}_t - \widehat{\boldsymbol{A}}_t) + \boldsymbol{\Sigma}_t (\boldsymbol{\Gamma}_t - \widehat{\boldsymbol{\Gamma}}_t))\boldsymbol{\mu}_t + \boldsymbol{\Sigma}_t (\boldsymbol{c}_t - \widehat{\boldsymbol{c}}_t) + (\boldsymbol{e}_t - \widehat{\boldsymbol{e}}_t), \\
h_t - \widehat{h}_t &= \frac{1}{2}\boldsymbol{\mu}_t^\top (\boldsymbol{\Gamma}_t - \widehat{\boldsymbol{\Gamma}}_t)\boldsymbol{\mu}_t + (\boldsymbol{c}_t - \widehat{\boldsymbol{c}}_t)^\top \boldsymbol{\mu}_t + (b_t - \widehat{b}_t) + \frac{1}{2}\operatorname{tr}((\boldsymbol{\Gamma}_t - \widehat{\boldsymbol{\Gamma}}_t)\boldsymbol{\Sigma}_t),
\end{aligned} \tag{15}$$

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 $t \mapsto (\widehat{\boldsymbol{A}}_t^\star, \widehat{\boldsymbol{e}}_t^\star, \widehat{b}_t^\star, \widehat{\boldsymbol{c}}_t^\star, \widehat{\boldsymbol{\Gamma}}_t^\star)$.

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

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

where we have

$$\boldsymbol{X}_{t_i} = \boldsymbol{\Sigma}_{t_i} + \boldsymbol{\Sigma}_{t_i} + q^{-1}\boldsymbol{I}, \; \boldsymbol{J} = (\boldsymbol{\Sigma}_{t_i,q} \boldsymbol{\Sigma}_{t_i,q})^{1/2} (\boldsymbol{I} - q(\boldsymbol{\Sigma}_{t_i} \boldsymbol{\Sigma}_{t_i,q}^{-1} \boldsymbol{\Sigma}_{t_i,q}^{-1} \boldsymbol{\Sigma}_{t_i})^{1/2}), \tag{48}$$

with

$$\boldsymbol{\Sigma}_{t_i,q} = q\boldsymbol{\Sigma}_{t_i} + \boldsymbol{I}, \; \boldsymbol{\Sigma}_{t_i,q} = q\boldsymbol{\Sigma}_{t_i} + \boldsymbol{I}. \tag{49}$$

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 $\det \boldsymbol{J}$. We denote $\boldsymbol{\Sigma}_{\Delta t} = \boldsymbol{\Sigma}_0 + \Delta t \boldsymbol{A}$ and $\widetilde{\boldsymbol{\Sigma}}_{\Delta t} = \boldsymbol{\Sigma}_0 + \Delta t \widetilde{\boldsymbol{A}}$ and $\boldsymbol{\Sigma}_{0,q} = \boldsymbol{I} + q\boldsymbol{\Sigma}_0$. For this, we need to compute $\boldsymbol{\Sigma}_{\Delta t} \boldsymbol{\Sigma}_{\Delta t,q}^{-1} \widetilde{\boldsymbol{\Sigma}}_{\Delta t} \widetilde{\boldsymbol{\Sigma}}_{\Delta t,q}^{-1}$, up to second order in Δt. We have that

$$\begin{aligned}
\boldsymbol{\Sigma}_{\Delta t} \boldsymbol{\Sigma}_{\Delta t,q}^{-1} &= (\boldsymbol{\Sigma}_0 + \Delta t \boldsymbol{A})(\boldsymbol{\Sigma}_{0,q}^{-1} - q\Delta t \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-1} + q^2 \Delta t^2 \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-1}) && (50) \\
&= \boldsymbol{\Sigma}_0 \boldsymbol{\Sigma}_{0,q}^{-1} + \Delta t (\boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-1} - q\boldsymbol{\Sigma}_0 \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-1}) && (51) \\
&\quad + \Delta t^2 q (q\boldsymbol{\Sigma}_0 \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-1} - \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-1}). && (52)
\end{aligned}$$

We also have

$$\begin{aligned}
\widetilde{\boldsymbol{\Sigma}}_{\Delta t,q}^{-1} \widetilde{\boldsymbol{\Sigma}}_{\Delta t} &= (\boldsymbol{\Sigma}_{0,q}^{-1} - q\Delta t \boldsymbol{\Sigma}_{0,q}^{-1} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1} + q^2 \Delta t^2 \boldsymbol{\Sigma}_{0,q}^{-1} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1})(\boldsymbol{\Sigma}_0 + \Delta t \widetilde{\boldsymbol{A}}) && (53) \\
&= \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{\Sigma}_0 + \Delta t (\boldsymbol{\Sigma}_{0,q}^{-1} \widetilde{\boldsymbol{A}} - q\boldsymbol{\Sigma}_{0,q}^{-1} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{\Sigma}_0) && (54) \\
&\quad + \Delta t^2 q (q\boldsymbol{\Sigma}_{0,q}^{-1} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{\Sigma}_0 - \boldsymbol{\Sigma}_{0,q}^{-1} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1} \widetilde{\boldsymbol{A}}). && (55)
\end{aligned}$$

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

the term directly to its left. We therefore have

$$\boldsymbol{M} = \boldsymbol{\Sigma}_0 \boldsymbol{\Sigma}_{0,q}^{-2} \boldsymbol{\Sigma}_0 \tag{56}$$

$$\boldsymbol{H}_1 = \boldsymbol{\Sigma}_0 (\boldsymbol{\Sigma}_{0,q}^{-2} \widetilde{\boldsymbol{A}} - q \boldsymbol{\Sigma}_{0,q}^{-2} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{\Sigma}_0) + \text{tt}. \tag{57}$$

$$\boldsymbol{H}_{2345} = q \boldsymbol{\Sigma}_0 (q \boldsymbol{\Sigma}_{0,q}^{-2} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{\Sigma}_0 - \boldsymbol{\Sigma}_{0,q}^{-2} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1} \widetilde{\boldsymbol{A}}) + \text{tt}. \tag{58}$$

$$+ \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-2} \widetilde{\boldsymbol{A}} + q^2 \boldsymbol{\Sigma}_0 \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-2} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{\Sigma}_0 \tag{59}$$

$$- q \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-2} \widetilde{\boldsymbol{A}} \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{\Sigma}_0 + \text{tt}. \tag{60}$$

Using the Woodbury formula we also have $\boldsymbol{\Sigma}_0 \boldsymbol{\Sigma}_{0,q}^{-1} = q^{-1}(\boldsymbol{I} - \boldsymbol{\Sigma}_{0,q}^{-1})$. We have that $\boldsymbol{\Sigma}_0$ and $\boldsymbol{\Sigma}_{0,q}^{-1}$ commute, and that $\boldsymbol{M}^{1/2} = \boldsymbol{\Sigma}_0 \boldsymbol{\Sigma}_{0,q}^{-1} = q^{-1}(\boldsymbol{I} - \boldsymbol{\Sigma}_{0,q}^{-1})$. We introduce $\boldsymbol{S}(\boldsymbol{Y})$ the unique solution $\boldsymbol{X}$ to the following Lyapunov equation

$$\boldsymbol{M}^{1/2} \boldsymbol{X} + \boldsymbol{X} \boldsymbol{M}^{1/2} = \boldsymbol{Y} \tag{61}$$

This solution is expressed in terms of an integral (and is linear in $\boldsymbol{Y}$), ie.

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

As a result, we have

$$\left(\boldsymbol{\Sigma}_{\Delta t} \boldsymbol{\Sigma}_{\Delta t, q}^{-1} \widetilde{\boldsymbol{\Sigma}}_{\Delta t, q}^{-1} \widetilde{\boldsymbol{\Sigma}}_{\Delta t} \right)^{1/2} = \boldsymbol{M}^{1/2} + \Delta t \boldsymbol{S}(\boldsymbol{H}_1) + \Delta t^2 \left(\boldsymbol{S}(\boldsymbol{H}_{2345}) - \boldsymbol{S}(\boldsymbol{S}(H_1)^2) \right) \tag{63}$$

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

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

where $\boldsymbol{L} = -q \boldsymbol{\Sigma}_{0,q} \boldsymbol{S}(\boldsymbol{H}_1)$ and $\boldsymbol{G} = -q \boldsymbol{\Sigma}_{0,q} \left(\boldsymbol{S}(\boldsymbol{H}_{2346}) - \boldsymbol{S}(\boldsymbol{S}(\boldsymbol{H}_1)^2) \right)$. We then have the following expansion at second order in Δt

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

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

$$\text{tr} \boldsymbol{L} = -q \text{tr} (\boldsymbol{\Sigma}_{0,q} \int_0^\infty e^{-\boldsymbol{M}^{1/2} t} \boldsymbol{H}_1 e^{-\boldsymbol{M}^{1/2} t} dt) = -\frac{q}{2} \text{tr} (\boldsymbol{M}^{-1/2} \boldsymbol{\Sigma}_{0,q} \boldsymbol{H}_1) \tag{66}$$

$$= -\frac{q}{2} \text{tr} (\boldsymbol{\Sigma}_{0,q}^2 \boldsymbol{\Sigma}_0^{-1} \boldsymbol{H}_1) = -\frac{q}{2} \left(\text{tr} (\boldsymbol{A} + \widetilde{\boldsymbol{A}}) - q \text{tr} ((\boldsymbol{A} + \widetilde{\boldsymbol{A}}) \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{\Sigma}_0) \right) \tag{67}$$

$$= -\frac{q}{2} \text{tr} (\boldsymbol{\Sigma}_{0,q}^{-1} (\boldsymbol{A} + \widetilde{\boldsymbol{A}})). \tag{68}$$

Computation of $\text{tr}\boldsymbol{G}$

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

$$\text{tr} (-q \boldsymbol{\Sigma}_{0,q} \boldsymbol{S}(\boldsymbol{H}_{2345})) = \frac{1}{2} \left(q^2 \text{tr} (\boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-1} \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-1} + \text{tt}.) - q \text{tr} (\boldsymbol{\Sigma}_0^{-1} \boldsymbol{A} \boldsymbol{\Sigma}_{0,q}^{-2} \widetilde{\boldsymbol{A}}) \right) \tag{69}$$

To compute $\text{tr}(\boldsymbol{\Sigma}_{0,q} \boldsymbol{S}(\boldsymbol{S}(\boldsymbol{H}_1)^2))$ we denote $\boldsymbol{M}^{1/2} = \sum_i u_i \boldsymbol{w}_i \boldsymbol{w}_i^\top$ the eigendecomposition of $\boldsymbol{M}^{1/2}$. We also denote $\boldsymbol{W}_i = \boldsymbol{w}_i \boldsymbol{w}_i^\top$. Therefore we have

$$\boldsymbol{S}(\boldsymbol{H}_1) = \sum_{i,j} \frac{1}{u_i + u_j} \boldsymbol{W}_i \boldsymbol{H}_1 \boldsymbol{W}_i \tag{70}$$

such that

$$\boldsymbol{\Sigma}_{0,q} \boldsymbol{S}(\boldsymbol{S}(\boldsymbol{H}_1)^2) = \sum_{i,j,k} \frac{(1 - u_k q)^2 (1 - u_j q)}{(u_i + u_k)(u_j + u_k)(u_i + u_j)} \boldsymbol{W}_i \left(u_i \widetilde{\boldsymbol{A}} + u_k \boldsymbol{A} \right) \boldsymbol{W}_k \left(u_k \widetilde{\boldsymbol{A}} + u_j \boldsymbol{A} \right) \boldsymbol{W}_j. \tag{71}$$

Taking the trace we have

$$\mathrm{tr}(\boldsymbol{\Sigma}_{0,q}\boldsymbol{S}(\boldsymbol{S}(\boldsymbol{H}_1)^2)) = \sum_{i,j}\frac{(1-u_jq)^2(1-u_iq)}{2u_i(u_i+u_j)^2}\mathrm{tr}\left(\boldsymbol{W}_i(u_i\widetilde{\boldsymbol{A}}+u_j\boldsymbol{A})\boldsymbol{W}_j(u_j\widetilde{\boldsymbol{A}}+u_i\boldsymbol{A})\right). \quad (72)$$

Additionally, we have that

$$\mathrm{tr}\left(\boldsymbol{W}_i(u_i\widetilde{\boldsymbol{A}}+u_j\boldsymbol{A})\boldsymbol{W}_j(u_j\widetilde{\boldsymbol{A}}+u_i\boldsymbol{A})\right) \quad (73)$$

$$= u_iu_j\mathrm{tr}\left(\boldsymbol{W}_i(\boldsymbol{A}-\widetilde{\boldsymbol{A}})\boldsymbol{W}_j(\boldsymbol{A}-\widetilde{\boldsymbol{A}})\right) + (u_i+u_j)^2\mathrm{tr}\left(\boldsymbol{W}_i\boldsymbol{A}\boldsymbol{W}_j\widetilde{\boldsymbol{A}}\right) \quad (74)$$

As a result we have

$$\mathrm{tr}\boldsymbol{G} = \frac{1}{2}q^2\mathrm{tr}(\boldsymbol{A}\boldsymbol{\Sigma}_{0,q}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}_{0,q}^{-1}+\mathrm{tt.}) + \frac{q}{2}\sum_{i,j}\frac{(1-u_jq)^2(1-u_iq)u_j}{(u_i+u_j)^2}\mathrm{tr}\left(\boldsymbol{W}_i(\boldsymbol{A}-\widetilde{\boldsymbol{A}})\boldsymbol{W}_j(\boldsymbol{A}-\widetilde{\boldsymbol{A}})\right) \quad (75)$$

Computation of $\mathrm{tr}\boldsymbol{L}^2$

Similarly we have

$$\boldsymbol{\Sigma}_{0,q}\boldsymbol{S}(\boldsymbol{H}_1) = \sum_{i,j}\frac{(1-u_jq)}{(u_i+u_j)}\boldsymbol{W}_i\left(u_i\widetilde{\boldsymbol{A}}+u_j\boldsymbol{A}\right)\boldsymbol{W}_j. \quad (76)$$

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

$$\mathrm{tr}((\boldsymbol{\Sigma}_{0,q}\boldsymbol{S}(\boldsymbol{H}_1))^2) = \sum_{i,j}\frac{(1-u_iq)(1-u_jq)u_iu_j}{(u_i+u_j)^2}\mathrm{tr}\left(\boldsymbol{W}_i(\boldsymbol{A}-\widetilde{\boldsymbol{A}})\boldsymbol{W}_j(\boldsymbol{A}-\widetilde{\boldsymbol{A}})\right) \quad (77)$$

$$+\mathrm{tr}(\boldsymbol{\Sigma}_{0,q}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}_{0,q}^{-1}\widetilde{\boldsymbol{A}}). \quad (78)$$

As a result we have

$$\mathrm{tr}\boldsymbol{L}^2 = q^2\mathrm{tr}(\boldsymbol{\Sigma}_{0,q}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}_{0,q}^{-1}\widetilde{\boldsymbol{A}}) + q^2\sum_{i,j}\frac{(1-u_iq)(1-u_jq)u_iu_j}{(u_i+u_j)^2}\mathrm{tr}\left(\boldsymbol{W}_i(\boldsymbol{A}-\widetilde{\boldsymbol{A}})\boldsymbol{W}_j(\boldsymbol{A}-\widetilde{\boldsymbol{A}})\right) \quad (79)$$

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

We have

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

Taking the determinant and Taylor expanding we have

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

$$\times\left(1+q\Delta t\mathrm{tr}(\boldsymbol{\Sigma}_{0,q}^{-1}(\boldsymbol{A}+\widetilde{\boldsymbol{A}}))+q^2\frac{\Delta t^2}{2}\left((\mathrm{tr}^2(\boldsymbol{\Sigma}_{0,q}^{-1}(\boldsymbol{A}+\widetilde{\boldsymbol{A}}))-\mathrm{tr}(\boldsymbol{\Sigma}_{0,q}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}_{0,q}^{-1}\boldsymbol{A}+\mathrm{tt.})\right)\right). \quad (82)$$

Expanding the square root we find

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

$$\times\left(1+q\frac{\Delta t}{2}\mathrm{tr}(\boldsymbol{\Sigma}_{0,q}^{-1}(\boldsymbol{A}+\widetilde{\boldsymbol{A}}))+q^2\frac{\Delta t^2}{8}\left(\mathrm{tr}^2(\boldsymbol{\Sigma}_{0,q}^{-1}(\boldsymbol{A}+\widetilde{\boldsymbol{A}}))-2\mathrm{tr}(\boldsymbol{\Sigma}_{0,q}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}_{0,q}^{-1}\boldsymbol{A}+\mathrm{tt.})\right)\right) \quad (84)$$

Computation of $\det \boldsymbol{J}$

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

$$\det \boldsymbol{J} = \left(1 + q\frac{\Delta t}{2}\mathrm{tr}(\boldsymbol{\Sigma}_{0,q}^{-1}(\boldsymbol{A} + \widetilde{\boldsymbol{A}})) + q^2\frac{\Delta t^2}{8}\left(\mathrm{tr}^2(\boldsymbol{\Sigma}_{0,q}^{-1}(\boldsymbol{A} + \widetilde{\boldsymbol{A}})) - 2\mathrm{tr}(\boldsymbol{\Sigma}_{0,q}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}_{0,q}^{-1}\boldsymbol{A} + \mathrm{tt.})\right)\right) \tag{85}$$

$$\times \left(1 + \Delta t \mathrm{tr}\boldsymbol{L} + \frac{\Delta t^2}{2}\left(\mathrm{tr}^2\boldsymbol{L} - \mathrm{tr}\boldsymbol{L}^2 + 2\mathrm{tr}\boldsymbol{G}\right)\right). \tag{86}$$

Using the result for $\mathrm{tr}\boldsymbol{L}$ 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

$$\det \boldsymbol{J} = \left(1 - q^2\frac{\Delta t^2}{4}\mathrm{tr}(\boldsymbol{\Sigma}_{0,q}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}_{0,q}^{-1}\boldsymbol{A} + \mathrm{tt.}) + \frac{\Delta t^2}{2}\left(-\mathrm{tr}\boldsymbol{L}^2 + 2\mathrm{tr}\boldsymbol{G}\right)\right). \tag{87}$$

Simplifying further we find that

$$\det \boldsymbol{J} = 1 + \frac{\Delta t^2 q}{2}\Bigg(\frac{q}{2}\mathrm{tr}(\boldsymbol{\Sigma}_{0,q}^{-1}(\boldsymbol{A} - \widetilde{\boldsymbol{A}})\boldsymbol{\Sigma}_{0,q}^{-1}(\boldsymbol{A} - \widetilde{\boldsymbol{A}})) \tag{88}$$

$$+ \sum_{i,j}\frac{u_j(1-u_iq)(1-u_jq)(1-q(u_j+u_i))}{(u_i+u_j)^2}\mathrm{tr}\left(\boldsymbol{W}_i(\boldsymbol{A} - \widetilde{\boldsymbol{A}})\boldsymbol{W}_j(\boldsymbol{A} - \widetilde{\boldsymbol{A}})\right)\Bigg) \tag{89}$$

This can be simplified

$$\Bigg(\frac{q}{2}\mathrm{tr}(\boldsymbol{\Sigma}_{0,q}^{-1}(\boldsymbol{A} - \widetilde{\boldsymbol{A}})\boldsymbol{\Sigma}_{0,q}^{-1}(\boldsymbol{A} - \widetilde{\boldsymbol{A}})) \tag{90}$$

$$+ \sum_{i,j}\frac{u_j(1-u_iq)(1-u_jq)(1-q(u_j+u_i))}{(u_i+u_j)^2}\mathrm{tr}\left(\boldsymbol{W}_i(\boldsymbol{A} - \widetilde{\boldsymbol{A}})\boldsymbol{W}_j(\boldsymbol{A} - \widetilde{\boldsymbol{A}})\right)\Bigg) \tag{91}$$

$$= \sum_{i,j}\frac{(1-qu_i)(1-qu_j)}{(u_i+u_j)^2}\left(\frac{q}{2}u_i^2 - \frac{q}{2}u_j^2 + u_j\right)\mathrm{tr}\left(\boldsymbol{W}_i(\boldsymbol{A} - \widetilde{\boldsymbol{A}})\boldsymbol{W}_j(\boldsymbol{A} - \widetilde{\boldsymbol{A}})\right). \tag{92}$$

which leaves us with the following simplification

$$\det \boldsymbol{J} = 1 + \frac{\Delta t^2 q}{2}\sum_{i,j}\frac{u_i(1-u_iq)(1-u_jq)}{(u_i+u_j)^2}\mathrm{tr}\left(\boldsymbol{W}_i(\boldsymbol{A} - \widetilde{\boldsymbol{A}})\boldsymbol{W}_j(\boldsymbol{A} - \widetilde{\boldsymbol{A}})\right)\Bigg). \tag{93}$$

Using $\boldsymbol{\Sigma}_0 = \sum_i \sigma_i^2 \boldsymbol{W}_i$, 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

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

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

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

Taylor expansion of the GHK term

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

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

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

$$\mathrm{GHK}\left(m_{\Delta t}\mathcal{N}(\boldsymbol{\mu}_{\Delta t}, \boldsymbol{\Sigma}_{\Delta t}), \widehat{m}_{\Delta t}\mathcal{N}(\widehat{\boldsymbol{\mu}}_{\Delta t}, \widehat{\boldsymbol{\Sigma}}_{\Delta t})\right) \tag{97}$$

$$= m_0 q^{-1}\Delta t^2\Bigg(\Big((\boldsymbol{v}-\widetilde{\boldsymbol{v}})^\top \boldsymbol{X}_0^{-1}(\boldsymbol{v}-\widetilde{\boldsymbol{v}}) + \frac{1}{2}(h-\widetilde{h})^2\Big) \tag{98}$$

$$+ q\sum_{i,j}\frac{\sigma_i^2\sigma_{j,q}^2}{(\sigma_{j,q}^2\sigma_i^2 + \sigma_{i,q}^2\sigma_j^2)^2}\mathrm{tr}\left(\boldsymbol{W}_i(\boldsymbol{A}-\widetilde{\boldsymbol{A}})\boldsymbol{W}_j(\boldsymbol{A}-\widetilde{\boldsymbol{A}})\right)\Bigg). \tag{99}$$

Integrating over all snapshots, the continuous time loss reads

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

The first order expansions $\boldsymbol{A}$, h, and $\boldsymbol{v}$ 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

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

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*

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

is strictly convex.

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

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

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

$$f(w(\alpha\theta_1+(1-\alpha)\theta_2, x)) = f(\alpha w(\theta_1, x)+(1-\alpha)w(\theta_2, x)) \leqslant \alpha f(w(\theta_1, x)+(1-\alpha)f(w(\theta_1, x)). \tag{104}$$

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.:

$$f(w(\alpha\theta_1+(1-\alpha)\theta_2, x)) = f(\alpha w(\theta_1, x)+(1-\alpha)w(\theta_2, x)) < \alpha f(w(\theta, x))+(1-\alpha)f(w(\theta, x)). \tag{105}$$

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

$$\ell(\alpha\theta_1 + (1-\alpha)\theta_2) < \alpha\ell(\theta_1) + (1-\alpha)\ell(\theta_2), \tag{106}$$

proving that ℓ is strictly convex. □

	Hidden layers	Batch size	Learning rate	Iterations
Bistable	64, 64, 64	256	0.01	25,000
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	64, 64, 64	64, 64, 64	256	0.003	5,000	0.01	0.1	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	64, 64, 64	–	256	0.003	5,000	0.01	–	–
	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	Coupled to growth	64, 64, 64	256	0.003	5,000	0.01	0.1	5
	Lineage	Coupled to growth	128, 128, 128	256	0.001	10,000	0.001	0.1	25
TIGON++	Bistable	64, 64, 64	64, 64, 64	256	0.003	5,000	0.01	0.1	5
	Lineage tracing	128, 128, 128	128, 128, 128	256	0.001	10,000	0.001	0.1	25

Table 6: Hyperparameter settings: dynamics

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

$$\theta \mapsto R_t(\theta) = \int \rho_t(\boldsymbol{x}) \left(\|\boldsymbol{v}_t(\boldsymbol{x})\|_2^2 + \alpha |g_t(\boldsymbol{x})|^2 \right) d\boldsymbol{x} \tag{107}$$

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 $t \mapsto \theta_t^\star$ for the functional $\mathcal{L}[\theta_t]$ [45]. □

C Implementation and experiment details

We implement Alg. 1 using PyTorch and employ the GeomLoss package [46] for computation of the unbalanced Sinkhorn divergence. All model training was carried out using a NVIDIA L40S GPU.

C.1 Score matching

While in principle any score matching approach to learn $\boldsymbol{s}_t(\boldsymbol{x}) = \nabla \log p_t(\boldsymbol{x})$ can be used within Alg. 1, in practice we employ denoising score matching [24] 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) $\boldsymbol{s}_\phi(t, \boldsymbol{x}, \eta) = \mathrm{NN}_\phi(d + 2, d)$ 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 $\eta_0 < \ldots < \eta_L$ 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 $\boldsymbol{s}_t(\boldsymbol{x})$ are parameterised using MLPs with ReLU activations. In all case, score training was carried out using noise-conditional denoising score matching [5, 24] 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 $\boldsymbol{v}_\theta(\boldsymbol{x})$ 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 [29]. We parameterise separately the force $\boldsymbol{v}_\theta(\boldsymbol{x})$

and growth $g_\theta(\boldsymbol{x})$ 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 $\boldsymbol{v}_\theta(t, \boldsymbol{x}) = \mathrm{NN}_\theta(d+1, d)(t, \boldsymbol{x})$ 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

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

and the resulting force is $(\boldsymbol{f}_\theta - \boldsymbol{g}_\theta)(\boldsymbol{x})$. 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 $(\boldsymbol{f}, \boldsymbol{g})$ and all other hyperparameter choices are as given in Table 6.

C.3 Training: TIGON++

The problem of dynamical transport for systems with mass imbalance was previously studied in the work [16]. In this work, the authors consider *deterministic* systems only and allow both the force $\boldsymbol{v}_t(\boldsymbol{x})$ and growth $g_t(\boldsymbol{x})$ to be time dependent. However, the TIGON algorithm relies on kernel density estimation (KDE) from the input data [16, 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 pointwise on estimated densities and is thus not "geometry-aware" in the sense of optimal transport based losses [46]. 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 $\boldsymbol{v}_\theta(t, \boldsymbol{x}) = \mathrm{NN}_\theta(d+1, d)(t, \boldsymbol{x})$ and growth $g_\theta(t, \boldsymbol{x}) = \mathrm{NN}_\theta(d+1, 1)(t, \boldsymbol{x})$ with ReLU activations. For a sampled data point $(\boldsymbol{x}_0, m_0 = 1)$ at $t = 0$, its state $(\boldsymbol{x}_{t_i}, m_{t_i})$ at each timepoint t_i is simulated by forward integration of the system

$$\dot{\boldsymbol{x}}_t = \boldsymbol{v}_\theta(t, \boldsymbol{x}_t), \qquad \dot{m}_t = g_\theta(t, \boldsymbol{x}_t) m_t, \tag{108}$$

and we form the empirical distribution $\widehat{\rho}_{t_i}(\boldsymbol{x}) = \sum_{k=1}^{N_i} m_{k,t_i} \delta(\widehat{\boldsymbol{x}}_{k,t_i} - \boldsymbol{x})$ 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.4 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 $U_t(\boldsymbol{x}_t)$ such that ρ_t satisfies

$$\partial_t \rho_t(\boldsymbol{x}) = -\nabla \cdot (\rho_t(\boldsymbol{x}) \nabla U_t(\boldsymbol{x})) + U_t(\boldsymbol{x}) \rho(\boldsymbol{x}). \tag{109}$$

That this is the case can be read from [47, Section A.3] or [48, Proposition 2.2]. This can be interpreted as a continuous dynamics where $U_t(\boldsymbol{x})$ is the *fitness* of state $\boldsymbol{x}$. The rate at which agents reproduce is prescribed by $U_t(\boldsymbol{x})$, and agents migrate to regions of higher fitness following $\nabla U_t(\boldsymbol{x})$. 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 $\boldsymbol{v}_t(\boldsymbol{x}) = \nabla U_t(\boldsymbol{x})$ and $g_t(\boldsymbol{x}) = U_t(\boldsymbol{x})$. We parameterise $U_t(\boldsymbol{x}) = \mathrm{NN}_\theta(d+1, 1)(t, \boldsymbol{x})$ using ReLU activations. All hyperparameter choices are listed in Table 6.

C.5 DeepRUOT

We use the existing DeepRUOT implementation provided by [32], 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 ([32, Section 5.3]) that aims to incorporate information from the governing Fokker-Planck equation. For the bistable system example (Fig. 3) we use their PyTorch implementation of [32, 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 [32] and accompanying code.

C.6 Forward simulation

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

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

For $t \in [0, 1]$ we simulate particles following $\mathrm{d}\boldsymbol{X}_t = \boldsymbol{v}(\boldsymbol{X}_t)\,\mathrm{d}t + \sigma\,\mathrm{d}\boldsymbol{B}_t$ using the Euler-Maruyama method. We set the noise level to $\sigma = 1/2$, and use the initial condition $\boldsymbol{X}_0 \sim \mathcal{N}(\boldsymbol{0}, 0.01\boldsymbol{I})$. At each Euler step, simulated particles divide with probability $b(\boldsymbol{X})\Delta t$. 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 [34, 35], corresponding to the networks `BF` and `HSC` in the collection of BoolODE benchmarking problems [34]. 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. $\boldsymbol{\sigma}(\boldsymbol{x}) = \alpha\sqrt{\boldsymbol{x}}$. We choose to use a more biophysically motivated noise model (16), and take

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

where $\boldsymbol{f}(\boldsymbol{x})$ 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

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

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 $0 \leqslant t \leqslant 1.25$ and $0 \leqslant t \leqslant 1$ respectively.

Fate probability computation We provide a straightforward definition of fate probability in what follows. Let $\boldsymbol{x}_t$ 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, 29]. Then the fate probability of $\boldsymbol{x}_t$ towards Ω_i is defined as the conditional probability:

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

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 $\boldsymbol{x}_t$ at time t, we empirically estimated true and inferred fate probabilities by forward simulation of either the ground truth SDE or inferred dynamics:

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

where M is the number of trials to sample.

C.7 Neural graphical model

For the Neural Graphical Model (NGM) example of Fig. 4, we use the architecture introduced in [36] as a drop-in parameterisation of the autonomous force $\boldsymbol{v}_\theta(\boldsymbol{x})$. 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 [36, 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.8 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 [37]. 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 [37]. 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 [32] considered only the 2D SPRING layout, be believe that further modification of its training pipeline may be necessary.

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 $\boldsymbol{v}_\theta(\boldsymbol{x})$ and growth $g_\theta(\boldsymbol{x})$ 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.3 and C.4 and the same hyperparameters as for UPFI and PFI.

C.9 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 $\rho_{t_i}/|\rho_{t_i}|$:

$$\frac{\rho_{t_i}}{|\rho_{t_i}|} \simeq \frac{1}{N_i} \sum_{k=1}^{N_i} \delta(\boldsymbol{x} - \boldsymbol{x}_{k,t_i}) \tag{110}$$

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

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

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

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

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.