## 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 $w^* = w^*(q(\boldsymbol{x},0),t)$ such that the induced state $q^*(\boldsymbol{x},t)$ satisfies $q^*(\boldsymbol{x},\infty) = p(\boldsymbol{x}|\mathcal{O}_{m+1})$. Moreover, $w^*$ has a fixed expression with respect to $p(\boldsymbol{x}|\mathcal{O}_m)$ and $p(o_{m+1}|\boldsymbol{x})$ across different $m$.*

*Proof.* By Theorem 2.2, $\tilde{w}^*(q(\boldsymbol{x},t),t) := \nabla_x \log q(\boldsymbol{x},t)$ can induce the optimal state $\tilde{q}^*(\boldsymbol{x},\infty) = p(\boldsymbol{x}|\mathcal{O}_{m+1})$ and achieve a zero loss, $d = 0$. Hence, $\tilde{w}^*$ 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 $w^* = w^*(q(\boldsymbol{x},0),t)$ such that the induced optimal state also gives a zero loss and thus $q^*(\boldsymbol{x},\infty) = p(\boldsymbol{x}|\mathcal{O}_{m+1})$.

More specifically, when the system is deterministic, a state $q(\boldsymbol{x},t)$ is just a deterministic result of the initial state $q(\boldsymbol{x},0)$ and the dynamics. The optimal flow determined by $\tilde{w}^*(q(\boldsymbol{x},t),t)$ is

$$f = \nabla_x \log p(\boldsymbol{x}|\mathcal{O}_m)p(o_{m+1}|\boldsymbol{x}) - \nabla_x \log q(\boldsymbol{x},t).$$

The continuity equation gives

$$\begin{aligned}\frac{\partial q(\boldsymbol{x},t)}{\partial t} =& - \nabla_x \cdot (q \nabla_x \log p(\boldsymbol{x}|\mathcal{O}_m)p(o_{m+1}|\boldsymbol{x})) \\ &+ \Delta_x q(\boldsymbol{x},t) \\ &:= g(p(\boldsymbol{x}|\mathcal{O}_m)p(o_{m+1}|\boldsymbol{x}), q(\boldsymbol{x},t))\end{aligned}$$

Hence, for any $\boldsymbol{x}$,

$$q(\boldsymbol{x},t) = q(\boldsymbol{x},0) + \int_0^t g(p(\boldsymbol{x}|\mathcal{O}_m)p(o_{m+1}|\boldsymbol{x}), q(\boldsymbol{x},t))\, d\tau.$$

The dynamcis $g$ is a fixed function of $p(\boldsymbol{x}|\mathcal{O}_m)$, $p(o_{m+1}|\boldsymbol{x})$ and $q(\boldsymbol{x},t)$, so the solution of this initial value problem(IVP) $q(\boldsymbol{x},t)$ is a fixed function of $p(\boldsymbol{x}|\mathcal{O}_m)$, $p(o_{m+1}|\boldsymbol{x})$, $q(\boldsymbol{x},0)$ and $t$, which can be written as

$$q(\boldsymbol{x},t) = \text{Solve-IVP}(p(\boldsymbol{x}|\mathcal{O}_m), p(o_{m+1}|\boldsymbol{x}), q(\boldsymbol{x},0), t).$$

Finally, we can write the optimal open-loop control as

$$\begin{aligned}&w^*(q(\boldsymbol{x},0),t) \\ =&\nabla_x \log(\text{Solve-IVP}(p(\boldsymbol{x}|\mathcal{O}_m), p(o_{m+1}|\boldsymbol{x}), q(\boldsymbol{x},0), t)).\end{aligned}$$

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

□

## B. Adjoint Method

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

Recall the loss for each task:

$$\mathcal{L}(\mathcal{T}) = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}\left(\log q_m^n(\boldsymbol{x}_m^n(T),T) - \log p(\boldsymbol{x}_m^n(T),\mathcal{O}_m)\right).$$

The loss of one particle $\boldsymbol{x}^n$ is

$$L^n := \frac{1}{M}\sum_{m=1}^{M} L_m^n,$$

where

$$L_m^n := -y_m^n(T) - \log p(\boldsymbol{x}_m^n(T),\mathcal{O}_m)$$

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

First, an adjoint process is defined as

$$\boldsymbol{p}_m(t) := \frac{\partial L^n}{\partial[\boldsymbol{x}_m^n(t), y_m^n(t)]}.$$

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

$$\frac{d\boldsymbol{p}_m}{dt} = -\frac{\partial}{\partial[\boldsymbol{x}_m^n(t), y_m^n(t)]} \begin{bmatrix} f_m(\boldsymbol{x}_m^n(t), \theta) \\ \nabla_{\boldsymbol{x}} \cdot f_m(\boldsymbol{x}_m^n(t), \theta) \end{bmatrix}^\top \boldsymbol{p}_m(t). \tag{21}$$

Note that

$$\boldsymbol{p}_m(T) = \sum_{m' \geq m} \frac{1}{M} \frac{\partial L_{m'}^n}{\partial[\boldsymbol{x}_m^n(T), y_m^n(T)]}. \tag{22}$$

Claim: The gradient of the loss is the solution of a backward ODE. That is to say, $\frac{\partial L^n}{\partial \theta} = \boldsymbol{z}_1(0)$, if $\boldsymbol{z}_M(T) = \mathbf{0}$ and

$$\frac{d\boldsymbol{z}_m(t)}{dt} = -\begin{bmatrix} \frac{\partial f_m}{\partial \theta}(\boldsymbol{x}_m^n(t), \theta) \\ \frac{\partial}{\partial \theta}\left[\nabla_{\boldsymbol{x}} \cdot f_m(\boldsymbol{x}_m^n(t), \theta)\right] \end{bmatrix}^\top \boldsymbol{p}_m(t), \tag{23}$$

and $\boldsymbol{z}_m(T) = \boldsymbol{z}_{m+1}(0)$, for $m = 0, \cdots, M-1$.

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

$$\begin{aligned}
\frac{d}{dt}\frac{\partial L^n}{\partial \theta} &= \frac{\partial}{\partial \theta} \sum_{m=1}^{M} \left( \frac{\partial L^n}{\partial \boldsymbol{x}_m^n(t)}^\top \frac{d\boldsymbol{x}_m^n(t)}{dt} + \frac{\partial L^n}{\partial y_m^n(t)} \frac{dy_m^n(t)}{dt} \right) \\
&= \frac{\partial}{\partial \theta} \sum_{m=1}^{M} \left[ \boldsymbol{p}_m(t)^\top \begin{bmatrix} f_m(\boldsymbol{x}_m^n(t), \theta) \\ \nabla_{\boldsymbol{x}} \cdot f_m(\boldsymbol{x}_m^n(t), \theta) \end{bmatrix} \right] \\
&= \sum_{m=1}^{M} \begin{bmatrix} \frac{\partial f_m}{\partial \theta}(\boldsymbol{x}_m^n(t), \theta) \\ \frac{\partial}{\partial \theta}\left[\nabla_{\boldsymbol{x}} \cdot f_m(\boldsymbol{x}_m^n(t), \theta)\right] \end{bmatrix}^\top \boldsymbol{p}_m(t)
\end{aligned}$$

Next, we have

$$0 - \frac{\partial L^n}{\partial \theta} = -\int_{t=0}^{T} \frac{d}{dt}\frac{\partial L^n}{\partial \theta} = \sum_{m=1}^{M} \int_{t=0}^{T} - \begin{bmatrix} \frac{\partial f_m}{\partial \theta}(\boldsymbol{x}_m^n(t), \theta) \\ \frac{\partial}{\partial \theta}\left[\nabla_{\boldsymbol{x}} \cdot f_m(\boldsymbol{x}_m^n(t), \theta)\right] \end{bmatrix}^\top \boldsymbol{p}_m(t) = \boldsymbol{z}_M(T) - \boldsymbol{z}_1(0).$$

Hence, $\frac{\partial L^n}{\partial \theta} = \boldsymbol{z}_1(0)$ if $\boldsymbol{z}_M(T) = \mathbf{0}$. $\square$

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** `Grad`$(\theta, \mathcal{X}_0, p(o|\boldsymbol{x}), \mathcal{O}_M)$**:**
- Denote $f_\theta^m = f_\theta(\mathcal{X}_m, o_{m+1}, \boldsymbol{x}(t), t)$ ▷ notation
- Set $y_0^n = -\log p(\boldsymbol{x}_0^n)$ for each $\boldsymbol{x}_0^n \in \mathcal{X}_0$
- **For all** $n = 1$ *to* $N$ **do**
  - **For** $m = 0$ *to* $M-1$ **do**
    - $\begin{bmatrix} \boldsymbol{x}_{m+1}^n \\ y_{m+1}^n \end{bmatrix} \leftarrow \begin{bmatrix} \boldsymbol{x}_m^n \\ y_m^n \end{bmatrix} + \int_0^T \begin{bmatrix} f_\theta^m \\ \nabla \cdot f_\theta^m \end{bmatrix} dt$
  - Set $\boldsymbol{p}_M^n(T) = \mathbf{0}$ and $\boldsymbol{z}_M^n(T) = \mathbf{0}$
  - **For** $m = M$ *to* $1$ **do**
    - $\boldsymbol{p}_m^n(T) \leftarrow \boldsymbol{p}_m^n(T) + \frac{1}{M} \frac{\partial L_m^n}{\partial [\boldsymbol{x}_m^n, y_m^n]}$
    - Solve ODEs in Eq. (4), Eq. (8), Eq. (21) and Eq. (23) for $\boldsymbol{x}_m^n(t)$, $\boldsymbol{p}_m^n(t)$ and $\boldsymbol{z}_m^n(t)$ backwardly from $T$ to 0
    - Set $\boldsymbol{x}_{m-1}^n(T) = \boldsymbol{x}_m^n(0)$, $\boldsymbol{p}_{m-1}^n(T) = \boldsymbol{p}_m^n(0)$ and $\boldsymbol{z}_{m-1}^n(T) = \boldsymbol{z}_m^n(0)$
- **return** $\frac{1}{N}\sum_{n=1}^N \frac{\partial L^n}{\partial \theta} = \frac{1}{N}\sum_{n=1}^N \boldsymbol{z}_1^n(0)$

## C. Experiment Details

### C.1. Evaluation Metric

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

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

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

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

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

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

$$\mathbb{E}_{x\sim p} - \log q(x) \approx \frac{1}{n}\sum_{n=1}^{N}(-\log q(x^n)), \tag{24}$$

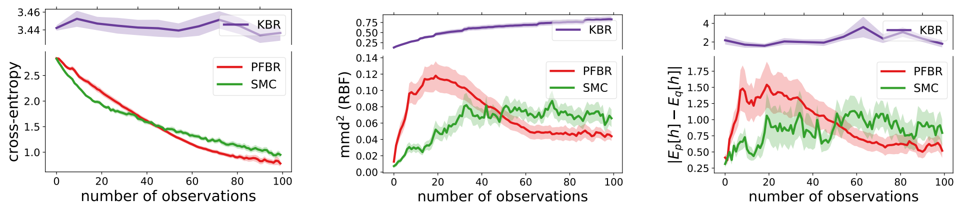
where $q(x)$ is approximated by kernel density estimation on the set of particles obtained from different sampling methods.

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

- $\mathbb{E}[\boldsymbol{x}] = \mu$
- $\mathbb{E}[\boldsymbol{x}^\top A \boldsymbol{x}] = tr(A\Sigma) + \mu^\top A \mu$
- $\mathbb{E}[(A\boldsymbol{x} + \boldsymbol{a})^\top (B\boldsymbol{x} + \boldsymbol{b})] = tr(A\Sigma B^\top) + (A\mu + \boldsymbol{a})^\top (B\mu + \boldsymbol{b})$

# D. More Experimental Results

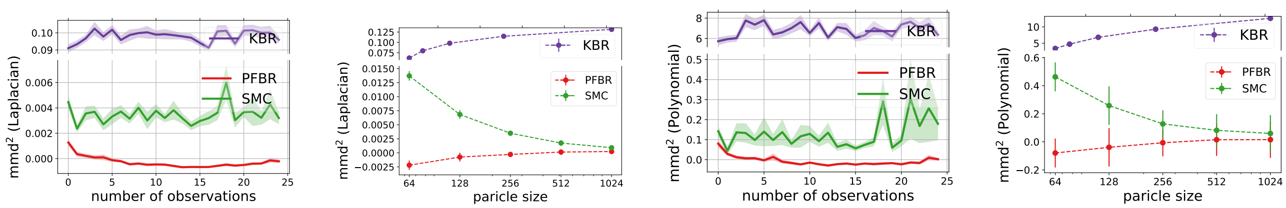
## D.1. Multivariate Guassian Model



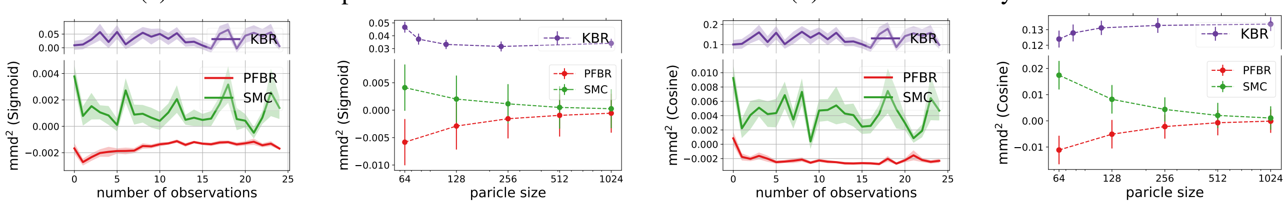

(a) cross-entropy (b) MMD$^2$ with RBF kernel (c) Integral estimation

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

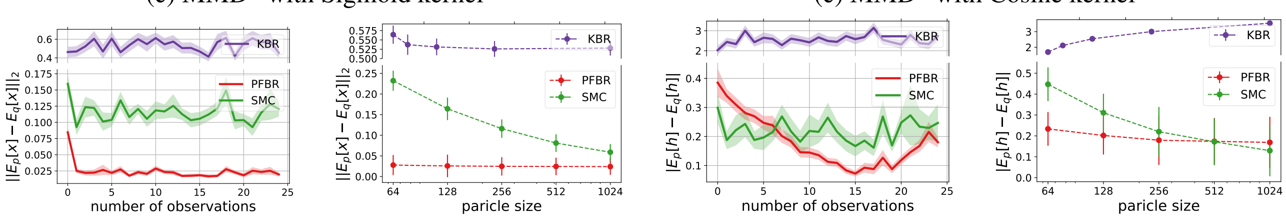
## D.2. LDS Model




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




(c) MMD$^2$ with Sigmoid kernel (c) MMD$^2$ with Cosine kernel




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

Figure 8: Experimental results on LDS model.