A. Existence of Unified Flow Operator

Theorem 2.3. For the optimal control problem in Eq. (13) and Eq. (14), there exists an open-loop control \( w^* = w^*(q(x,0), t) \) such that the induced state \( q^*(x, t) \) satisfies \( q^*(x, \infty) = p(x|O_{m+1}) \). Moreover, \( w^* \) has a fixed expression with respect to \( p(x|O_m) \) and \( p(o_{m+1}|x) \) across different \( m \).

Proof. By Theorem 2.2, \( \dot{w}^*(q(x, t), t) := \nabla_x \log q(x, t) \) can induce the optimal state \( \tilde{q}^*(x, \infty) = p(x|O_{m+1}) \) and achieve a zero loss, \( d = 0 \). Hence, \( \dot{w}^* \) is an optimal closed-loop control for this problem.

Although in general closed-loop control has a stronger characterization to the solution, in a deterministic system like Eq. (14), the optimal closed-loop control and the optimal open-loop control will give the same control law and thus the same optimality to the loss function (Dreyfus, 1964). Hence, there exists an optimal open-loop control \( w^* = w^*(q(x, 0), t) \) such that the induced optimal state also gives a zero loss and thus \( q^*(x, \infty) = p(x|O_{m+1}) \).

More specifically, when the system is deterministic, a state \( q(x, t) \) is just a deterministic result of the initial state \( q(x, 0) \) and the dynamics. The optimal flow determined by \( \dot{w}^*(q(x, t), t) \) is
\[
f = \nabla_x \log p(x|O_m)p(o_{m+1}|x) - \nabla_x \log q(x, t).
\]

The continuity equation gives
\[
\frac{\partial q(x, t)}{\partial t} = - \nabla_x \cdot (q \nabla_x \log p(x|O_m)p(o_{m+1}|x)) + \Delta_x q(x, t)
\]
\[
: = g(p(x|O_m)p(o_{m+1}|x), q(x, t))
\]

Hence, for any \( x \),
\[
q(x, t) = q(x, 0) + \int_0^t g(p(x|O_m)p(o_{m+1}|x), q(x, t)) \, d\tau.
\]

The dynamics \( g \) is a fixed function of \( p(x|O_m), p(o_{m+1}|x) \) and \( q(x, t) \), so the solution of this initial value problem (IVP) \( q(x, t) \) is a fixed function of \( p(x|O_m), p(o_{m+1}|x), q(x, 0) \) and \( t \), which can be written as
\[
q(x, t) = \text{Solve-IVP}(p(x|O_m), p(o_{m+1}|x), q(x, 0), t).
\]

Finally, we can write the optimal open-loop control as
\[
w^*(q(x, 0), t)
\]
\[
= \nabla_x \log(\text{Solve-IVP}(p(x|O_m), p(o_{m+1}|x), q(x, 0), t)).
\]

Hence, \( w^*(q(x, 0), t) \) has a fixed form across different \( m \).

\[\square\]

B. Adjoint Method

To explain it more clearly, let us denote the evolution of the \( n \)-th particles at the \( m \)-th stage by \( x^n_m(T) \) for \( t \in [0, T] \). Note that \( x^n_m(T) = x^n_{m+1}(0) \). (Then the notation \( x^n_m \) in the main text will become \( x^n_m(T) \).)

Recall the loss for each task:
\[
L(T) = \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N (\log q^n_m(x^n_m(T), T) - \log p(x^n_m(T), O_m)) .
\]

The loss of one particle \( x^n \) is
\[
\ell^n := \frac{1}{M} \sum_{m=1}^M L^n_m,
\]
where
\[
L^n_m := - y^n_m(T) - \log p(x^n_m(T), O_m)
\]
Particle Flow Bayes’ Rule


and $y_m^n(t) := -\log q_m^n(x_m^n(t), t)$.

First, an adjoint process is defined as

$$p_m(t) := \frac{\partial L^n}{\partial [x_m^n(t), y_m^n(t)]}.$$  

Denote $f_m(x(t), \theta) = f_0(X_m, o_{m+1}, x(t), t)$. During the $m$-th stage, the adjoint process follows the following differential equation

$$\frac{dp_m}{dt} = -\frac{\partial}{\partial [x_m^n(t), y_m^n(t)]} \left[ f_m(x_m^n(t), \theta) \nabla \cdot f_m(x_m^n(t), \theta) \right]^\top p_m(t).$$  

(21)

Note that

$$p_m(T) = \sum_{m' \geq m} \frac{1}{M} \frac{\partial L_{m'}}{\partial [x_{m'}(T), y_{m'}(T)]}.$$  

(22)

Claim: The gradient of the loss is the solution of a backward ODE. That is to say, $\frac{\partial L^n}{\partial \theta} = z_1(0)$, if $z_M(T) = 0$ and

$$\frac{dz_m(t)}{dt} = \left[ \frac{\partial f_m(x_m^n(t), \theta)}{\partial \theta} \left( \nabla \cdot f_m(x_m^n(t), \theta) \right) \right]^\top p_m(t),$$  

(23)

and $z_m(T) = z_{m+1}(0)$, for $m = 0, \cdots, M - 1$.

**Proof.** First, we can compute $\frac{d}{dt} \frac{\partial L^n}{\partial \theta}$:

$$\frac{d}{dt} \frac{\partial L^n}{\partial \theta} = \frac{\partial}{\partial \theta} \sum_{m=1}^{M} \left( \frac{\partial L^n}{\partial x_m^n(t)} \frac{\partial x_m^n(t)}{dt} + \frac{\partial L^n}{\partial y_m^n(t)} \frac{\partial y_m^n(t)}{dt} \right)$$

$$= \frac{\partial}{\partial \theta} \sum_{m=1}^{M} \left[ p_m(t)^\top \left[ f_m(x_m^n(t), \theta) \nabla \cdot f_m(x_m^n(t), \theta) \right] \right]$$

$$= \sum_{m=1}^{M} \left[ \frac{\partial f_m(x_m^n(t), \theta)}{\partial \theta} \left( \nabla \cdot f_m(x_m^n(t), \theta) \right) \right]^\top p_m(t)$$

Next, we have

$$0 - \frac{\partial L^n}{\partial \theta} = -\int_{t=0}^{T} \frac{d}{dt} \frac{\partial L^n}{\partial \theta} = \sum_{m=1}^{M} \int_{t=0}^{T} -\left[ \frac{\partial f_m(x_m^n(t), \theta)}{\partial \theta} \left( \nabla \cdot f_m(x_m^n(t), \theta) \right) \right]^\top p_m(t) = z_M(T) - z_1(0).$$

Hence, $\frac{\partial L^n}{\partial \theta} = z_1(0)$ if $z_M(T) = 0$.

An algorithm for computing $\frac{\partial L}{\partial \theta}$ is summarized in Algorithm 2. A nice python package of realizing this algorithm is provided by Chen et al. (2018).
Algorithm 2 Adjoint Method of Computing the Gradient

Function $\text{Grad}(\theta, X_0, p(a|x), O_M)$:

Denote $f^{\theta}_M = f_0(X_m, o_{m+1}, x(t), t)$
Set $y^0_0 = -\log p(x^0_0)$ for each $x^0_0 \in X_0$

For all $n = 1$ to $N$

For all $m = 0$ to $M - 1$

Set $p_{m+1} = 0$ and $z_{m+1}^0 = 0$

Solve ODEs in Eq. (4), Eq. (8), Eq. (21) and Eq. (23) for $x^0_m(t), p^0_m(t)$ and $z^0_m(t)$ backwardly from $T$ to $0$

Set $x^0_m = x^0_m(0), p^0_m = p^0_m(0)$ and $z^0_m = z^0_m(0)$

Return $\frac{1}{N} \sum_{n=1}^{N} \frac{\partial L^m}{\partial \theta} = \frac{1}{N} \sum_{n=1}^{N} z^T(0)$

C. Experiment Details

C.1. Parameterization

Overall we parameterize the flow velocity as

$$f = h \left( \frac{1}{N} \sum_{n=1}^{N} \phi(x^m_n, o_{m+1}, x(t), t) \right),$$

where both $\phi$ and $h$ are neural networks. For instance, let $\text{ctx} = [\frac{1}{N} \sum_{n=1}^{N} \phi(x^m_n)^T, o_{m+1}^T]$ be the context of this conditional flow, where $\phi$ is a dense feed-forward neural network, a specific neural architecture we use in the experiment is

$$f = \text{Gated}_k \left[ \cdot \cdot \cdot [\text{ctx}, \text{Gated}_2 ([\text{ctx}, \text{Gated}_1 ([\text{ctx}, x(t)^T]^T, t)]^T, t)]^T \cdot \cdot \cdot , t \right],$$

where $\text{Gated}_j(y, t) = (W_j y + b_j) * \sigma(t v_j + c_j) + t c_j,$

where $\ast$ is element-wise multiplication. The number of layers $k$ can be tuned, but in general $h$ is a shallow network.

C.2. Evaluation Metric

MMD$^2$ The maximum mean discrepancy (MMD) of the true posterior $p$ and the estimated posterior $q$ is defined as

$$\text{MMD}^2[F, p, q] := \sup_{f \in F} (\mathbb{E}_{x \sim p}[f(x)] - \mathbb{E}_{y \sim q}[f(y)]).$$

When $F$ is a unit ball in a characteristic RKHS, Gretton et al. (2012) showed that the squared MMD is

$$\text{MMD}^2[F, p, q] = E[k(x, x')] - 2E[k(x, y)] + E[k(y, y')],$$

where $x, x' \sim p$ and $y, y' \sim q$.

Cross-entropy Evaluating the KL divergence is equivalent to evaluating the cross-entropy.

$$\mathbb{E}_{x \sim p} - \log q(x) \approx \frac{1}{n} \sum_{n=1}^{N} (-\log q(x^n)),$$

where $q(x)$ is approximated by kernel density estimation on the set of particles obtained from different sampling methods.

Integral Evaluation When the true posterior is a Gaussian distribution $N(\mu, \Sigma)$, the expectation of the following test functions have closed-form expressions.

- $\mathbb{E}[x] = \mu$
- $\mathbb{E}[x^T A x] = tr(A \Sigma) + \mu^T A \mu$
- $\mathbb{E}[(A x + a)^T (B x + b)] = tr(A \Sigma B^T) + (A \mu + a)^T (B \mu + b)$
D. More Experimental Results

D.1. Multivariate Guassian Model

![Experimental results on 2 dimensional multivariate Gaussian model.](image1)

(a) cross-entropy  (b) MMD\(^2\) with RBF kernel  (c) Integral estimation

Figure 7: Experimental results on 2 dimensional multivariate Gaussian model.

D.2. LDS Model

![Experimental results on LDS model.](image2)

(a) MMD\(^2\) with Laplacian kernel  (b) MMD\(^2\) with Polynomial kernel

(c) MMD\(^2\) with Sigmoid kernel  (d) Integral estimation on \(h(\mathbf{x}) = \mathbf{x}\)

(e) Integral estimation on \(h(\mathbf{x}) = (A\mathbf{x} + \mathbf{a})^\top (B\mathbf{x} + \mathbf{b})\)

Figure 8: Experimental results on LDS model.