An Optimal Transport Formulation of the Bayes Law for Nonlinear Filtering Algorithms

Presented at
IEEE Conference on Decision and Control, Cancun, Mexico, Dec. 2022

Amirhossein Taghvaei

Joint work with Bamdad Hosseini

University of Washington, Seattle

Dec 9, 2022

Optimal transport formulation of the Bayes law

Bayes law: $$P(X|Y) = \frac{P(X)P(Y|X)}{P(Y)}$$

$$= \nabla_x \bar{f}(\cdot;Y)\#P_X$$

where $$\bar{f} = \underset{f \in L^1(\mathcal{X}\times\mathcal{Y})}{\arg\min} \ \mathbb{E}_{(X,Y)\sim P_X \otimes P_Y}[f(X;Y)] + \mathbb{E}_{(X,Y)\sim P_{XY}}[f^\star(X;Y)]$$

Computational properties:

- Only requires samples $(X_i, Y_i) \sim P_{XY}$ (data-driven/simulation based)
- Enables construction of "approximate" posterior distributions
- Allows application of ML tools (stochastic optimization and neural nets)

Optimal transport formulation of the Bayes law

$$\textbf{Bayes law:} \quad P(X|Y) = \frac{P(X)P(Y|X)}{P(Y)}$$

$$= \nabla_x \bar{f}(\cdot;Y)\#P_X$$

where $\bar{f} = \underset{f \in L^1(\mathcal{X}\times\mathcal{Y})}{\arg\min}\ \mathbb{E}_{(X,Y)\sim P_X \otimes P_Y}[f(X;Y)] + \mathbb{E}_{(X,Y)\sim P_{XY}}[f^\star(X;Y)]$

Computational properties:

- Only requires samples $(X_i, Y_i) \sim P_{XY}$ (data-driven/simulation based)
- Enables construction of "approximate" posterior distributions
- Allows application of ML tools (stochastic optimization and neural nets)

Optimal transport formulation of the Bayes law

Bayes law: $$P(X|Y) = \frac{P(X)P(Y|X)}{P(Y)} = \nabla_x \bar{f}(\cdot;Y)\#P_X$$

where $$\bar{f} = \operatorname*{arg\,min}_{f \in L^1(\mathcal{X}\times\mathcal{Y})} \mathbb{E}_{(X,Y)\sim P_X \otimes P_Y}[f(X;Y)] + \mathbb{E}_{(X,Y)\sim P_{XY}}[f^\star(X;Y)]$$

Computational properties:

- Only requires samples $(X_i, Y_i) \sim P_{XY}$ (data-driven/simulation based)
- Enables construction of "approximate" posterior distributions
- Allows application of ML tools (stochastic optimization and neural nets)

Optimal transport formulation of the Bayes law

$$\textbf{Bayes law:} \quad P(X|Y) = \frac{P(X)P(Y|X)}{P(Y)}$$
$$= \nabla_x \bar{f}(\cdot;Y)\#P_X$$

where $\bar{f} = \underset{f \in L^1(\mathcal{X}\times\mathcal{Y})}{\arg\min}\; \mathbb{E}_{(X,Y)\sim P_X \otimes P_Y}[f(X;Y)] + \mathbb{E}_{(X,Y)\sim P_{XY}}[f^\star(X;Y)]$

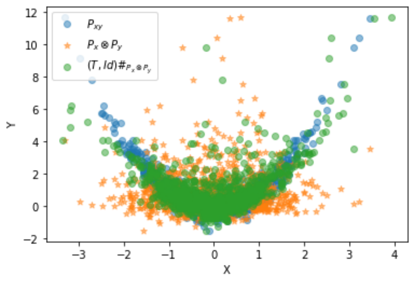
Computational properties:

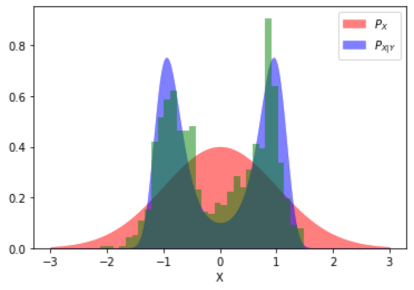
- Only requires samples $(X_i, Y_i) \sim P_{XY}$ (data-driven/simulation based)
- Enables construction of "approximate" posterior distributions
- Allows application of ML tools (stochastic optimization and neural nets)

Numerical example

Approximation with input convex neural networks (ICNN)

- $X \sim N(0,1)$
- $Y = X^2 + \sigma_w W$





Outline

- Background on the filtering problem
- Variational Optimal Transport Particle Filters

Outline

- Background on the filtering problem
- Variational Optimal Transport Particle Filters

Nonlinear filtering problem
Mathematical model

$$
\begin{tikzcd}
{} \arrow[r] & X_{k-1} \arrow[r] \arrow[d] & X_k \arrow[r] \arrow[d] & X_{k+1} \arrow[r] \arrow[d] & {} \\
& Y_{k-1} & Y_k & Y_{k+1} &
\end{tikzcd}
$$

- X_k is the state (unknown)
- Y_k is the observation
- dynamic and observation model are given

Questions: Given history of observation $Y_{1:k} := \{Y_1, \dots, Y_k\}$,

- What is the most likely value of X_k?
- What is the probability of $X_k \in A$?
- What is the best m.s.e estimate for X_k?
- ...

Answer: given by the conditional distribution $\pi_k = \mathsf{P}(X_k | Y_{1:k})$ (posterior, belief)

J. Xiong, An introduction to stochastic filtering theory, 2008

Nonlinear filtering problem
Mathematical model

$$\begin{tikzcd} {} \arrow[r] & X_{k-1} \arrow[r] \arrow[d] & X_k \arrow[r] \arrow[d] & X_{k+1} \arrow[r] \arrow[d] & {} \\ & Y_{k-1} & Y_k & Y_{k+1} & \end{tikzcd}$$

- X_k is the state (unknown)
- Y_k is the observation
- dynamic and observation model are given

Questions: Given history of observation $Y_{1:k} := \{Y_1, \ldots, Y_k\}$,

- What is the most likely value of X_k?
- What is the probability of $X_k \in A$?
- What is the best m.s.e estimate for X_k?
- ...

Answer: given by the conditional distribution $\pi_k = \mathsf{P}(X_k | Y_{1:k})$ (posterior, belief)

J. Xiong, An introduction to stochastic filtering theory, 2008

Nonlinear filtering problem
Mathematical model

$$
\begin{tikzcd}
{} \arrow[r] & X_{k-1} \arrow[r] \arrow[d] & X_k \arrow[r] \arrow[d] & X_{k+1} \arrow[r] \arrow[d] & {} \\
& Y_{k-1} & Y_k & Y_{k+1} &
\end{tikzcd}
$$

- X_k is the state (unknown)
- Y_k is the observation
- dynamic and observation model are given

Questions: Given history of observation $Y_{1:k} := \{Y_1, \dots, Y_k\}$,

- What is the most likely value of X_k? $\arg\max_x \mathsf{P}(X_k = x | Y_{1:k})$
- What is the probability of $X_k \in A$? $\int_A \mathsf{P}(X_k = x | Y_{1:k}) \mathrm{d}x$
- What is the best m.s.e estimate for X_k? $\int x \mathsf{P}(X_k = x | Y_{1:k}) \mathrm{d}x$
- ...

Answer: given by the conditional distribution $\pi_k = \mathsf{P}(X_k | Y_{1:k})$ (posterior, belief)

J. Xiong, An introduction to stochastic filtering theory, 2008

Recursive update for posterior

In principle: given $\pi_k = \mathsf{P}(X_k|Y_{1:k})$, obtain $\pi_{k+1} = \mathsf{P}(X_{k+1}|Y_{1:k+1})$ according to

- Step 1: information update

$$\pi_k = \mathsf{P}(X_k|Y_{1:k}) \xrightarrow{\text{Bayes law}} \tilde{\pi}_k = \mathsf{P}(X_k|Y_{1:k+1})$$

- Step 2: propagation update

$$\tilde{\pi}_k = \mathsf{P}(X_k|Y_{1:k+1}) \xrightarrow{\text{dynamics}} \pi_{k+1} = \mathsf{P}(X_{k+1}|Y_{1:k+1})$$

In practice: No closed-form solution except special cases (linear-Gaussian)

Nonlinear filtering algorithm:

- numerical approximation of π_k
- numerical implementation of the updates

Recursive update for posterior

In principle: given $\pi_k = \mathsf{P}(X_k|Y_{1:k})$, obtain $\pi_{k+1} = \mathsf{P}(X_{k+1}|Y_{1:k+1})$ according to

- Step 1: information update

$$\pi_k = \mathsf{P}(X_k|Y_{1:k}) \xrightarrow{\text{Bayes law}} \tilde{\pi}_k = \mathsf{P}(X_k|Y_{1:k+1})$$

- Step 2: propagation update

$$\tilde{\pi}_k = \mathsf{P}(X_k|Y_{1:k+1}) \xrightarrow{\text{dynamics}} \pi_{k+1} = \mathsf{P}(X_{k+1}|Y_{1:k+1})$$

In practice: No closed-form solution except special cases (linear-Gaussian)

Nonlinear filtering algorithm:

- numerical approximation of π_k
- numerical implementation of the updates

Recursive update for posterior

In principle: given $\pi_k = \mathsf{P}(X_k|Y_{1:k})$, obtain $\pi_{k+1} = \mathsf{P}(X_{k+1}|Y_{1:k+1})$ according to

- Step 1: information update

$$\pi_k = \mathsf{P}(X_k|Y_{1:k}) \xrightarrow{\text{Bayes law}} \tilde{\pi}_k = \mathsf{P}(X_k|Y_{1:k+1})$$

- Step 2: propagation update

$$\tilde{\pi}_k = \mathsf{P}(X_k|Y_{1:k+1}) \xrightarrow{\text{dynamics}} \pi_{k+1} = \mathsf{P}(X_{k+1}|Y_{1:k+1})$$

In practice: No closed-form solution except special cases (linear-Gaussian)

Nonlinear filtering algorithm:

- numerical approximation of π_k
- numerical implementation of the updates

Recursive update for posterior

In principle: given $\pi_k = \mathsf{P}(X_k|Y_{1:k})$, obtain $\pi_{k+1} = \mathsf{P}(X_{k+1}|Y_{1:k+1})$ according to

- Step 1: information update

$$\pi_k = \mathsf{P}(X_k|Y_{1:k}) \xrightarrow{\text{Bayes law}} \tilde{\pi}_k = \mathsf{P}(X_k|Y_{1:k+1})$$

- Step 2: propagation update

$$\tilde{\pi}_k = \mathsf{P}(X_k|Y_{1:k+1}) \xrightarrow{\text{dynamics}} \pi_{k+1} = \mathsf{P}(X_{k+1}|Y_{1:k+1})$$

In practice: No closed-form solution except special cases (linear-Gaussian)

Nonlinear filtering algorithm:

- numerical approximation of π_k
- numerical implementation of the updates

Recursive update for posterior

In principle: given $\pi_k = \mathsf{P}(X_k|Y_{1:k})$, obtain $\pi_{k+1} = \mathsf{P}(X_{k+1}|Y_{1:k+1})$ according to

- Step 1: information update

$$\pi_k = \mathsf{P}(X_k|Y_{1:k}) \xrightarrow{\text{Bayes law}} \tilde{\pi}_k = \mathsf{P}(X_k|Y_{1:k+1})$$

- Step 2: propagation update

$$\tilde{\pi}_k = \mathsf{P}(X_k|Y_{1:k+1}) \xrightarrow{\text{dynamics}} \pi_{k+1} = \mathsf{P}(X_{k+1}|Y_{1:k+1})$$

In practice: No closed-form solution except special cases (linear-Gaussian)

Nonlinear filtering algorithm:

- numerical approximation of π_k
- numerical implementation of the updates

Kalman filter
Linear Gaussian setting

- linear dynamics: $X_{k+1} = AX_k + Be_k$
- linear observation: $Y_k = HX_K + W_k$

Kalman filter: posterior π_k is Gaussian $N(m_k, \Sigma_k)$

$$\text{Update for mean:} \quad m_{k+1} = \underbrace{Am_k}_{\text{dynamics}} + \underbrace{\mathsf{K}_k(Y_k - Hm_k)}_{\text{correction}}$$

$$\text{Update for variance:} \quad \Sigma_{k+1} = \text{(Ricatti equation)}$$

- application in navigation and guidance
- fails to represent multi-modal distributions $\rightarrow$ particle filters

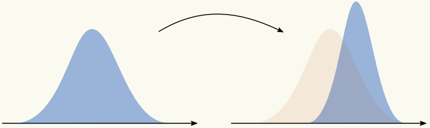
R. E. Kalman, A New Approach to Linear Filtering and Prediction Problems, 1960
R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory, 1961

- linear dynamics: $X_{k+1} = AX_k + Be_k$
- linear observation: $Y_k = HX_K + W_k$

Kalman filter: posterior π_k is Gaussian $N(m_k, \Sigma_k)$

Update for mean: $m_{k+1} = \underbrace{Am_k}_{\text{dynamics}} + \underbrace{\mathsf{K}_k(Y_k - Hm_k)}_{\text{correction}}$

Update for variance: $\Sigma_{k+1} =$ (Ricatti equation)



- application in navigation and guidance
- fails to represent multi-modal distributions → particle filters

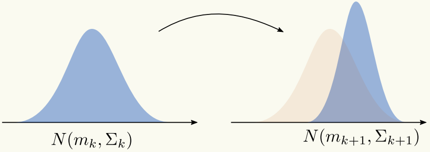
R. E. Kalman, A New Approach to Linear Filtering and Prediction Problems, 1960
R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory, 1961

- linear dynamics: $X_{k+1} = AX_k + Be_k$
- linear observation: $Y_k = HX_K + W_k$

Kalman filter: posterior π_k is Gaussian $N(m_k, \Sigma_k)$

$$\text{Update for mean:} \quad m_{k+1} = \underbrace{Am_k}_{\text{dynamics}} + \underbrace{\mathsf{K}_k(Y_k - Hm_k)}_{\text{correction}}$$

$$\text{Update for variance:} \quad \Sigma_{k+1} = \text{(Ricatti equation)}$$

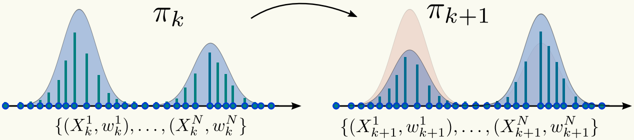


- application in navigation and guidance
- fails to represent multi-modal distributions $\to$ particle filters

R. E. Kalman, A New Approach to Linear Filtering and Prediction Problems, 1960
R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory, 1961

Particle filters
Monte-Carlo approximation

- approximate π_k with weighted empirical distribution of particles
- apply the update rule to the particles and weights



- Step 1: update the weights according to Bayes rule

$$w_{k+1}^i \propto w_k^i P(Y_k | X_k^i)$$

- Step 2: update particles according to the dynamics

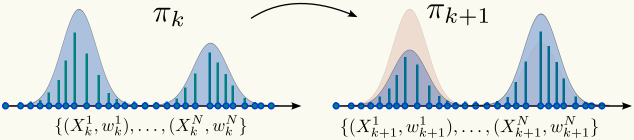
Properties:

- exact in the limit as $N \to \infty$
- weight degeneracy $\to$ curse of dimensionality

N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation (1993).
P. Del Moral, A.Guionnet. On the stability of interacting processes with applications to filtering and genetic algorithms. (2001)
A. Doucet and A. Johansen, A Tutorial on Particle Filtering and Smoothing: Fifteen years later (2008).
P. Bickel, B. Li, and T. Bengtsson, Sharp failure rates for the bootstrap particle filter in high dimensions (2008).

Particle filters
Monte-Carlo approximation

- approximate π_k with weighted empirical distribution of particles
- apply the update rule to the particles and weights



- Step 1: update the weights according to Bayes rule

$$w_{k+1}^i \propto w_k^i P(Y_k | X_k^i)$$

- Step 2: update particles according to the dynamics

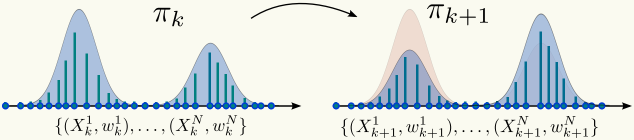
Properties:

- exact in the limit as $N \to \infty$
- weight degeneracy $\to$ curse of dimensionality

N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation (1993).
P. Del Moral, A.Guionnet. On the stability of interacting processes with applications to filtering and genetic algorithms. (2001)
A. Doucet and A. Johansen, A Tutorial on Particle Filtering and Smoothing: Fifteen years later (2008).
P. Bickel, B. Li, and T. Bengtsson, Sharp failure rates for the bootstrap particle filter in high dimensions (2008).

Particle filters
Monte-Carlo approximation

- approximate π_k with weighted empirical distribution of particles
- apply the update rule to the particles and weights



- Step 1: update the weights according to Bayes rule

$$w_{k+1}^i \propto w_k^i P(Y_k | X_k^i)$$

- Step 2: update particles according to the dynamics

Properties:

- exact in the limit as $N \to \infty$
- weight degeneracy $\to$ curse of dimensionality

N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation (1993).
P. Del Moral, A.Guionnet. On the stability of interacting processes with applications to filtering and genetic algorithms. (2001)
A. Doucet and A. Johansen, A Tutorial on Particle Filtering and Smoothing: Fifteen years later (2008).
P. Bickel, B. Li, and T. Bengtsson, Sharp failure rates for the bootstrap particle filter in high dimensions (2008).

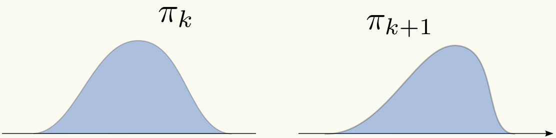
Outline

- Background on the filtering problem
- Variational Optimal Transport Particle Filters

Outline

- Background on the filtering problem
- Variational Optimal Transport Particle Filters

Transport view point



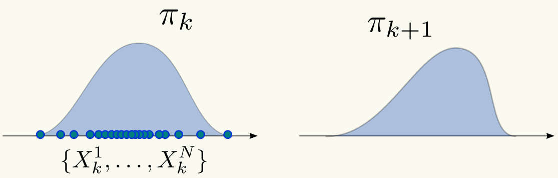
- suppose we have particles that represent samples from π_k
- we like to generate new set of particles that represent samples from π_{k+1}
- the dynamic update is straightforward, however, the Bayes update is challenging

Transport view-point: update particles with a transport map from π_k to π_{k+1}

$$X_{k+1}^i = T_k(X_k^i)$$

Question: How to numerically approximate the transport map T_k?

Transport view point



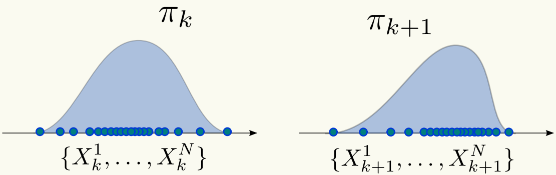
- suppose we have particles that represent samples from π_k
- we like to generate new set of particles that represent samples from π_{k+1}
- the dynamic update is straightforward, however, the Bayes update is challenging

Transport view-point: update particles with a transport map from π_k to π_{k+1}

$$X_{k+1}^i = T_k(X_k^i)$$

Question: How to numerically approximate the transport map T_k?

Transport view point



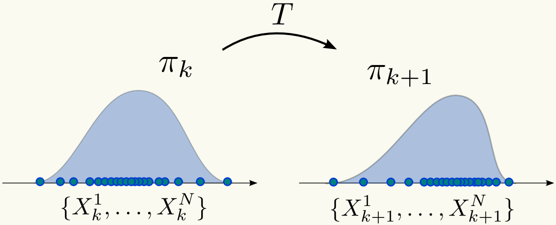
- suppose we have particles that represent samples from π_k
- we like to generate new set of particles that represent samples from π_{k+1}
- the dynamic update is straightforward, however, the Bayes update is challenging

Transport view-point: update particles with a transport map from π_k to π_{k+1}

$$X_{k+1}^i = T_k(X_k^i)$$

Question: How to numerically approximate the transport map T_k?

Transport view point



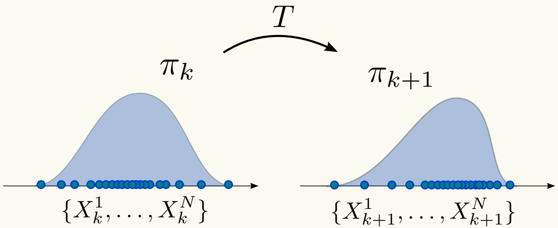
- suppose we have particles that represent samples from π_k
- we like to generate new set of particles that represent samples from π_{k+1}
- the dynamic update is straightforward, however, the Bayes update is challenging

Transport view-point: update particles with a transport map from π_k to π_{k+1}

$$X_{k+1}^i = T_k(X_k^i)$$

Question: How to numerically approximate the transport map T_k?

Transport view point



- suppose we have particles that represent samples from π_k
- we like to generate new set of particles that represent samples from π_{k+1}
- the dynamic update is straightforward, however, the Bayes update is challenging

Transport view-point: update particles with a transport map from π_k to π_{k+1}

$$X_{k+1}^i = T_k(X_k^i)$$

Question: How to numerically approximate the transport map T_k?

Ensemble Kalman filter (EnKF)

Example in the linear-Gaussian setting

- linear observation: $Y_k = HX_K + W_k$
- update law for the particles:

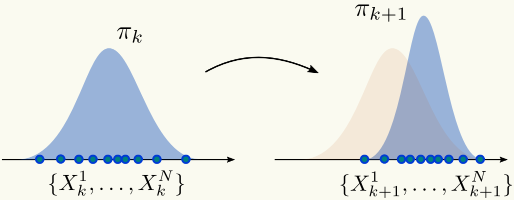
$$X_{k+1}^i = X_k^i + K_k^{(N)}(Y_k - Y_k^i)$$
$$Y_k^i = HX_k^i + W_k^i$$

- where $K^{(N)} = \text{Cov}(X_k^i, Y_k^i)\text{Cov}(Y_k^i)^{-1}$

- linear observation: $Y_k = HX_K + W_k$
- update law for the particles:

$$X^i_{k+1} = X^i_k + K^{(N)}_k (Y_k - Y^i_k)$$
$$Y^i_k = HX^i_k + W^i_k$$

- where $K^{(N)} = \text{Cov}(X^i_k, Y^i_k)\text{Cov}(Y^i_k)^{-1}$



Curse of dimensionality
Linear Gaussian setting

- Method 1: Ensemble Kalman filter
- Method 2: standard particle filter
- Compare number of particles to achieve error ϵ:

Question: How can we extend EnKF to nonlinear setting?

S. C. Surace, A. Kutschireiter, J. Pfister, How to avoid the curse of dimensionality: scalability of particle filters . . . , SIAM review, 2019
A. Taghvaei, P. G. Mehta, An optimal transport formulation of ensemble Kalman filter, (TAC) 2020

Curse of dimensionality
Linear Gaussian setting

- Method 1: Ensemble Kalman filter
- Method 2: standard particle filter
- Compare number of particles to achieve error ϵ:

Question: How can we extend EnKF to nonlinear setting?

S. C. Surace, A. Kutschireiter, J. Pfister, How to avoid the curse of dimensionality: scalability of particle filters . . . , SIAM review, 2019
A. Taghvaei, P. G. Mehta, An optimal transport formulation of ensemble Kalman filter, (TAC) 2020

Curse of dimensionality
Linear Gaussian setting

- Method 1: Ensemble Kalman filter
- Method 2: standard particle filter
- Compare number of particles to achieve error ϵ:

Question: How can we extend EnKF to nonlinear setting?

S. C. Surace, A. Kutschireiter, J. Pfister, How to avoid the curse of dimensionality: scalability of particle filters . . . , SIAM review, 2019
A. Taghvaei, P. G. Mehta, An optimal transport formulation of ensemble Kalman filter, (TAC) 2020

Curse of dimensionality
Linear Gaussian setting

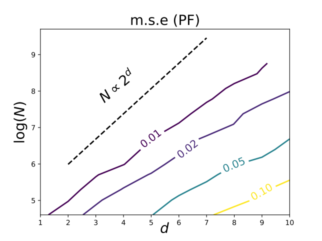
- Method 1: Ensemble Kalman filter
- Method 2: standard particle filter
- Compare number of particles to achieve error ϵ:

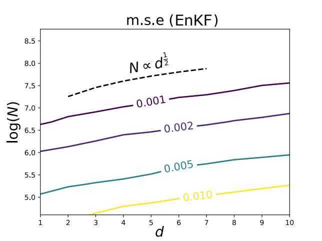
Question: How can we extend EnKF to nonlinear setting?

S. C. Surace, A. Kutschireiter, J. Pfister, How to avoid the curse of dimensionality: scalability of particle filters ..., SIAM review, 2019
A. Taghvaei, P. G. Mehta, An optimal transport formulation of ensemble Kalman filter, (TAC) 2020

Curse of dimensionality
Linear Gaussian setting

- Method 1: Ensemble Kalman filter
- Method 2: standard particle filter
- Compare number of particles to achieve error ϵ:



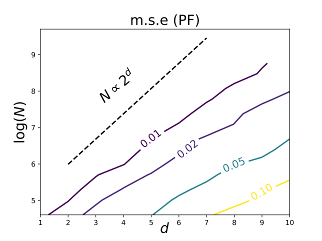


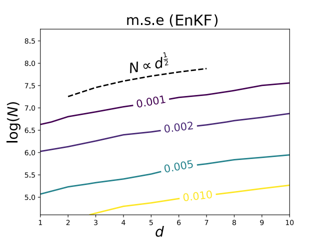
PF: $N \approx O(\frac{e^d}{\epsilon})$ EnKF: $N \approx O(\frac{d^2}{\epsilon})$

Question: How can we extend EnKF to nonlinear setting?

S. C. Surace, A. Kutschireiter, J. Pfister, How to avoid the curse of dimensionality: scalability of particle filters . . . , SIAM review, 2019
A. Taghvaei, P. G. Mehta, An optimal transport formulation of ensemble Kalman filter, (TAC) 2020

- Method 1: Ensemble Kalman filter
- Method 2: standard particle filter
- Compare number of particles to achieve error ϵ:





PF: $N \approx O(\dfrac{e^d}{\epsilon})$ EnKF: $N \approx O(\dfrac{d^2}{\epsilon})$

Question: How can we extend EnKF to nonlinear setting?

S. C. Surace, A. Kutschireiter, J. Pfister, How to avoid the curse of dimensionality: scalability of particle filters . . . , SIAM review, 2019
A. Taghvaei, P. G. Mehta, An optimal transport formulation of ensemble Kalman filter, (TAC) 2020

Variational optimal transport filter

- observation $Y_k \sim h(\cdot|X_k)$
- given particles $\{X_k^i\}_{i=1}^N \sim \pi_k$, generate

$$Y_k^i \sim h(\cdot|X_k^i)$$

- use $\{(X_k^i, Y_k^i)\}_{i=1}^N$ to obtain $\bar{f}$ by solving

$$\min_{f \in \mathcal{F}} \frac{1}{N}\sum_{i=1}^N f(X_k^i; Y_k^{\sigma_i}) + \frac{1}{N}\sum_{i=1}^N f^\star(X_k^i; Y_k^i)$$

- where $\mathcal{F}$ is a paramteric class of functions
  - Class of quadratic functions $\rightarrow$ Optimal Transport EnKF
  - Input convex neural networks
- update the particles according to

$$X_{k+1}^i = \nabla \bar{f}_k(X_k^i, Y_k)$$

Variational optimal transport filter

- observation $Y_k \sim h(\cdot|X_k)$
- given particles $\{X_k^i\}_{i=1}^N \sim \pi_k$, generate

$$Y_k^i \sim h(\cdot|X_k^i)$$

- use $\{(X_k^i, Y_k^i)\}_{i=1}^N$ to obtain $\bar{f}$ by solving

$$\min_{f \in \mathcal{F}} \frac{1}{N}\sum_{i=1}^N f(X_k^i; Y_k^{\sigma_i}) + \frac{1}{N}\sum_{i=1}^N f^\star(X_k^i; Y_k^i)$$

- where $\mathcal{F}$ is a paramteric class of functions
  - Class of quadratic functions $\rightarrow$ Optimal Transport EnKF
  - Input convex neural networks
- update the particles according to

$$X_{k+1}^i = \nabla \bar{f}_k(X_k^i, Y_k)$$

Variational optimal transport filter

- observation $Y_k \sim h(\cdot|X_k)$
- given particles $\{X_k^i\}_{i=1}^N \sim \pi_k$, generate

$$Y_k^i \sim h(\cdot|X_k^i)$$

- use $\{(X_k^i, Y_k^i)\}_{i=1}^N$ to obtain $\bar{f}$ by solving

$$\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^N f(X_k^i; Y_k^{\sigma_i}) + \frac{1}{N} \sum_{i=1}^N f^\star(X_k^i; Y_k^i)$$

- where $\mathcal{F}$ is a paramteric class of functions
  - Class of quadratic functions $\rightarrow$ Optimal Transport EnKF
  - Input convex neural networks
- update the particles according to

$$X_{k+1}^i = \nabla \bar{f}_k(X_k^i, Y_k)$$

Variational optimal transport filter

- observation $Y_k \sim h(\cdot|X_k)$
- given particles $\{X_k^i\}_{i=1}^N \sim \pi_k$, generate

$$Y_k^i \sim h(\cdot|X_k^i)$$

- use $\{(X_k^i, Y_k^i)\}_{i=1}^N$ to obtain $\bar{f}$ by solving

$$\min_{f \in \mathcal{F}} \frac{1}{N}\sum_{i=1}^N f(X_k^i; Y_k^{\sigma_i}) + \frac{1}{N}\sum_{i=1}^N f^\star(X_k^i; Y_k^i)$$

- where $\mathcal{F}$ is a paramteric class of functions
  - Class of quadratic functions $\rightarrow$ Optimal Transport EnKF
  - Input convex neural networks
- update the particles according to

$$X_{k+1}^i = \nabla \bar{f}_k(X_k^i, Y_k)$$

Variational optimal transport filter

- observation $Y_k \sim h(\cdot|X_k)$
- given particles $\{X_k^i\}_{i=1}^N \sim \pi_k$, generate

$$Y_k^i \sim h(\cdot|X_k^i)$$

- use $\{(X_k^i, Y_k^i)\}_{i=1}^N$ to obtain $\bar{f}$ by solving

$$\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^{N} f(X_k^i; Y_k^{\sigma_i}) + \frac{1}{N} \sum_{i=1}^{N} f^\star(X_k^i; Y_k^i)$$

- where $\mathcal{F}$ is a paramteric class of functions
  - Class of quadratic functions $\rightarrow$ Optimal Transport EnKF
  - Input convex neural networks
- update the particles according to

$$X_{k+1}^i = \nabla \bar{f}_k(X_k^i, Y_k)$$

Derivation of the variational formula

- We want to find a map T that transports P_X to $P_{X|Y}$ with minimum cost

$$\min_T \mathbb{E}_{X\sim P_X}[\|T(X)-X\|^2], \quad \text{s.t.} \quad T\#P_X = P_{X|Y}$$

- The Kantorovich dual formulation removes the constraint

$$\min_{f\in L^1(\mathcal{X})} \mathbb{E}_{X\sim P_X}[f(X)] + \mathbb{E}_{X\sim P_{X|Y}}[f^\star(X)] \quad \text{but } P_{X|Y} \text{ is not available}$$

- Take expectation with respect to Y

$$\min_{f\in L^1(\mathcal{X}\times\mathcal{Y})} \mathbb{E}_{(X,Y)\sim P_X\otimes P_Y}[f(X;Y)] + \mathbb{E}_{(X,Y)\sim P_{XY}}[f^\star(X;Y)]$$

Theorem

Assume $\mathbb{E}[\|X\|^2]<\infty$ and P_X admits density.
Then, the variational problem admits a unique solution $\bar{f}$ that satisfies:

$$P_{X|Y} = \nabla_x \bar{f}(\cdot;Y)\#P_X, \quad \text{(a.e.)}$$

Derivation of the variational formula

- We want to find a map T that transports P_X to $P_{X|Y}$ with minimum cost

$$\min_T \mathbb{E}_{X\sim P_X}[\|T(X)-X\|^2], \quad \text{s.t.} \quad T\#P_X = P_{X|Y}$$

- The Kantorovich dual formulation removes the constraint

$$\min_{f\in L^1(\mathcal{X})} \mathbb{E}_{X\sim P_X}[f(X)] + \mathbb{E}_{X\sim P_{X|Y}}[f^\star(X)] \quad \text{but } P_{X|Y} \text{ is not available}$$

- Take expectation with respect to Y

$$\min_{f\in L^1(\mathcal{X}\times\mathcal{Y})} \mathbb{E}_{(X,Y)\sim P_X\otimes P_Y}[f(X;Y)] + \mathbb{E}_{(X,Y)\sim P_{XY}}[f^\star(X;Y)]$$

Theorem

Assume $\mathbb{E}[\|X\|^2] < \infty$ and P_X admits density.
Then, the variational problem admits a unique solution $\bar{f}$ that satisfies:

$$P_{X|Y} = \nabla_x \bar{f}(\cdot;Y)\#P_X, \quad \text{(a.e.)}$$

Derivation of the variational formula

- We want to find a map T that transports P_X to $P_{X|Y}$ with minimum cost

$$\min_{T} \mathbb{E}_{X\sim P_X}[\|T(X) - X\|^2], \quad \text{s.t.} \quad T\#P_X = P_{X|Y}$$

- The Kantorovich dual formulation removes the constraint

$$\min_{f\in L^1(\mathcal{X})} \mathbb{E}_{X\sim P_X}[f(X)] + \mathbb{E}_{X\sim P_{X|Y}}[f^\star(X)] \quad \text{but } P_{X|Y} \text{ is not available}$$

- Take expectation with respect to Y

$$\min_{f\in L^1(\mathcal{X}\times\mathcal{Y})} \mathbb{E}_{(X,Y)\sim P_X\otimes P_Y}[f(X;Y)] + \mathbb{E}_{(X,Y)\sim P_{XY}}[f^\star(X;Y)]$$

Theorem

Assume $\mathbb{E}[\|X\|^2] < \infty$ and P_X admits density.
Then, the variational problem admits a unique solution $\bar{f}$ that satisfies:

$$P_{X|Y} = \nabla_x \bar{f}(\cdot;Y)\#P_X, \quad \text{(a.e.)}$$

Derivation of the variational formula

- We want to find a map T that transports P_X to $P_{X|Y}$ with minimum cost

$$\min_T \mathbb{E}_{X\sim P_X}[\|T(X)-X\|^2], \quad \text{s.t.} \quad T\#P_X = P_{X|Y}$$

- The Kantorovich dual formulation removes the constraint

$$\min_{f\in L^1(\mathcal{X})} \mathbb{E}_{X\sim P_X}[f(X)] + \mathbb{E}_{X\sim P_{X|Y}}[f^\star(X)] \quad \text{but } P_{X|Y} \text{ is not available}$$

- Take expectation with respect to Y

$$\min_{f\in L^1(\mathcal{X}\times\mathcal{Y})} \mathbb{E}_{(X,Y)\sim P_X\otimes P_Y}[f(X;Y)] + \mathbb{E}_{(X,Y)\sim P_{XY}}[f^\star(X;Y)]$$

Theorem

Assume $\mathbb{E}[\|X\|^2]<\infty$ and P_X admits density.
Then, the variational problem admits a unique solution $\bar{f}$ that satisfies:

$$P_{X|Y} = \nabla_x \bar{f}(\cdot;Y)\#P_X, \quad \text{(a.e.)}$$

Summary

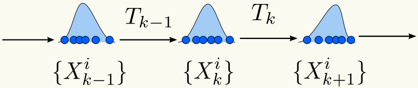
- **Mathematical model:**

$$\begin{tikzcd} {} \arrow[r] & X_{k-1} \arrow[r] \arrow[d] & X_k \arrow[r] \arrow[d] & X_{k+1} \arrow[r] \arrow[d] & {} \\ & Y_{k-1} & Y_k & Y_{k+1} & \end{tikzcd}$$

- **Nonlinear filtering:** compute the posterior $\pi_k = \mathsf{P}(X_k | Y_{1:k})$

$$\longrightarrow \pi_{k-1} \longrightarrow \pi_k \longrightarrow \pi_{k+1} \longrightarrow$$

- **OT approach:**



- **Variational problem:**

$$T_k = \nabla_x \bar{f}_k, \quad \text{where} \quad \bar{f}_k = \underset{f \in \mathcal{F}}{\arg\min}\ J^{(N)}(f; \{(X^i_k, Y^i_k)\})$$

Optimal transportation methods in nonlinear filtering: The feedback particle filter, IEEE CSM, 2021