Deterministic particle flows for constraining SDEs

Dimitra Maoutsa
Technical University of Berlin
& University of Potsdam
dimitra.maoutsa@tu-berlin.de

Manfred Opper
Technical University of Berlin
& University of Birmingham
manfred.opper@tu-berlin.de

Abstract

Devising optimal interventions for diffusive systems often requires the solution of the Hamilton-Jacobi-Bellman (HJB) equation, a nonlinear backward partial differential equation (PDE), that is, in general, nontrivial to solve. Existing control methods either tackle the HJB directly with grid-based PDE solvers, or resort to iterative stochastic path sampling to obtain the necessary controls. Here, we present a framework that interpolates between these two approaches. By reformulating the optimal interventions in terms of logarithmic gradients (*scores*) of two forward probability flows, and by employing deterministic particle methods for solving Fokker-Planck equations, we introduce a novel fully *deterministic* framework that computes the required optimal interventions in *one shot*.

1 Introduction

Constrained diffusions and optimal control. Consider the problem of imposing constraints $\mathcal{C}$ to the state of a stochastic system, whose *unconstrained* dynamics are described by a stochastic differential equation (SDE)

$$dX_t = f(X_t, t)dt + \sigma dW_t, \tag{1}$$

with drift $f(x,t) \in \mathcal{R}^d \times \mathcal{R} \rightarrow \mathcal{R}^d$, diffusion coefficient $\sigma \in \mathcal{R}$[1], and W denoting a d-dimensional vector of Wiener processes acting as white noise sources. For a time interval $[0,\, T]$, the constraints $\mathcal{C}$ may involve either terminal state X_T of the system through the function $\chi(x) \in \mathcal{R}^d \rightarrow \mathcal{R}$, or its transient state through a path constraining function $U(x,t) \in \mathcal{R}^d \times \mathcal{R} \rightarrow \mathcal{R}$ for $t \leq T$ that penalises or rewards specific regions of the state space.

One way to obtain the path probability measure $\mathbb{Q}$ of the *constrained* process is by *reweighting* paths $X_{0:T}$ generated from Eq.(1) over the interval $[0,\, T]$. Path weights result from the *Radon–Nikodym derivative* with respect to the path measure of the unconstrained process $\mathbb{P}_f$

$$\frac{d\mathbb{Q}}{d\mathbb{P}_f}(X_{0:T}) = \frac{\chi(X_T)}{Z} \exp\left[-\int_0^T U(X_t,t)dt\right], \tag{2}$$

where Z is a normalising constant.

How shall we modify the system of Eq.(1) to incorporate the desired constraints $\mathcal{C}$ into its dynamics, while also minimising the relative entropy between the path distributions of the constrained and unconstrained processes?

Problems of this form appear often in physics, biology, and engineering, and are relevant for calculation of rare event probabilities [1, 2], latent state estimation of partially observed systems [3–7], or for precise manipulation of stochastic systems to target states [8, 9] with applications on artificial selection [10, 11], motor control [12], epidemiology, and more [13–18]. Yet, although

[1]For brevity, we restrict ourselves to state- and time- independent scalar diffusions, but the framework easily generalises to time-dependent multiplicative noise settings with non-isotropic diffusion functions.

Fourth Workshop on Machine Learning and the Physical Sciences (NeurIPS 2021).

stochastic optimal control problems are prevalent in most scientific fields, their numerical solution remains computationally demanding for most practical problems.

The constrained process, defined by the weight of Eq.(2), can be also expressed as a diffusion process with the same diffusion σ, but with a modified time-dependent drift $g(x,t)$ [19, 20]. The computation of this drift adjustment or *control* $u(x,t) \doteq g(x,t) - f(x,t)$ amounts to solving a stochastic control problem to obtain the optimal interventions $u^*(x,t)$ that minimise the expected cost

$$\mathcal{J}(x,t) \doteq \min_u \mathbb{E}_{\mathbb{P}_g} \left[\int_0^T \left(\frac{1}{2\sigma^2} \|u(X_t,t)\|^2 + U(X_t,t) \right) dt - \ln \chi(X_T) \right]. \tag{3}$$

The expectation $\mathbb{E}_{\mathbb{P}_g}$ is over paths induced by the *constrained* SDE $dX_t = g(X_t,t)dt + \sigma dW_t$. (4). The first term in Eq.(3) results from minimising the Kullback-Leibler divergence between the path measures induced by the unconstrained and the constrained processes [21].

Finding the exact optimal controls for general stochastic control problems amounts to solving the HJB equation [22], a computationally demanding *nonlinear* second order PDE. However, control problems characterised by the cost of Eq.(3), known as *Path Integral-* or *Kullback-Leibler-control* (**PI/KL-control**)[4, 8, 21, 23], admit a simpler solution. For this class of problems, the logarithmic Hopf-Cole variable transformation [24], ie. $\mathcal{J}(x,t) = -\log(\phi_t(x))$, linearises the HJB equation [8], and the optimally adjusted drift becomes

$$g(x,t) = f(x,t) + \sigma^2 \nabla \ln \phi_t(x), \tag{5}$$

where $\phi_t(x)$ is a solution to the backward linear PDE

$$\frac{\partial \phi_t(x)}{\partial t} + \mathcal{L}_f^\dagger \phi_t(x) - U(x,t)\phi_t(x) = 0, \tag{6}$$

with terminal condition $\phi_T(x) = \chi(x)$, and $\mathcal{L}_f^\dagger \phi_t(x) \doteq f(x,t)\nabla \cdot \phi_t(x) + \frac{\sigma^2}{2}\nabla^2 \phi_t(x)$ denotes the *adjoint* Fokker–Planck operator acting on $\phi_t(x)$.

The PDE of Eq.(6) is often treated either with grid based solvers [25, 26], or with iterative stochastic path sampling frameworks [8, 21, 27–29]. Both approaches suffer, in general, from high computational complexity with increasing system dimension. (However note recent neural network advances towards this direction [30, 31].)

2 Method

Constrained diffusion densities from backward smoothing. To avoid directly solving the backward PDE (Eq. (6)), we view the *marginal density* $q_t(x)$ of the constrained process as the *smoothing* density in an inference setting. By considering Eq.(2) as a likelihood function, and treating the costs $U(x,t)$ and $\chi(x)$ as continuous time observations of the process X_t, the path measure $\mathbb{Q}$, i.e. the product of the *a priori* distribution $\mathbb{P}_f$ with the likelihood, can be interpreted as the posterior distribution over paths that account for the observations as soft constraints. Thus, drawing inspiration from *forward–backward* smoothing algorithms for hidden Markov models [32, 33], we factorise the marginal density into two terms that account for past and future constraints separately

$$q_t(x) \propto \rho_t(x)\phi_t(x). \tag{7}$$

The density $\rho_t(x)$ satisfies the forward *filtering* equation with initial condition $\rho_0(x)$ (Eq.(8)),

$$\frac{\partial \rho_t(x)}{\partial t} = \mathcal{L}_f \rho_t(x) - U(x,t)\rho_t(x), \tag{8}$$

while the marginal constrained density $q_t(x)$ fulfils a Fokker–Planck equation (Eq.(9))

$$\frac{\partial q_t(x)}{\partial t} = \mathcal{L}_g q_t(x). \tag{9}$$

Here the Fokker–Planck operator $\mathcal{L}_g$ is defined for the process with optimal drift $g(x,t)$ (Eq.(5)).

By replacing $\phi_t(x)$ with $q_t(x)/\rho_t(x)$ in Eq.(5), we obtain a new representation of the optimal drift in terms of the logarithmic gradients (*score functions*) of two forward probability flows, $q_t(x)$ and $\rho_t(x)$

$$g(x,t) = f(x,t) + \sigma^2 \Big(\nabla \ln q_t(x) - \nabla \ln \rho_t(x) \Big). \tag{10}$$

This formulation still requires knowledge of the unknown $\nabla \ln q_t(x)$. Yet, by inserting Eq.(10) into Eq.(9), and introducing a time-reversion through the variable $\tau = T - t$, we obtain a Fokker–Planck equation for the flow $\tilde{q}_\tau(x) = q_{T-\tau}(x)$

$$\frac{\partial \tilde{q}_\tau(x)}{\partial \tau} = -\nabla \cdot \Big(\big(\sigma^2 \nabla \ln \rho_{T-\tau}(x) - f(x, T-\tau) \big) \tilde{q}_\tau(x) \Big) + \frac{\sigma^2}{2} \nabla^2 \tilde{q}_\tau(x), \tag{11}$$

that depends only on the time-reversed forward flow $\rho_t(x)$, with $\tilde{q}_0 \propto \rho_T(x)\chi(x)$. Thus, for the exact computation of the optimal controls $u^*(x,t) = \sigma^2 \big(\nabla \ln \tilde{q}_{T-t}(x) - \nabla \ln \rho_t(x) \big)$, we require the logarithmic gradients of two forward probability flows $\tilde{q}_{T-t}(x)$ and $\rho_t(x)$.

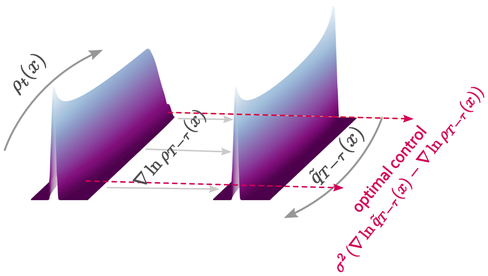

Figure 1: **Schematic of the proposed control framework.**

Deterministic particle dynamics. To sample the two forward flows $\rho_t(x)$ and $\tilde{q}_{T-t}(x)$ (Eq.(8) and Eq.(9)) we employ a recent deterministic particle framework for solving Fokker–Planck equations [34], modified to fit our purposes. We approximate $\rho_t(x)$ with the empirical distribution $\hat{\rho}_t(x) \approx \frac{1}{N} \sum_{i=1}^N \delta\left(x - X_t^{(i)}\right)$ constructed from an ensemble of N "particles" $X_t^{(i)}$.

For flows **without** path costs ($U(x,t) \equiv 0$), we express the particle dynamics as a system of ordinary differential equations (ODEs) [34]

$$dX_t^{(i)} = f(X_t^{(i)}, t)\, dt - \sigma^2/2 \hat{\nabla} \ln \hat{\rho}_t(X_t^{(i)})\, dt, \tag{12}$$

where $\hat{\nabla} \ln \hat{\rho}_t(X_t^{(i)})$ denotes the estimated score function of the empirical distribution $\hat{\rho}_t(x)$.

For flows **with** path costs ($U(x,t) \neq 0$), the flow dynamics in terms of operator exponentials[2] reads

$$\rho_{t+\delta t}(x) = e^{\delta t (\mathcal{L}_f - U(x,t))} \rho_t(x) = e^{-\delta t U(x,t)} e^{\delta t \mathcal{L}_f} \rho_t(x) + \mathcal{O}((\delta t)^2). \tag{13}$$

We interpret Eq.(13) as the concatenation of two processes: a density propagation by the uncontrolled Fokker–Planck equation, followed by a multiplication by a factor $e^{-\delta t U(x,t)}$. To simulate this two-stage process for a time interval δt, we first evolve the particles following Eq.(12) to auxiliary positions $Y_t^{(i)}$ and assign to each particle i a *weight* $\Omega_i(t) \propto e^{-\delta t U(Y_t^{(i)}, t)}$. To transform the weighted particles to unweighted ones, we employ the *ensemble transform particle filter* [35]. This method provides an optimal transport map that *deterministically* transforms an ensemble of weighted particles into an ensemble of uniformly weighted ones by maximising the covariance between the two ensembles.

Sparse kernel score function estimator. To estimate the scores of the flows $\rho_t(x)$ and $\tilde{q}_t(x)$ for the particle evolution (Eq.(12)) and the estimation of optimal controls $u^*(x,t)$, we employ a *sparse kernel score function estimator* [34]. More precisely, we obtain each dimensional component $a \in [1, \dots, d]$ of $\nabla \ln \rho(x)$ from the solution of the *variational* problem of minimising the functional $\mathcal{I}_\alpha[h, \rho]$

$$\partial_\alpha \ln \rho(x) = \arg\min_h \; \mathcal{I}_\alpha[h, \rho](x) = \arg\min_h \int \rho(x) \left(2\partial_\alpha h(x) + h^2(x) \right) dx. \tag{14}$$

To regularise this optimisation we assume that h belongs to a Reproducing Kernel Hilbert Space associated with a radial basis function kernel, and employ a *sparse kernel approximation* by expressing h as a linear combination of the kernel evaluated at $M \ll N$ *inducing points* Z_i, $h(x) = \sum_{i=1}^M b_i K(Z_i, x)$. (See AppendixA.1 for the exact formulation of the estimator.)

3 Numerical Experiments

We employed the proposed method (deterministic particle flow control-**DPF**) on a model that can be thought of as describing the *mean* phenotype (x, y) of a population evolving on a phenotypic

[2] In the second equality, we considered that for small δt the commutator of the two operators $\mathcal{L}_f$ and $U(x,t)$ is negligible.

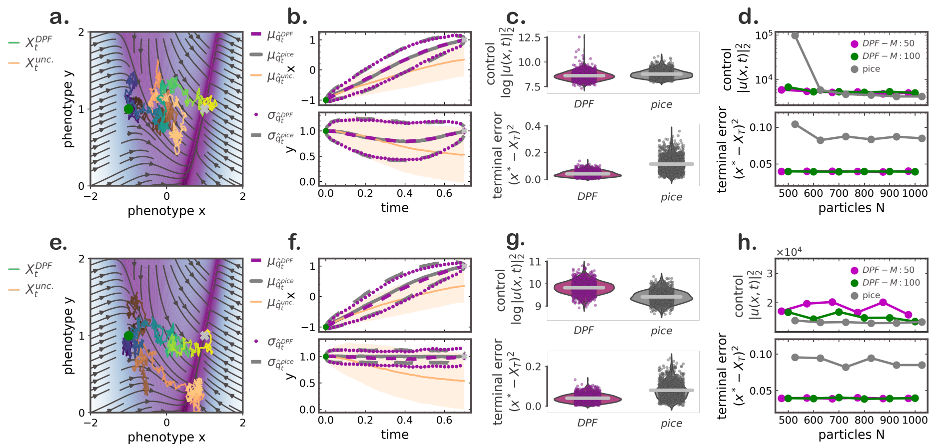

Figure 2: **Deterministic particle flow (DPF) control provides optimal interventions to drive the system to target state (grey cross). (a.)** Example controlled trajectory (blue-yellow) successfully reaches target, while an uncontrolled one (orange) remains in the vicinity of initial state for the same time interval. **(b.)** Agreement between summary statistics of marginal densities estimated from 1000 trajectories controlled by DPF (purple) and PICE (grey) (transient mean $\mu_{\hat{q}_t}$ and standard deviation $\sigma_{\hat{q}_t}$). Orange indicates mean and standard deviation of 1000 uncontrolled trajectories. (Used $N = 400$ particles for DPF, and $N = 500$ for PICE to obtain the optimal controls.) **(c.)** Comparison of **(upper)** (logarithmic) control energy, and **(lower)** deviation of terminal state from target for each controlled trajectory (dots) with interventions computed according to DPF (magenta) and PICE (grey). Light grey lines identify the mean of each quantity over the 1000 trajectories. **(d.)(upper)** Control energy, and **(lower)** terminal error for increasing particle number N. (inducing point number for DPF magenta: $M = 50$, green: $M = 100$). Grey line indicates the performance of PICE in the same setting. **(e.-h.)** Same as (a.-d.) with additional path constraint $U((x, y), t) = 10^3(y - 1)^2$.

landscape F under adaptive pressures $f([x, y], t) = \nabla F(x, y) = \nabla((1 - x)^2 + (y - x^2)^2)$ and genetic drift represented by white noise [10].(See Appendix A.3 for more biological relevance.)

Starting from initial state $\mathbf{x}_0 = (-1, 1)$, we evaluate our framework on two scenarios: one with only terminal constraints $\chi(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}^*)$, with $\mathbf{x}^* = (1, 1)$ (Figure 2(a.-d.)), and one with the same terminal constraints coupled with a path cost that limits fluctuations along the y axis (Figure 2(e.-h.)). In both settings, we benchmark the proposed approach against the path integral cross entropy method (**PICE**) [27], by comparing summary statistics, control costs $\|u(\mathbf{x}, t)\|_2^2$, and deviations from target $\|X_T - \mathbf{x}^*\|^2$ of 1000 independent trajectories controlled by each framework (purple:DPF, grey:PICE).

The proposed method successfully controlled the system towards the predefined target ($\mathbf{x}^*$-grey cross) (Figure 2 a.,e.), and showed complete agreement with PICE in terms of the transient mean and standard deviation of the marginal densities $q_t(x)$ captured by the 1000 trajectories controlled with each framework. Comparing the control effort characterising the optimality of the interventions (Figure 2 c.,g.), both methods dissipated comparable energy with DPF showing slightly larger variance among individual trajectories. Nevertheless, by examining the terminal errors, DPF was consistently more accurate and precise in reaching the target. Comparing the performance of both approaches for increasing particle number employed for obtaining the controls, DPF delivered more efficient controls from PICE for small number of particles ($N = 500$), while both methods were comparable for increasing particle number (Figure 2 d.). These results suggest that the proposed framework delivers equally optimal controls with PICE in one shot, while it is also relatively more accurate in reaching the targets.

4 Conclusions

Forward-backward algorithms for smoothing densities have been extensively used in hidden Markov models. Here by relying on the duality between inference and control [21, 36–38], we borrowed ideas from the inference literature to derive non-iterative sampling schemes for constraining diffusive systems. By employing score function estimation, and recent deterministic approaches for solving Fokker–Planck equations, we proposed a stochastic control framework that relies solely on deterministic particle dynamics. Our method interpolates between classical space-discretising PDE solutions that are inherently non-iterative, and stochastic Monte Carlo 'particle' methods that rely on the Feynaman-Kac formula to obtain PDE solutions through sample paths.

Dynamical equations for diffusion processes conditioned to reach specific terminal constraints have been already known in the field of statistical mechanics [2, 39–42] obtained from the Doob's h-transform [43]. The resulting effective drift from the h-transform is equivalent to the optimal drift obtained from the Path Integral control formalism. Yet, this series of work has been mostly employed on simple systems due the intractability of the logarithmic gradient of the backward PDE (Eq.(6)) for more general systems.

Following the PI-control formulation, a cluster of stochastic control approaches have solved the linearised HJB equation by low-rank tensor [44], or path integral approximations, namely Laplace approximation [45], Monte Carlo sampling [46], or kernel embeddings [47]. Alternative approaches have employed iterative optimisation of the controls with objectives evaluated on simulated sample paths [1, 27–29, 48, 49].

Recent interest in score-based generative modeling that employs SDEs to perturb data distributions [18] for learning generative models of a given dataset, has triggered the development of methods that may be viewed as solving an optimal control problem, or share elements with the work presented here. Song et al. in [18] employ score matching and a reverse time SDEs together with our deterministic probability flow dynamics [34] to construct generative models for images. Although this method has common ingredients with **DPF**, the two frameworks are built on different substrates (neural networks vs. kernels), and for different purposes (generative modeling vs. optimal control). In fact, in the current work we are specifically interested in extracting the optimal interventions from the logarithmic gradients of the simulated probability flows. Similarly methods that solve the Schrödinger bridge problem [16, 50, 51], developed in parallel with **DPF**, tackle the optimal transport problem of matching two arbitrary distributions between the initial and terminal times, and may be view as solving a stochastic optimal control problem with constraints the initial and target distribution [52, 53].

The major limitation in applying the proposed method more broadly in higher dimensional systems is the curse of dimensionality. The number of particles required to provide enough evidence for accurate score estimation increases with system dimension, and more advanced methods of model reduction shall be combined with the present work. A further computational bottleneck when path constraints are pertinent is the computational complexity of ensemble transform particle filter algorithm ($\mathcal{O}(N^3 \log N)$ [54]).Yet, there is room for improvement here by applying entropy regularised approaches for particle reweighting [55].

Acknowledgements

We thank the two anonymous reviewers for their suggestions and comments.

The research of Dimitra Maoutsa and Manfred Opper has been partially funded by Deutsche Forschungsgemeinschaft (DFG)-SFB1294/ 1-318763901.

References

[1] Carsten Hartmann and Christof Schütte. Efficient rare event simulation by optimal nonequilibrium forcing. *Journal of Statistical Mechanics: Theory and Experiment*, 2012(11):P11004, 2012.

[2] Raphaël Chetrite and Hugo Touchette. Variational and optimal control representations of conditioned and driven processes. *Journal of Statistical Mechanics: Theory and Experiment*, 2015(12):P12001, 2015.

[3] Jin Won Kim and Prashant G Mehta. An optimal control derivation of nonlinear smoothing equations. In *Proceedings of the Workshop on Dynamics, Optimization and Computation held in honor of the 60th birthday of Michael Dellnitz*, pages 295–311. Springer, 2020.

[4] Emanuel Todorov. Efficient computation of optimal actions. *Proceedings of the National Academy of Sciences of the United States of America*, 106(28):11478–11483, 2009.

[5] Jose Casadiego, Dimitra Maoutsa, and Marc Timme. Inferring network connectivity from event timing patterns. *Physical review letters*, 121(5):054101, 2018.

[6] Xuechen Li, Ting-Kam Leonard Wong, Ricky TQ Chen, and David Duvenaud. Scalable gradients for stochastic differential equations. In *International Conference on Artificial Intelligence and Statistics*, pages 3870–3882. PMLR, 2020.

[7] Belinda Tzen and Maxim Raginsky. Theoretical guarantees for sampling and inference in generative models with latent diffusions. In *Conference on Learning Theory*, pages 3084–3114. PMLR, 2019.

[8] Hilbert J Kappen. Linear theory for control of nonlinear stochastic systems. *Physical Review Letters*, 95(20):200201, 2005.

[9] Daniel K Wells, William L Kath, and Adilson E Motter. Control of stochastic and induced switching in biophysical networks. *Physical Review X*, 5(3):031036, 2015.

[10] Armita Nourmohammad and Ceyhun Eksin. Optimal evolutionary control for artificial selection on molecular phenotypes. *Physical Review X*, 11(1):011044, 2021.

[11] Ziwei Zhong, Brandon G Wong, Arjun Ravikumar, Garri A Arzumanyan, Ahmad S Khalil, and Chang C Liu. Automated continuous evolution of proteins in-vivo. *ACS synthetic biology*, 9(6):1270–1276, 2020.

[12] Ta-Chu Kao, Mahdieh S Sadabadi, and Guillaume Hennequin. Optimal anticipatory control as a theory of motor preparation: a thalamo-cortical circuit model. *Neuron*, 109(9):1567–1581, 2021.

[13] Alexandre Iolov, Susanne Ditlevsen, and André Longtin. Stochastic optimal control of single neuron spike trains. *Journal of Neural Engineering*, 11(4):046004, 2014.

[14] Stephen H Scott. Optimal feedback control and the neural basis of volitional motor control. *Nature Reviews Neuroscience*, 5(7):532–545, 2004.

[15] Espen Bernton, Jeremy Heng, Arnaud Doucet, and Pierre E Jacob. Schrödinger Bridge Samplers. *arXiv preprint arXiv:1912.13170*, 2019.

[16] Francisco Vargas, Pierre Thodoroff, Neil D Lawrence, and Austen Lamacraft. Solving Schrödinger Bridges via Maximum Likelihood. *arXiv preprint arXiv:2106.02081*, 2021.

[17] Ioannis Exarchos and Evangelos A Theodorou. Stochastic optimal control via forward and backward stochastic differential equations and importance sampling. *Automatica*, 87:159–165, 2018.

[18] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[19] Igor Vladimirovich Girsanov. On transforming a certain class of stochastic processes by absolutely continuous substitution of measures. *Theory of Probability and Its Applications*, 5(3):285–301, 1960.

[20] Bernt Øksendal. *Stochastic differential equations*. Springer, 2003.

[21] Hilbert J Kappen, Vicenç Gómez, and Manfred Opper. Optimal control as a graphical model inference problem. *Machine Learning*, 87(2):159–182, 2012.

[22] Richard Bellman. Dynamic programming and Lagrange multipliers. *Proceedings of the National Academy of Sciences of the United States of America*, 42(10):767, 1956.

[23] Evangelos A Theodorou and Emanuel Todorov. Relative entropy and free energy dualities: Connections to path integral and kl control. In *2012 IEEE 51st IEEE Conference on Decision and Control (cdc)*, pages 1466–1473. IEEE, 2012.

[24] Wendell H Fleming. Exit probabilities and optimal stochastic control. *Applied Mathematics and Optimization*, 4(1):329–346, 1977.

[25] Jochen Garcke and Axel Kröner. Suboptimal feedback control of PDEs by solving HJB equations on adaptive sparse grids. *Journal of Scientific Computing*, 70(1):1–28, 2017.

[26] Mario Annunziato and Alfio Borzì. A Fokker–Planck control framework for multidimensional stochastic processes. *Journal of Computational and Applied Mathematics*, 237(1):487–507, 2013.

[27] Hilbert Johan Kappen and Hans Christian Ruiz. Adaptive importance sampling for control and inference. *Journal of Statistical Physics*, 162(5):1244–1266, 2016.

[28] Wei Zhang, Han Wang, Carsten Hartmann, Marcus Weber, and Christof Schuette. Applications of the cross-entropy method to importance sampling and optimal control of diffusions. *SIAM Journal on Scientific Computing*, 36(6):A2654–A2672, 2014.

[29] Nikolas Nüsken and Lorenz Richter. Solving high-dimensional Hamilton–Jacobi–Bellman PDEs using neural networks: perspectives from the theory of controlled diffusions and measures on path space. *Partial Differential Equations and Applications*, 2(4):1–48, 2021.

[30] Nicolas Macris and Raffaele Marino. Solving non-linear Kolmogorov equations in large dimensions by using deep learning: a numerical comparison of discretization schemes. *arXiv preprint arXiv:2012.07747*, 2020.

[31] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *International Conference on Learning Representations*, 2020.

[32] Yariv Ephraim and Neri Merhav. Hidden markov processes. *IEEE Transactions on Information Theory*, 48(6):1518–1569, 2002.

[33] Leonard E Baum, Ted Petrie, George Soules, and Norman Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171, 1970.

[34] Dimitra Maoutsa, Sebastian Reich, and Manfred Opper. Interacting particle solutions of Fokker–Planck equations through gradient–log–density estimation. *Entropy*, 22(8):802, 2020.

[35] Sebastian Reich. A nonparametric ensemble transform method for Bayesian inference. *SIAM Journal on Scientific Computing*, 35(4):A2013–A2024, 2013.

[36] Emanuel Todorov. General duality between optimal control and estimation. In *2008 47th IEEE Conference on Decision and Control*, pages 4286–4292. IEEE, 2008.

[37] Sergey Levine. Reinforcement learning and control as probabilistic inference: Tutorial and review. *arXiv preprint arXiv:1805.00909*, 2018.

[38] Hagai Attias. Planning by probabilistic inference. In *International Workshop on Artificial Intelligence and Statistics*, pages 9–16. PMLR, 2003.

[39] Henri Orland. Generating transition paths by Langevin bridges. *The Journal of chemical physics*, 134(17):174114, 2011.

[40] Alain Mazzolo. Constrained Brownian processes and constrained Brownian bridges. *Journal of Statistical Mechanics: Theory and Experiment*, 2017(2):023203, 2017.

[41] Juraj Szavits-Nossan and Martin R Evans. Inequivalence of nonequilibrium path ensembles: the example of stochastic bridges. *Journal of Statistical Mechanics: Theory and Experiment*, 2015(12):P12008, 2015.

[42] Satya N Majumdar and Henri Orland. Effective Langevin equations for constrained stochastic processes. *Journal of Statistical Mechanics: Theory and Experiment*, 2015(6):P06039, 2015.

[43] JL Doob. Classical potential theory and its probabilistic counterpart: advanced problems. *Grundlehren der mathematischen Wissenschaften*, 262, 1982.

[44] Matanya B Horowitz, Anil Damle, and Joel W Burdick. Linear Hamilton Jacobi Bellman equations in high dimensions. In *53rd IEEE Conference on Decision and Control*, pages 5880–5887. IEEE, 2014.

[45] Hilbert J Kappen. Path integrals and symmetry breaking for optimal control theory. *Journal of statistical mechanics: theory and experiment*, 2005(11):P11011, 2005.

[46] Bart Van Den Broek, Wim Wiegerinck, and Bert Kappen. Graphical model inference in optimal control of stochastic multi-agent systems. *Journal of Artificial Intelligence Research*, 32:95–122, 2008.

[47] Konrad Rawlik, Marc Toussaint, and Sethu Vijayakumar. Path integral control by reproducing kernel Hilbert space embedding. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

[48] Evangelos Theodorou, Freek Stulp, Jonas Buchli, and Stefan Schaal. An iterative path integral stochastic optimal control approach for learning robotic tasks. *IFAC Proceedings Volumes*, 44(1):11594–11601, 2011.

[49] Sep Thijssen and HJ Kappen. Path integral control and state-dependent feedback. *Physical Review E*, 91(3):032104, 2015.

[50] Gefei Wang, Yuling Jiao, Qian Xu, Yang Wang, and Can Yang. Deep Generative Learning via Schr\"{o}dinger Bridge. *arXiv preprint arXiv:2106.10410*, 2021.

[51] Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion Schr\" odinger Bridge with Applications to Score-Based Generative Modeling. *arXiv preprint arXiv:2106.01357*, 2021.

[52] Michele Pavon, Giulio Trigila, and Esteban G Tabak. The Data-Driven Schrödinger Bridge. *Communications on Pure and Applied Mathematics*, 74(7):1545–1573, 2021.

[53] Kenneth Caluya and Abhishek Halder. Wasserstein Proximal Algorithms for the Schr\"{o} dinger Bridge Problem: Density Control with Nonlinear Drift. *IEEE Transactions on Automatic Control*, 2021.

[54] Dimitris Bertsimas and John N Tsitsiklis. *Introduction to linear optimization*, volume 6. Athena Scientific Belmont, MA, 1997.

[55] Adrien Corenflos, James Thornton, George Deligiannidis, and Arnaud Doucet. Differentiable particle filtering via entropy-regularized optimal transport. In *International Conference on Machine Learning*, pages 2100–2111. PMLR, 2021.

[56] Ofir Pele and Michael Werman. Fast and robust earth mover's distances. In *2009 IEEE 12th International Conference on Computer Vision*, pages 460–467. IEEE, September 2009.

[57] Michael L Waskom. Seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021.

[58] William B Provine. *Sewall Wright and evolutionary biology*. University of Chicago Press, 1989.

[59] N.H. Barton, D.E. Briggs, J.A. Eisen, D.B. Goldstein, and N.H. Patel. *Evolution*. Cold Spring Harbor Laboratory Series. Cold Spring Harbor Laboratory Press, 2007.

[60] Michael C Whitlock. Variance-induced peak shifts. *Evolution*, 49(2):252–259, 1995.

[61] Joel G Kingsolver, Hopi E Hoekstra, Jon M Hoekstra, David Berrigan, Sacha N Vignieri, CE Hill, Anhthu Hoang, Patricia Gibert, and Peter Beerli. The strength of phenotypic selection in natural populations. *The American Naturalist*, 157(3):245–261, 2001.

[62] Armita Nourmohammad, Stephan Schiffels, and Michael Lässig. Evolution of molecular phenotypes under stabilizing selection. *Journal of Statistical Mechanics: Theory and Experiment*, 2013(01):P01012, 2013.

[63] Torsten Held, Armita Nourmohammad, and Michael Lässig. Adaptive evolution of molecular phenotypes. *Journal of Statistical Mechanics: Theory and Experiment*, 2014(9):P09029, 2014.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] Although we do not foresee direct societal impact, since stochastic control algorithms may be used for malicious military and financial engineering purposes, we acknowledge this possibility
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] The main assumptions of our approach are that the dynamics of the system under consideration may be captured by an SDE which we consider *known*, and that the control costs arise in a quadratic form in the cost functional, i.e. the cost functional relevant for the system under consideration has the form of Eq. (3).

(b) Did you include complete proofs of all theoretical results? [Yes] The main results are Eq.(11) and evolution of the probability flows with deterministic particle dynamics. We indicate in the main text how to arrive to Eq.(11), and in the paragraph **Deterministic particle dynamics** we describe our approach for representing the probability flows $\rho_t(x)$ and $\tilde{q}_t(x)$ with deterministic particle dynamics.

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We will provide the code upon acceptance. Nevertheless in the main text we give all the parameters used for the numerical experiments (mainly in the Figure 2 caption), and additionally provide in the appendix the exact formulation of the score function estimator (see Appendix A.1) and a pseudo-code outlining the main framework (see Appendix A.2).
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] The "training" in our setting is the evaluation of the kernels. We provide in the appendix details on the kernels, hyperparameters, and precise form of the score estimator.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] In Figure 2.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] We didn't use any GPU, cluster etc. Only a plain personal laptop.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] In the Appendix, we do cite FastEMD, employed for solving the optimal transport problem in the ensemble transform particle filter, and Seaborn used for data visualisation. We benchmarked our method against the Path Integral Cross Entropy framework (PICE) that we implemented alone and cite in the main text.
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Appendix

A.1 Score function estimator

The empirical formulation of the score estimator from N particles representing an unknown density $\rho(x)$ is

$$\partial_\alpha \ln \rho(x) \approx \sum_{i=1}^{M} \left(\sum_{k=1}^{M} B_{ik}(x) \sum_{l=1}^{N} \nabla_{X_l} K(X_l, Z_k) \right), \tag{15}$$

with B_{ik} denoting the i-th row, and k-th column of the matrix $B(x)$ defined as

$$B(x) \doteq K(x, \mathcal{Z}) \left[\lambda I + (K(\mathcal{Z}, \mathcal{Z}))^{-1} (K(\mathcal{X}, \mathcal{Z}))^\top (K(\mathcal{X}, \mathcal{Z})) \right]^{-1} (K(\mathcal{Z}, \mathcal{Z}))^{-1}, \tag{16}$$

where $\mathcal{X} = \{X_i\}_{i=1}^N$ and $\mathcal{Z} = \{Z_i\}_{i=1}^M$ denote the sets of samples and inducing points respectively, while I stands for an $M \times M$ identity matrix. (Here we set the regularising constant $\lambda = 10^{-3}$). We used an gaussian kernel

$$K(x, x') = \exp\left[-\frac{1}{2l^2}\|x - x'\|^2\right], \tag{17}$$

where the lengthscale l was set to two times the standard deviation of the particle ensemble for each time step.

A.2 Implementation details

Here we provide the algorithm for computing optimal interventions $u^*(x,t)$. Since the initial conditions for the flows $\rho_t(x)$ and $\tilde{q}_t(x)$ are delta functions centered around the initial and target state, x_0 and x_1, i.e $\rho_0(x) = \delta(x - x_0)$ and $\tilde{q}_t(x) = \delta(x - x_1)$, we employ a single stochastic step at the beginning of each (forward and time-reversed) flow propagation. Since the inducing point number M employed in the gradient–log–density estimation is considerably smaller than sample number N, i.e., $M \ll N$, the overall computational complexity of a *single* gradient-log-density evaluation amounts to $\mathcal{O}\left(N\, M^2\right)$. We perform Euler integration for the ODEs, and Euler-Maruyama for stochastic simulations. For all numerical integrations we employ $dt = 10^{-3}$ discretisation step.

Algorithm A1: Deterministic Particle Flow (DPF) control

Input: N, M: scalars, number of particles and number of inducing points
t_0, t_1, dt: scalars, initial and final timepoints, and discretisation step
x_0, x_1: $1 \times d$, $1 \times d$ initial and target state
f: drift function
σ: noise amplitude
$U(x,t)$: function, path constraint (optional)

Output: Z, B: $d \times N \times (t_1 - t_0)/dt$, samples from forward flows $\rho_t(x)$ and $q_t(x)$
$u^*(x,t)$: functions from $\mathbb{R}^d \to \mathbb{R}^d$ for each $(t_1 - t0)/dt$ time step, time- and state-dependent controls

$k = (t_1 - t_0)/dt$ `// number of timesteps`
`// Forward propagation of flow` $\rho_t(x)$
$Z_{ti=0} = x_0$ `// initialise particles' positions`
$Z_{ti=1} = Z_0 + dt f(Z_0, t_0) + \sigma\mathcal{N}(0, \sqrt{dt})$ `// 1st step is stochastic`
For $ti = 2 : k$ `// deterministic propagation`
  $Z_{ti+1} = Z_{ti} + dt\left(f(Z_{ti}, t) - \frac{1}{2}\sigma^2\nabla \log\rho(Z_{ti})\right)$
  If $\exists$ path cost $U(x,t)$:
    $W = \exp\left(-U(Z_{ti+1}, t)\, dt\right)$
    $T^* =$ EnsembleTransformParticleFilter(Z_{ti+1}, W)
    $Z_{ti+1} = Z_{ti+1} \cdot T^*$
`// Time-reversed propagation of flow` $q_t(x)$
$B_{ti=k} = x_1$ `// initialise particles' positions`
`// 1st step is stochastic`
$B_{ti=k-1} = B_k - dt f(B_k, t_1) + dt\sigma^2\nabla\log\rho(Z_k) + \sigma\mathcal{N}(0, \sqrt{dt})$
For $ti = k - 2 : 0$ `// deterministic propagation`
  $B_{ti-1} = B_{ti} - dt f(B_{ti}, t) + dt\sigma^2\nabla\log\rho(Z_{ti}) + dt\frac{1}{2}\sigma^2\nabla\log q(B_{ti})$
`// Compute` $u^*(x\, t)$
For $ti = 2 : k$
  $u^*(x, ti) = \sigma^2\nabla\log q(B_{ti}) - \sigma^2\nabla\log\rho(Z_{ti})$

For the numerical experiments with path constraints, we solved the optimal transport problem with the implementation of FastEMD [56].

For some of the visualisations of our results we used the Seaborn [57] python toolbox.

A.3 Controlling evolving populations

For an evolving population, the main evolutionary drivers comprise fitness and mutation forces that continuously adjust the composition of phenotypes within the population, while genetic drift perturbs the whole process stochastically. We describe the evolution of the *mean phenotypes* $\mathbf{x} := (x, y)$ of the population by the overdamped Langevin equation

$$d\mathbf{x} = C \cdot \nabla F(\mathbf{x})dt + \sigma dW, \tag{18}$$

with $F(x)$ denoting the phenotypic landscape in the presence of natural selection [58], where the landscape axes represent different phenotypic traits, and σ the noise amplitude that rescales the genetic drift, i.e. the stochastic term, according to the population size n and the covariance matrix C, $\sigma = C^{1/2} n^{-1}$. The gradient of the landscape $f(x) = C \cdot \nabla F(x)$ captures the adaptive pressures under natural selection.

Here, we assume an asymmetric rugged landscape [59], that may arise in small sized populations with small variance, and multi-modal individual fitness functions [60], supported also by empirical findings indicating asymmetry in selection landscapes [61]. In this setting, optimal control can be thought of as artificial selection that diverts evolving populations from strictly following natural selection, and drives them to externally imposed target states [10]. Path constraints are essential to prevent changes of co-varying phenotypic traits. Artificial selection to promote the prevalence of a phenotypic trait in a population may lead to undesired variations along covarying traits. Therefore, path constraints may be employed to reduce the fluctuations along covarying, but not targeted traits. For simplicity we consider the covariance matrix $C = I$ constant, given the much smaller timescales upon which its fluctuations unfold [62], and its weak dependency on the evolutionary selection strength [63].

The dynamics of Eq.(18) describe the evolution of populations in the presence of natural selection towards an evolutionary optimum, captured by the maximum of the adaptive landscape, adhering thereby to physiological and environmental constraints.

To study and understand the outcomes and dynamics of adaptive evolution, we need to devise intervention protocols that drive phenotypes towards non-evolutionary optimum states $\mathbf{x}^*$, or through evolutionary trajectories that deviate from the gradient of the phenotypic landscape. This intervention is implemented through artificial selection, which, following [10], we formulate here as a time- and state- dependent perturbation $\mathbf{u}(x, t)$ to the natural selection

$$d\mathbf{x} = \Big(f(\mathbf{x}) + u(\mathbf{x}, t)\Big)dt + \sigma d\mathbf{W}. \tag{19}$$