Controlled Interacting Particle Systems for Estimation and Sampling

Amirhossein Taghvaei

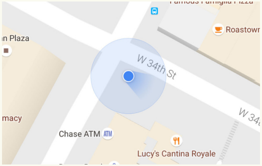
Postdoctoral Scholar
Department of Mechanical and Aerospace Engineering
University of California, Irvine

Presented at the Department of Aeronautics & Astronautics
University of Washington, Seattle

Feb 25, 2021

UCI

Uncertainty is everywhere



GPS location

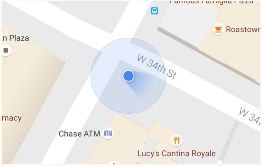
weather forecast

COVID-19

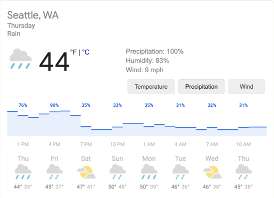
We have to deal with uncertainty

I. Stewart, Do Dice Play God? The Mathematics of Uncertainty. United States: Basic Books. 2019

Uncertainty is everywhere



GPS location



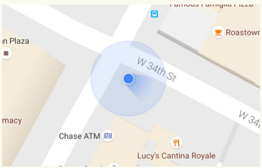
weather forecast

COVID-19

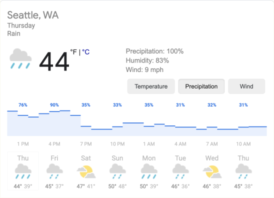
We have to deal with uncertainty

I. Stewart, Do Dice Play God? The Mathematics of Uncertainty. United States: Basic Books. 2019

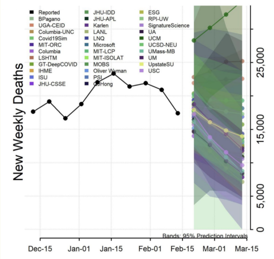
Uncertainty is everywhere



GPS location



weather forecast

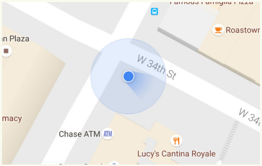


COVID-19

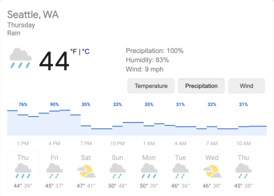
We have to deal with uncertainty

I. Stewart, Do Dice Play God? The Mathematics of Uncertainty. United States: Basic Books. 2019

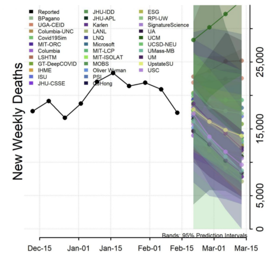
Uncertainty is everywhere



GPS location



weather forecast



COVID-19

imperfect models

noise in measurements

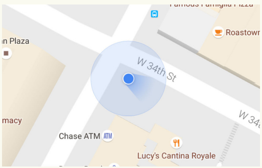
uncertainty

stochastic algorithms

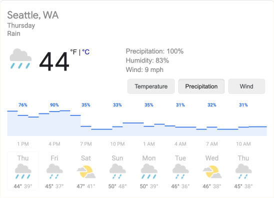
We have to deal with uncertainty

I. Stewart, Do Dice Play God? The Mathematics of Uncertainty. United States: Basic Books. 2019

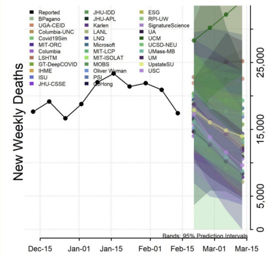
Uncertainty is everywhere



GPS location



weather forecast



COVID-19

imperfect models

noise in measurements

uncertainty

stochastic algorithms

We have to deal with uncertainty

I. Stewart, Do Dice Play God? The Mathematics of Uncertainty. United States: Basic Books. 2019

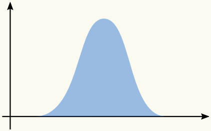
Probability theory: (quantify uncertainty)

Optimal transport theory: (geometry for distributions)

Allows application of control and optimization techniques to distributions

To be certain about uncertainty

Probability theory: (quantify uncertainty)



Optimal transport theory: (geometry for distributions)

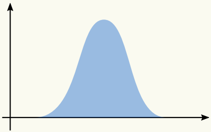
Nobel prize (1975)

Fields medal (2010)

Allows application of control and optimization techniques to distributions

To be certain about uncertainty

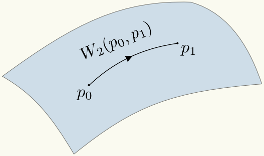
Probability theory: (quantify uncertainty)



Optimal transport theory: (geometry for distributions)

Nobel prize (1975)

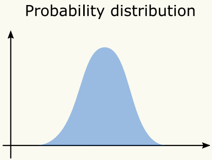
Fields medal (2010)



Allows application of control and optimization techniques to distributions

To be certain about uncertainty

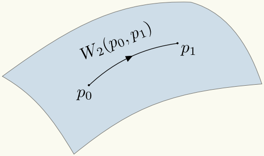
Probability theory: (quantify uncertainty)



Optimal transport theory: (geometry for distributions)

Nobel prize (1975)

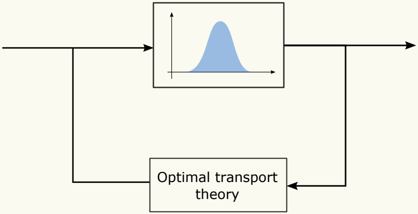
Fields medal (2010)



Allows application of control and optimization techniques to distributions

Research overview

Main theme: Control/Optimization for probability distributions



Nonlinear Estimation/filtering — this talk!

- publications in IEEE TAC, SIAM UQ, ASME, IEEE CSM

Machine learning

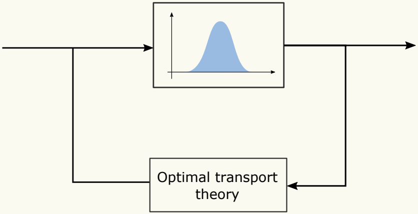
- publications in NeurIPS 17, ICML 19, ICML 20

Stochastic thermodynamics

- publications in Automatica, Scientific reports

Research overview

Main theme: Control/Optimization for probability distributions



Nonlinear Estimation/filtering → this talk!

- publications in IEEE TAC, SIAM UQ, ASME, IEEE CSM

Machine learning

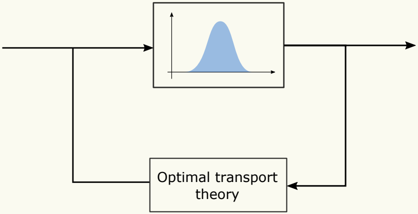
- publications in NeurIPS 17, ICML 19, ICML 20

Stochastic thermodynamics

- publications in Automatica, Scientific reports

Research overview

Main theme: Control/Optimization for probability distributions



Nonlinear Estimation/filtering → this talk!

- publications in IEEE TAC, SIAM UQ, ASME, IEEE CSM

Machine learning

- publications in NeurIPS 17, ICML 19, ICML 20

Stochastic thermodynamics

- publications in Automatica, Scientific reports

Outline

Part I: background and motivation

Part II: mean-field control design

Part III: gain function approximation

Part IV: applications

Outline

Part I: background and motivation

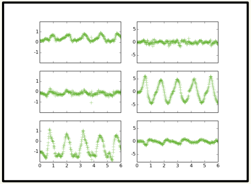
Part II: mean-field control design

Part III: gain function approximation

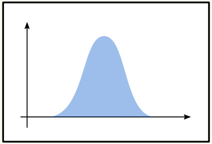
Part IV: applications

Nonlinear filtering problem

state



observation



posterior distribution

Hidden state: physical activity

Observation: inertial motion sensors

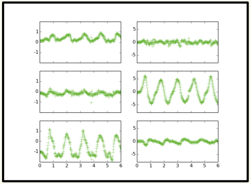
Problem: estimate the state based on observation

Probabilistic approach: compute the conditional probability distribution (posterior)

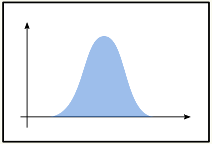
Internship at university start-up company, Rithmio, 2014-2015

Nonlinear filtering problem

state



observation



posterior distribution

Hidden state: physical activity

Observation: inertial motion sensors

Problem: estimate the state based on observation

Probabilistic approach: compute the conditional probability distribution (posterior)

Internship at university start-up company, Rithmio, 2014-2015

$$\begin{array}{ccccccc} \longrightarrow & X_{k-1} & \longrightarrow & X_k & \longrightarrow & X_{k+1} & \longrightarrow \\ & \downarrow & & \downarrow & & \downarrow & \\ & Z_{k-1} & & Z_k & & Z_{k+1} & \end{array}$$

State process: $X_k \sim a(\cdot|X_{k-1}), \qquad X_0 \sim \pi_0(\cdot)$

Observation process: $Z_k \sim l(\cdot|X_k)$

Objective: compute $\pi_k(\cdot) := \mathsf{P}(X_k \in \cdot|Z_{1:k})$

In principle: recursive update for posterior

$$\pi_k \xrightarrow{\text{dynamics}} \tilde{\pi}_k \xrightarrow[\text{Bayes rule}]{\text{correction}} \pi_{k+1}$$

In practice: no finite-dimensional solution

- notable exception: linear Gaussian case

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

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

State process: $X_k \sim a(\cdot|X_{k-1}), \qquad X_0 \sim \pi_0(\cdot)$

Observation process: $Z_k \sim l(\cdot|X_k)$

Objective: compute $\pi_k(\cdot) := \mathsf{P}(X_k \in \cdot|Z_{1:k})$

In principle: recursive update for posterior

$$\pi_k \xrightarrow{\text{dynamics}} \tilde{\pi}_k \xrightarrow[\text{Bayes rule}]{\text{correction}} \pi_{k+1}$$

In practice: no finite-dimensional solution

- notable exception: linear Gaussian case

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

Nonlinear filtering problem
Mathematical model

$$
\begin{array}{ccccccc}
\longrightarrow X_{k-1} & \longrightarrow & X_k & \longrightarrow & X_{k+1} & \longrightarrow \\
\downarrow & & \downarrow & & \downarrow & \\
Z_{k-1} & & Z_k & & Z_{k+1} &
\end{array}
$$

State process: $X_k \sim a(\cdot|X_{k-1}), \qquad X_0 \sim \pi_0(\cdot)$

Observation process: $Z_k \sim l(\cdot|X_k)$

Objective: compute $\pi_k(\cdot) := \mathsf{P}(X_k \in \cdot|Z_{1:k})$

In principle: recursive update for posterior

$$\pi_k \xrightarrow{\text{dynamics}} \tilde{\pi}_k \xrightarrow[\text{Bayes rule}]{\text{correction}} \pi_{k+1}$$

In practice: no finite-dimensional solution

- notable exception: linear Gaussian case

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

Nonlinear filtering problem
Continuous-time formulation

State process: $\mathrm{d}X_t = \text{(dynamic model)} \qquad X_0 \sim \pi_0(\cdot)$

Observation process: $\mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t$

Objective: compute $\pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]})$

In principle: continuous update law for posterior

$$\mathrm{d}\pi_t = \text{(dynamics)} + \text{(correction)}$$

In practice: no finite-dimensional solution

- notable exception: linear Gaussian case

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

Nonlinear filtering problem
Continuous-time formulation

State process: $\mathrm{d}X_t = \text{(dynamic model)} \qquad X_0 \sim \pi_0(\cdot)$

Observation process: $\mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t$

$$Y_t := \frac{\mathrm{d}}{\mathrm{d}t} Z_t = h(X_t) + \text{white noise}$$

Objective: compute $\pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]})$

In principle: continuous update law for posterior

$$\mathrm{d}\pi_t = \text{(dynamics)} + \text{(correction)}$$

In practice: no finite-dimensional solution

- notable exception: linear Gaussian case

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

Nonlinear filtering problem
Continuous-time formulation

State process: $\mathrm{d}X_t = \text{(dynamic model)} \qquad X_0 \sim \pi_0(\cdot)$

Observation process: $\mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t$

Objective: compute $\pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]})$

In principle: continuous update law for posterior

$$\mathrm{d}\pi_t = \text{(dynamics)} + \text{(correction)}$$

In practice: no finite-dimensional solution

- notable exception: linear Gaussian case

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

Nonlinear filtering problem

Continuous-time formulation

State process: $\mathrm{d}X_t = \text{(dynamic model)} \qquad X_0 \sim \pi_0(\cdot)$

Observation process: $\mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t$

Objective: compute $\pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]})$

In principle: continuous update law for posterior

$$\mathrm{d}\pi_t = \text{(dynamics)} + \text{(correction)}$$

In practice: no finite-dimensional solution

- notable exception: linear Gaussian case

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

- linear dynamics: $AX_t\mathrm{d}t + \mathrm{d}B_t$
- linear observation: $h(x) = Hx$

Kalman filter: posterior π_t is Gaussian $N(m_t, \Sigma_t)$

Update for mean: $\mathrm{d}m_t = \underbrace{Am_t\mathrm{d}t}_{\text{dynamics}} + \underbrace{\mathsf{K}_t(\mathrm{d}Z_t - Hm_t\mathrm{d}t)}_{\text{correction}}$

Update for variance: $\dot{\Sigma}_t =$ (Ricatti equation)

Kalman gain: $\mathsf{K}_t = \Sigma_t H^T$

Properties:

- strong theoretical properties
- if state dimension is d, computational cost $O(d^2) \rightarrow$ Ensemble Kalman filter
- exact only in linear Gaussian setting $\rightarrow$ Particle filter

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

Kalman-Bucy filter
Linear Gaussian setting

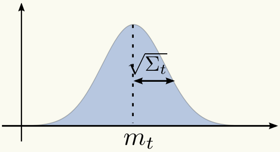
- linear dynamics: $AX_t\mathrm{d}t + \mathrm{d}B_t$
- linear observation: $h(x) = Hx$

Kalman filter: posterior π_t is Gaussian $N(m_t, \Sigma_t)$

Update for mean: $\mathrm{d}m_t = \underbrace{Am_t\mathrm{d}t}_{\text{dynamics}} + \underbrace{\mathsf{K}_t(\mathrm{d}Z_t - Hm_t\mathrm{d}t)}_{\text{correction}}$

Update for variance: $\dot{\Sigma}_t =$ (Ricatti equation)

Kalman gain: $\mathsf{K}_t = \Sigma_t H^T$



Properties:

- strong theoretical properties
- if state dimension is d, computational cost $O(d^2)$ → Ensemble Kalman filter
- exact only in linear Gaussian setting → Particle filter

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

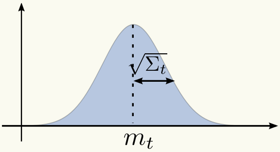
- linear dynamics: $AX_t\mathrm{d}t + \mathrm{d}B_t$
- linear observation: $h(x) = Hx$

Kalman filter: posterior π_t is Gaussian $N(m_t, \Sigma_t)$

Update for mean: $\mathrm{d}m_t = \underbrace{Am_t\mathrm{d}t}_{\text{dynamics}} + \underbrace{\mathsf{K}_t(\mathrm{d}Z_t - Hm_t\mathrm{d}t)}_{\text{correction}}$

Update for variance: $\dot{\Sigma}_t =$ (Ricatti equation)

Kalman gain: $\mathsf{K}_t = \Sigma_t H^T$



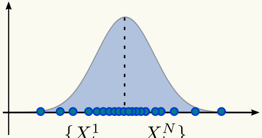
Properties:

- strong theoretical properties
- if state dimension is d, computational cost $O(d^2)$ $\rightarrow$ Ensemble Kalman filter
- exact only in linear Gaussian setting $\rightarrow$ Particle filter

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

Ensemble Kalman filter

Monte-Carlo approximation of Kalman filter



$\{X_t^1, \dots, X_t^N\}$

Posterior distribution: $\pi_t \approx \frac{1}{N}\sum_{i=1}^{N} \delta_{X_t^i}$

Update for particles: $\mathrm{d}X_t^i = \text{(dynamics)} + \underbrace{\mathsf{K}_t^{(N)}(\mathrm{d}Z_t - \frac{HX_t^i + Hm_t^{(N)}}{2}\mathrm{d}t)}_{\text{correction}}$

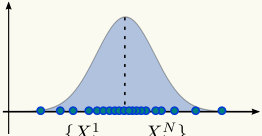
Kalman gain: $\mathsf{K}_t^{(N)} = \Sigma_t^{(N)} H^T$

- exact mean and variance in mean-field limit ($N \to \infty$)
- computational cost $O(Nd)$, efficient when $d \gg N$
- not exact in nonlinear setting

G. Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model . . . 1994.
K. Bergemann and S. Reich. An ensemble Kalman-Bucy filter for continuous data assimilation, 2012

Ensemble Kalman filter

Monte-Carlo approximation of Kalman filter



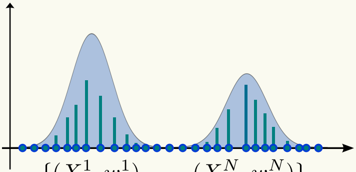
Posterior distribution: $\pi_t \approx \frac{1}{N} \sum_{i=1}^{N} \delta_{X_t^i}$

Update for particles: $\mathrm{d}X_t^i = (\text{dynamics}) + \underbrace{\mathsf{K}_t^{(N)}(\mathrm{d}Z_t - \frac{HX_t^i + Hm_t^{(N)}}{2}\mathrm{d}t)}_{\text{correction}}$

Kalman gain: $\mathsf{K}_t^{(N)} = \Sigma_t^{(N)} H^T$

- exact mean and variance in mean-field limit $(N \to \infty)$
- computational cost $O(Nd)$, efficient when $d \gg N$
- not exact in nonlinear setting

G. Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model ... 1994.
K. Bergemann and S. Reich. An ensemble Kalman-Bucy filter for continuous data assimilation, 2012



$$\{(X_t^1, w_t^1), \dots, (X_t^N, w_t^N)\}$$

Algorithm:

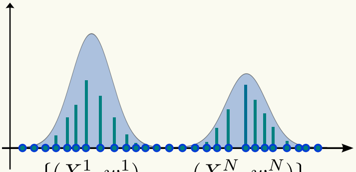
- approximate π_t with <u>weighted</u> empirical distribution $\sum_{i=1}^{N} w_i \delta_{X_t^i}$
- update the weights using observation and Bayes rule (importance sampling)

Properties:

- exact in the limit as $N \to \infty$
- weight degeneracy $\to$ curse of dimensionality
- no feedback control structure

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
Sequential importance sampling



Algorithm:

- approximate π_t with weighted empirical distribution $\sum_{i=1}^{N} w_i \delta_{X_t^i}$
- update the weights using observation and Bayes rule (importance sampling)

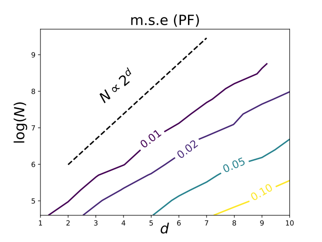
Properties:

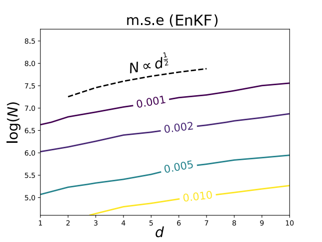
- exact in the limit as $N \to \infty$
- weight degeneracy $\to$ curse of dimensionality
- no feedback control structure

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).

Curse of dimensionality
Linear Gaussian setting

- number of particles to achieve error ϵ:





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

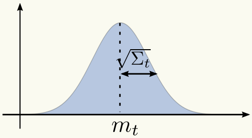
EnKF: $N \approx O(\frac{d^2}{\epsilon})$

Question: Can we generalize EnKF to nonlinear setting?

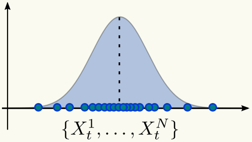
S. C. Surace, A. Kutschireiter, J. Pfister, How to avoid the curse of dimensionality: scalability of particle filters . . . , SIAM review, 2019
A. Taghvaei, P. G. Mehta, An optimal transport formulation of ensemble Kalman filter, (TAC) 2020

Part I: summary

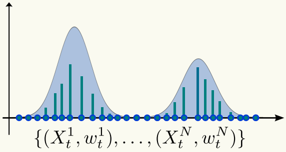
Kalman filter



Ensemble Kalman filter



Particle filter



KF: $\mathrm{d}m_t = (\text{dynamics}) + \mathsf{K}_t(\mathrm{d}Z_t - Hm_t\mathrm{d}t)$

EnKF: $\mathrm{d}X_t^i = (\text{dynamics}) + \mathsf{K}_t^{(N)}(\mathrm{d}Z_t - \frac{HX_k^i + Hm_t^{(N)}}{2}\mathrm{d}t)$

PF: no feedback control structure

Question: Can we generalize EnKF to nonlinear and non-Gaussian setting?

Outline

Part I: background and motivation

Part II: mean-field control design

Part III: gain function approximation

Part III: applications

References:

- A. Taghvaei, P. G. Mehta, Optimal Transportation Methods in Nonlinear Filtering: The feedback particle filter, IEEE Control Systems Magazine (CSM), Accepted
- A. Taghvaei, P. G. Mehta, An optimal transport formulation of the Ensemble Kalman filter, Transactions of Automatic Control (TAC), Accepted

Outline

Part I: background and motivation

Part II: mean-field control design

Part III: gain function approximation

Part III: applications

References:

- A. Taghvaei, P. G. Mehta, Optimal Transportation Methods in Nonlinear Filtering: The feedback particle filter, IEEE Control Systems Magazine (CSM), Accepted
- A. Taghvaei, P. G. Mehta, An optimal transport formulation of the Ensemble Kalman filter, Transactions of Automatic Control (TAC), Accepted

Mean-field design

$$\text{State process:} \quad \mathrm{d}X_t = \text{(dynamic model)} \qquad X_0 \sim \pi_0(\cdot)$$

$$\text{Observation process:} \quad \mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t$$

$$\text{Objective:} \quad \text{compute } \pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]})$$

Idea:

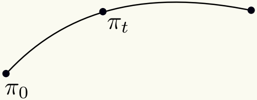
Mean-field design

State process: $\mathrm{d}X_t = \text{(dynamic model)} \qquad X_0 \sim \pi_0(\cdot)$

Observation process: $\mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t$

Objective: compute $\pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]})$

Idea:



Mean-field design

$$\text{State process:} \quad \mathrm{d}X_t = \text{(dynamic model)} \qquad X_0 \sim \pi_0(\cdot)$$

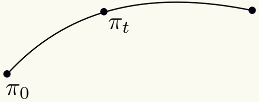
$$\text{Observation process:} \quad \mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t$$

$$\text{Objective:} \quad \text{compute } \pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]})$$

Idea:

- construct a controlled stochastic process

$$\mathrm{d}\bar{X}_t = \text{(dynamics)} + \underbrace{u_t(\bar{X}_t)\mathrm{d}t + \mathsf{K}_t(\bar{X}_t)\mathrm{d}Z_t}_{\text{control}}$$



- design the control law such that

$$\bar{X}_t \sim \pi_t, \quad \forall t \geq 0 \quad \text{(exactness)}$$

- realize $\bar{X}_t$ with N particles $\{X_t^1, \ldots, X_t^N\}$

Mean-field design

State process: $\mathrm{d}X_t = \text{(dynamic model)} \qquad X_0 \sim \pi_0(\cdot)$

Observation process: $\mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t$

Objective: compute $\pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]})$

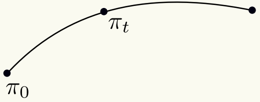
Idea:

- construct a controlled stochastic process

$$\mathrm{d}\bar{X}_t = \text{(dynamics)} + \underbrace{u_t(\bar{X}_t)\mathrm{d}t + \mathsf{K}_t(\bar{X}_t)\mathrm{d}Z_t}_{\text{control}}$$

- design the control law such that

$$\bar{X}_t \sim \pi_t, \quad \forall t \geq 0 \quad \text{(exactness)}$$



- realize $\bar{X}_t$ with N particles $\{X_t^1, \ldots, X_t^N\}$

Mean-field design

$$\text{State process:} \quad \mathrm{d}X_t = \text{(dynamic model)} \qquad X_0 \sim \pi_0(\cdot)$$

$$\text{Observation process:} \quad \mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t$$

$$\text{Objective:} \quad \text{compute } \pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]})$$

Idea:

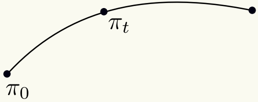
- construct a controlled stochastic process

$$\mathrm{d}\bar{X}_t = \text{(dynamics)} + \underbrace{u_t(\bar{X}_t)\mathrm{d}t + \mathsf{K}_t(\bar{X}_t)\mathrm{d}Z_t}_{\text{control}}$$

- design the control law such that

$$\bar{X}_t \sim \pi_t, \quad \forall t \geq 0 \quad \text{(exactness)}$$

- realize $\bar{X}_t$ with N particles $\{X_t^1, \ldots, X_t^N\}$



Example

Objective: construct $\bar{X}_t \sim \pi_t = N(0, 1+t)$

Two solutions:

As $N \to \infty$, they have the same marginal distribution π_t

However, different couplings between two time instants

Example

Objective: construct $\bar{X}_t \sim \pi_t = N(0, 1+t)$

Two solutions:

(I) $\mathrm{d}\bar{X}_t = \mathrm{d}B_t, \quad \bar{X}_0 \sim N(0,1)$

(II) $\frac{\mathrm{d}}{\mathrm{d}t}\bar{X}_t = \frac{\bar{X}_t}{2\bar{\Sigma}_t}, \quad \bar{X}_0 \sim N(0,1)$

As $N \to \infty$, they have the same marginal distribution π_t

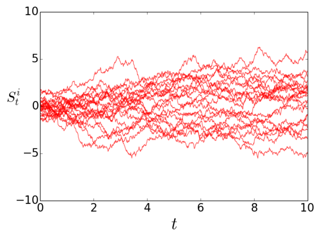
However, different couplings between two time instants

Example

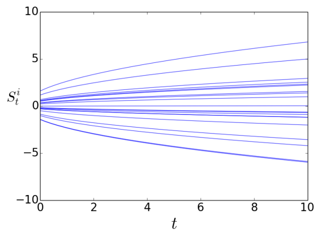
Objective: construct $\bar{X}_t \sim \pi_t = N(0, 1+t)$

Two solutions:

(I) $\mathrm{d}X_t^i = \mathrm{d}B_t^i$



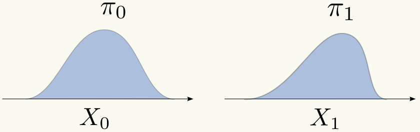
(II) $\dfrac{\mathrm{d}}{\mathrm{d}t} X_t^i = \dfrac{X_t^i}{\frac{2}{N}\sum_{j=1}^N (X_t^j)^2}$



As $N \to \infty$, they have the same marginal distribution π_t

However, different couplings between two time instants

Optimal transport construction



- optimal transport coupling

$$\min_{\text{coupling}} \ \mathsf{E}\left[|X_0 - X_1|^2\right] \quad \text{s.t.} \quad X_0 \sim \pi_0, \quad X_1 \sim \pi_1$$

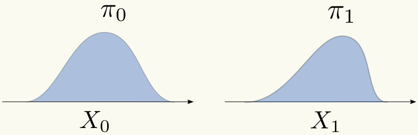
- Brenier's theorem: optimal coupling is deterministic and of gradient form

$$X_1 = \nabla\Phi(X_0)$$

- adapt the procedure to continuous-time filtering setup $\rightarrow$ feedback particle filter

C. Villani, Topics in optimal transportation, American Mathematical Soc., 2003

Optimal transport construction



- optimal transport coupling

$$\min_{\text{coupling}} \ \mathsf{E}\left[|X_0 - X_1|^2\right] \quad \text{s.t.} \quad X_0 \sim \pi_0, \quad X_1 \sim \pi_1$$

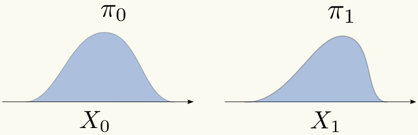
- Brenier's theorem: optimal coupling is deterministic and of gradient form

$$X_1 = \nabla\Phi(X_0)$$

- adapt the procedure to continuous-time filtering setup $\rightarrow$ feedback particle filter

C. Villani, Topics in optimal transportation, American Mathematical Soc., 2003

Optimal transport construction



- optimal transport coupling

$$\min_{\text{coupling}} \quad \mathsf{E}\left[|X_0 - X_1|^2\right] \quad \text{s.t.} \quad X_0 \sim \pi_0, \quad X_1 \sim \pi_1$$

- Brenier's theorem: optimal coupling is deterministic and of gradient form

$$X_1 = \nabla\Phi(X_0)$$

- adapt the procedure to continuous-time filtering setup $\rightarrow$ feedback particle filter

C. Villani, Topics in optimal transportation, American Mathematical Soc., 2003

Feedback particle filter (FPF)

State process: $\mathrm{d}X_t = \text{(dynamic model)} \qquad X_0 \sim \pi_0(\cdot)$

Observation process: $\mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t$

Objective: construct $\bar{X}_t \sim \pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]})$

Mean-field process:

$$\mathrm{d}\bar{X}_t = \text{(dynamic model)} + \underbrace{\mathsf{K}_t(\bar{X}_t) \circ (\mathrm{d}Z_t - \frac{h(\bar{X}_t) + \hat{h}_t}{2}\mathrm{d}t)}_{\text{correction}}, \quad \bar{X}_0 \overset{\text{i.i.d}}{\sim} \pi_0$$

- Gain function $\mathsf{K}_t(x) = \nabla \phi_t(x)$ where ϕ_t solves the Poisson eq.

$$\frac{1}{\rho_t(x)} \nabla \cdot (\rho_t(x) \nabla \phi_t(x)) = h(x) - \hat{h}_t$$

- $\hat{h}_t = \mathsf{E}[h(\bar{X}_t) | \mathcal{Z}_t]$
- ρ_t is probability density for $\bar{X}_t$

Feedback particle filter (FPF)

$$\begin{aligned} \text{State process:} \quad & \mathrm{d}X_t = \text{(dynamic model)} \qquad X_0 \sim \pi_0(\cdot) \\ \text{Observation process:} \quad & \mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t \\ \text{Objective:} \quad & \text{construct } \bar{X}_t \sim \pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]}) \end{aligned}$$

Mean-field process:

$$\mathrm{d}\bar{X}_t = \text{(dynamic model)} + \underbrace{\mathsf{K}_t(\bar{X}_t) \circ (\mathrm{d}Z_t - \frac{h(\bar{X}_t) + \hat{h}_t}{2}\mathrm{d}t)}_{\text{correction}}, \quad \bar{X}_0 \overset{\text{i.i.d}}{\sim} \pi_0$$

- Gain function $\mathsf{K}_t(x) = \nabla \phi_t(x)$ where ϕ_t solves the Poisson eq.

$$\frac{1}{\rho_t(x)} \nabla \cdot (\rho_t(x) \nabla \phi_t(x)) = h(x) - \hat{h}_t$$

- $\hat{h}_t = \mathsf{E}[h(\bar{X}_t)|\mathcal{Z}_t]$
- ρ_t is probability density for $\bar{X}_t$

Feedback particle filter (FPF)

State process: $\mathrm{d}X_t = (\text{dynamic model}) \qquad X_0 \sim \pi_0(\cdot)$

Observation process: $\mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t$

Objective: construct $\bar{X}_t \sim \pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]})$

Mean-field process:

$$\mathrm{d}\bar{X}_t = (\text{dynamic model}) + \underbrace{\mathsf{K}_t(\bar{X}_t) \circ (\mathrm{d}Z_t - \frac{h(\bar{X}_t) + \hat{h}_t}{2}\mathrm{d}t)}_{\text{correction}}, \quad \bar{X}_0 \overset{\text{i.i.d}}{\sim} \pi_0$$

- Gain function $\mathsf{K}_t(x) = \nabla\phi_t(x)$ where ϕ_t solves the Poisson eq.

$$\frac{1}{\rho_t(x)}\nabla \cdot (\rho_t(x)\nabla\phi_t(x)) = h(x) - \hat{h}_t$$

- $\hat{h}_t = \mathsf{E}[h(\bar{X}_t)|\mathcal{Z}_t]$
- ρ_t is probability density for $\bar{X}_t$

Feedback particle filter (FPF)

State process: $\mathrm{d}X_t = \text{(dynamic model)} \qquad X_0 \sim \pi_0(\cdot)$

Observation process: $\mathrm{d}Z_t = h(X_t)\mathrm{d}t + \mathrm{d}W_t$

Objective: construct $\bar{X}_t \sim \pi_t(\cdot) := \mathsf{P}(X_t \in \cdot | Z_{[0,t]})$

Mean-field process:

$$\mathrm{d}\bar{X}_t = \text{(dynamic model)} + \underbrace{\mathsf{K}_t(\bar{X}_t) \circ (\mathrm{d}Z_t - \frac{h(\bar{X}_t) + \hat{h}_t}{2}\mathrm{d}t)}_{\text{correction}}, \quad \bar{X}_0 \overset{\text{i.i.d}}{\sim} \pi_0$$

- Gain function $\mathsf{K}_t(x) = \nabla \phi_t(x)$ where ϕ_t solves the Poisson eq.

$$\frac{1}{\rho_t(x)} \nabla \cdot (\rho_t(x) \nabla \phi_t(x)) = h(x) - \hat{h}_t$$

- $\hat{h}_t = \mathsf{E}[h(\bar{X}_t)|\mathcal{Z}_t]$
- ρ_t is probability density for $\bar{X}_t$

Finite-N approximation

Update formula:

$$\mathrm{d}X_t^i = \text{(dynamic model)} + \underbrace{\mathsf{K}_t^{(N)}(X_t^i) \circ (\mathrm{d}Z_t - \frac{h(X_t^i) + \hat{h}_t^{(N)}}{2}\mathrm{d}t)}_{\text{correction}}, \quad \bar{X}_0 \overset{\text{i.i.d}}{\sim} \pi_0$$

- Gain function $\mathsf{K}_t^{(N)}(x)$ is approximated in terms of particles

$$\mathsf{K}_t^{(N)} = \text{Algorithm}(\{X_t^1, \ldots, X_t^N\}, h)$$

- $\hat{h}_t^{(N)} = \frac{1}{N}\sum_{i=1}^{N} h(X_t^i)$

Main challenge: Gain function approximation

Finite-N approximation

Update formula:

$$\mathrm{d}X_t^i = \text{(dynamic model)} + \underbrace{\mathsf{K}_t^{(N)}(X_t^i) \circ (\mathrm{d}Z_t - \frac{h(X_t^i) + \hat{h}_t^{(N)}}{2}\mathrm{d}t)}_{\text{correction}}, \quad \bar{X}_0 \overset{\text{i.i.d}}{\sim} \pi_0$$

- Gain function $\mathsf{K}_t^{(N)}(x)$ is approximated in terms of particles

$$\mathsf{K}_t^{(N)} = \text{Algorithm}(\{X_t^1, \ldots, X_t^N\}, h)$$

- $\hat{h}_t^{(N)} = \frac{1}{N}\sum_{i=1}^{N} h(X_t^i)$

Main challenge: Gain function approximation

Finite-N approximation

Update formula:

$$\mathrm{d}X_t^i = \text{(dynamic model)} + \underbrace{\mathsf{K}_t^{(N)}(X_t^i) \circ (\mathrm{d}Z_t - \frac{h(X_t^i) + \hat{h}_t^{(N)}}{2}\mathrm{d}t)}_{\text{correction}}, \quad \bar{X}_0 \overset{\text{i.i.d}}{\sim} \pi_0$$

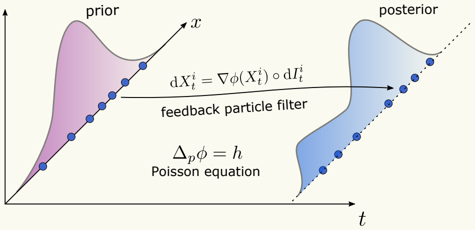
- Gain function $\mathsf{K}_t^{(N)}(x)$ is approximated in terms of particles

$$\mathsf{K}_t^{(N)} = \text{Algorithm}(\{X_t^1, \ldots, X_t^N\}, h)$$

- $\hat{h}_t^{(N)} = \dfrac{1}{N}\displaystyle\sum_{i=1}^{N} h(X_t^i)$

Main challenge: Gain function approximation

Part II: summary



KF: $\mathrm{d}m_t = (\text{dynamical model}) + \mathsf{K}_t(\mathrm{d}Z_t - Hm_t\mathrm{d}t)$

EnKF: $\mathrm{d}X_t^i = (\text{dynamical model}) + \mathsf{K}_t^{(N)}(\mathrm{d}Z_t - \frac{HX_t^i + Hm_t^{(N)}}{2}\mathrm{d}t)$

FPF: $\mathrm{d}X_t^i = (\text{dynamical model}) + \mathsf{K}_t^{(N)}(X_t^i) \circ (\mathrm{d}Z_t - \frac{h(X_t^i) + \hat{h}_t^{(N)}}{2}\mathrm{d}t)$

$\boxed{\text{Difficulty of filtering} = \text{gain function approximating}}$

Outline

Part I: background and motivation

Part II: mean-field control design

Part III: gain function approximation

Part III: applications

References:

- A. Taghvaei, P. G. Mehta, S. P. Meyn, Diffusion map-based algorithm for gain function approximation in the feedback particle filter, SIAM Journal on Uncertainty Quantification, 8(3):1090–1117, 2020

Outline

Part I: background and motivation

Part II: mean-field control design

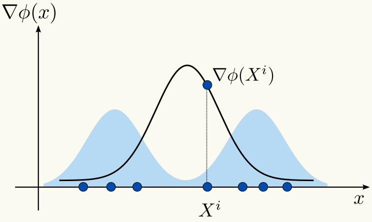
Part III: gain function approximation

Part III: applications

References:

- A. Taghvaei, P. G. Mehta, S. P. Meyn, Diffusion map-based algorithm for gain function approximation in the feedback particle filter, SIAM Journal on Uncertainty Quantification, 8(3):1090–1117, 2020

Gain function approximation ($\mathsf{K}(x) = \nabla\phi(x)$)
Problem formulation



Poisson equation:

$$-\Delta_\rho \phi(x) = h(x) - \hat{h}$$

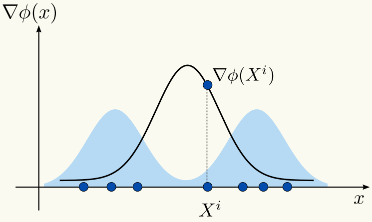
where $\Delta_\rho \phi := \dfrac{1}{\rho}\nabla \cdot (\rho \nabla \phi)$

Computational problem:

Given: $\{X^1, \ldots, X^N\} \stackrel{\text{i.i.d}}{\sim} \rho$

Approximate: $\{\nabla\phi(X^1), \ldots, \nabla\phi(X^N)\}$

Gain function approximation ($\mathsf{K}(x) = \nabla\phi(x)$)
Problem formulation



Poisson equation:

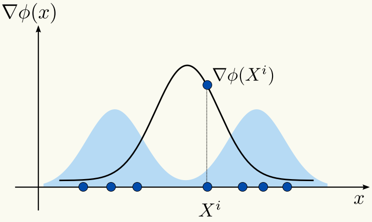
$$-\Delta_\rho \phi(x) = h(x) - \hat{h}$$

where $\Delta_\rho \phi := \Delta\phi + \nabla \log(\rho) \cdot \nabla\phi$

Computational problem:

Given: $\{X^1, \ldots, X^N\} \overset{\text{i.i.d}}{\sim} \rho$

Approximate: $\{\nabla\phi(X^1), \ldots, \nabla\phi(X^N)\}$



Poisson equation:

$$-\Delta_\rho \phi(x) = h(x) - \hat{h}$$

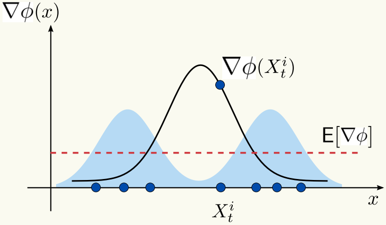
where $\Delta_\rho \phi := \dfrac{1}{\rho}\nabla\cdot(\rho\nabla\phi)$

Computational problem:

$$\begin{aligned}\text{Given:}\quad & \{X^1,\ldots,X^N\} \overset{\text{i.i.d}}{\sim} \rho \\ \text{Approximate:}\quad & \{\nabla\phi(X^1),\ldots,\nabla\phi(X^N)\}\end{aligned}$$

Gain function approximation
Constant gain approximation

$$\mathsf{K}_{\text{const}} := \arg\min_{\mathsf{K}\in\mathbb{R}^d} \int |\nabla\phi(x) - \mathsf{K}|^2 \rho(x)\mathrm{d}x$$



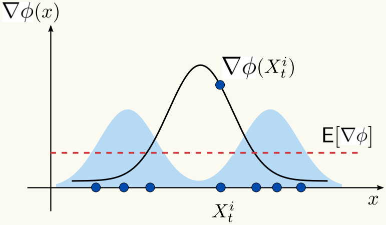
A closed-form formula:

$$\mathsf{K}_{\text{const}} = \int (h(x) - \hat{h}) x \rho(x)\mathrm{d}x \approx \frac{1}{N}\sum_{i=1} (h(X^i) - \hat{h}^{(N)}) X^i$$

- FPF $\xrightarrow{\text{const. gain approx.}}$ Ensemble Kalman filter (EnKF)
- Can we improve this approximation?

A. Taghvaei, J de Wiljes, P. G. Mehta, and S. Reich. Kalman filter and its modern extensions... ASME, 2017

$$\mathsf{K}_{\mathsf{const}} := \arg\min_{\mathsf{K}\in\mathbb{R}^d} \int |\nabla\phi(x) - \mathsf{K}|^2 \rho(x)\mathrm{d}x$$



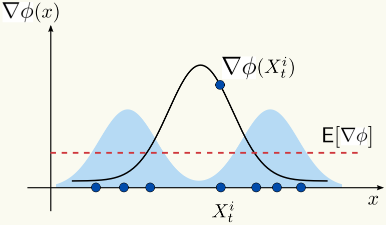
A closed-form formula:

$$\mathsf{K}_{\mathsf{const}} = \int (h(x) - \hat{h}) x \rho(x)\mathrm{d}x \approx \frac{1}{N}\sum_{i=1}(h(X^i) - \hat{h}^{(N)})X^i$$

- FPF $\xrightarrow{\text{const. gain approx.}}$ Ensemble Kalman filter (EnKF)
- Can we improve this approximation?

A. Taghvaei, J de Wiljes, P. G. Mehta, and S. Reich. Kalman filter and its modern extensions... ASME, 2017

$$\mathsf{K}_{\text{const}} := \arg\min_{\mathsf{K}\in\mathbb{R}^d} \int |\nabla\phi(x) - \mathsf{K}|^2 \rho(x)\mathrm{d}x$$



A closed-form formula:

$$\mathsf{K}_{\text{const}} = \int (h(x) - \hat{h}) x \rho(x)\mathrm{d}x \approx \frac{1}{N}\sum_{i=1} (h(X^i) - \hat{h}^{(N)})X^i$$

- FPF $\xrightarrow{\text{const. gain approx.}}$ Ensemble Kalman filter (EnKF)
- Can we improve this approximation?

A. Taghvaei, J de Wiljes, P. G. Mehta, and S. Reich. Kalman filter and its modern extensions... ASME, 2017

Gain function approximation
Three formulations, Three algorithms

$$-\Delta_\rho \phi = h - \hat{h}$$

(I) Weak formulation: (Galerkin algorithm)

$$\langle \nabla\phi, \nabla\psi \rangle = \langle h - \hat{h}, \psi \rangle, \quad \forall \psi \in H^1(\rho)$$

where $\langle f, g \rangle := \int f(x)g(x)\rho(x)\mathrm{d}x$

(II) Semigroup formulation: (Diffusion-map algorithm)

$$\phi = P_t\phi + \int_0^t P_s(h - \hat{h})\mathrm{d}s$$

where $P_t := e^{t\Delta_\rho}$ is the semigroup

(III) Variational formulation: (Neural net. approximation)

$$\min_{\phi \in H_0^1(\rho)} \int \left[\frac{1}{2}|\nabla\phi(x)|^2 - \phi(x)(h(x) - \hat{h}) \right] \rho(x)\mathrm{d}x$$

$$-\Delta_\rho \phi = h - \hat{h}$$

(I) Weak formulation: (Galerkin algorithm)

$$\langle \nabla\phi, \nabla\psi \rangle = \langle h - \hat{h}, \psi \rangle, \quad \forall \psi \in H^1(\rho)$$

where $\langle f, g \rangle := \int f(x)g(x)\rho(x)\mathrm{d}x$

(II) Semigroup formulation: (Diffusion-map algorithm)

$$\phi = P_t\phi + \int_0^t P_s(h - \hat{h})\mathrm{d}s$$

where $P_t := e^{t\Delta_\rho}$ is the semigroup

(III) Variational formulation: (Neural net. approximation)

$$\min_{\phi \in H_0^1(\rho)} \int \left[\frac{1}{2}|\nabla\phi(x)|^2 - \phi(x)(h(x) - \hat{h}) \right] \rho(x)\mathrm{d}x$$

$$-\Delta_\rho \phi = h - \hat{h}$$

(I) Weak formulation: (Galerkin algorithm)

$$\langle \nabla\phi, \nabla\psi \rangle = \langle h - \hat{h}, \psi \rangle, \quad \forall \psi \in H^1(\rho)$$

where $\langle f, g \rangle := \int f(x)g(x)\rho(x)\mathrm{d}x$

(II) Semigroup formulation: (Diffusion-map algorithm)

$$\phi = P_t\phi + \int_0^t P_s(h - \hat{h})\mathrm{d}s$$

where $P_t := e^{t\Delta_\rho}$ is the semigroup

(III) Variational formulation: (Neural net. approximation)

$$\min_{\phi \in H_0^1(\rho)} \int \left[\frac{1}{2}|\nabla\phi(x)|^2 - \phi(x)(h(x) - \hat{h}) \right] \rho(x)\mathrm{d}x$$

$$-\Delta_\rho \phi = h - \hat{h}$$

(I) Weak formulation: (Galerkin algorithm)

$$\langle \nabla\phi, \nabla\psi \rangle = \langle h - \hat{h}, \psi \rangle, \quad \forall \psi \in H^1(\rho)$$

where $\langle f, g \rangle := \int f(x) g(x) \rho(x) \mathrm{d}x$

(II) Semigroup formulation: (Diffusion-map algorithm)

$$\phi = P_t \phi + \int_0^t P_s (h - \hat{h}) \mathrm{d}s$$

where $P_t := e^{t\Delta_\rho}$ is the semigroup

(III) Variational formulation: (Neural net. approximation)

$$\min_{\phi \in H_0^1(\rho)} \int \left[\frac{1}{2} |\nabla \phi(x)|^2 - \phi(x)(h(x) - \hat{h}) \right] \rho(x) \mathrm{d}x$$

$$-\Delta_\rho \phi = h - \hat{h}$$

(I) Weak formulation: (Galerkin algorithm)

$$\langle \nabla \phi, \nabla \psi \rangle = \langle h - \hat{h}, \psi \rangle, \quad \forall \psi \in H^1(\rho)$$

where $\langle f, g \rangle := \int f(x) g(x) \rho(x) \mathrm{d}x$

(II) Semigroup formulation: (Diffusion-map algorithm)

$$\phi = P_t \phi + \int_0^t P_s (h - \hat{h}) \mathrm{d}s$$

where $P_t := e^{t\Delta_\rho}$ is the semigroup

(III) Variational formulation: (Neural net. approximation)

$$\min_{\phi \in H_0^1(\rho)} \int \left[\frac{1}{2} |\nabla \phi(x)|^2 - \phi(x)(h(x) - \hat{h}) \right] \rho(x) \mathrm{d}x$$

$$-\Delta_\rho \phi = h - \hat{h}$$

(I) Weak formulation: (Galerkin algorithm)

$$\langle \nabla\phi, \nabla\psi \rangle = \langle h - \hat{h}, \psi \rangle, \quad \forall \psi \in H^1(\rho)$$

where $\langle f, g \rangle := \int f(x)g(x)\rho(x)\mathrm{d}x$

(II) Semigroup formulation: (Diffusion-map algorithm)

$$\phi = P_t\phi + \int_0^t P_s(h - \hat{h})\mathrm{d}s$$

where $P_t := e^{t\Delta_\rho}$ is the semigroup

(III) Variational formulation: (Neural net. approximation)

$$\min_{\phi \in H_0^1(\rho)} \int \left[\frac{1}{2}|\nabla\phi(x)|^2 - \phi(x)(h(x) - \hat{h}) \right] \rho(x)\mathrm{d}x$$

$$-\Delta_\rho \phi = h - \hat{h}$$

(I) Weak formulation: (Galerkin algorithm)

$$\langle \nabla\phi, \nabla\psi \rangle = \langle h - \hat{h}, \psi \rangle, \quad \forall \psi \in H^1(\rho)$$

where $\langle f, g \rangle := \int f(x)g(x)\rho(x)\mathrm{d}x$

(II) Semigroup formulation: (Diffusion-map algorithm)

$$\phi = P_t\phi + \int_0^t P_s(h - \hat{h})\mathrm{d}s$$

where $P_t := e^{t\Delta_\rho}$ is the semigroup

(III) Variational formulation: (Neural net. approximation)

$$\min_{\phi \in H_0^1(\rho)} \int \left[\frac{1}{2}|\nabla\phi(x)|^2 - \phi(x)(h(x) - \hat{h}) \right] \rho(x)\mathrm{d}x$$

(I) Semigroup formulation:

$$\phi = P_\epsilon \phi + \int_0^\epsilon P_s(h - \hat{h})\mathrm{d}s$$

where $P_\epsilon := e^{\epsilon \Delta_\rho}$ is the semigroup

(II) Diffusion map approximation: (Coifman, et. al. 2006)

$$P_\epsilon f(x) \overset{\epsilon\downarrow 0}{\approx} T_\epsilon f(x) := \frac{1}{n_\epsilon(x)} \int g_\epsilon(x,y) \frac{f(y)\rho(y)}{\sqrt{(g_\epsilon * \rho)(y)}} \mathrm{d}y$$

(III) Empirical approximation :

$$T_\epsilon f(x) \overset{N\uparrow\infty}{\approx} T_\epsilon^{(N)} f(x) := \frac{1}{n_\epsilon^{(N)}(x)} \sum_{i=1}^N g_\epsilon(x, X^i) \frac{f(X^i)}{\sqrt{\sum_{j=1}^N g_\epsilon(X^i, X^j)}}$$

A. Taghvaei, P. G. Mehta, S. P. Meyn Diffusion map-based algorithm for gain function approximation in the feedback particle filter, SIAM UQ 2020

(I) Semigroup formulation:

$$\phi = P_\epsilon \phi + \int_0^\epsilon P_s(h - \hat{h})\mathrm{d}s$$

where $P_\epsilon := e^{\epsilon \Delta_\rho}$ is the semigroup

(II) Diffusion map approximation: (Coifman, et. al. 2006)

$$P_\epsilon f(x) \overset{\epsilon \downarrow 0}{\approx} T_\epsilon f(x) := \frac{1}{n_\epsilon(x)} \int g_\epsilon(x,y) \frac{f(y)\rho(y)}{\sqrt{(g_\epsilon * \rho)(y)}} \mathrm{d}y$$

(III) Empirical approximation :

$$T_\epsilon f(x) \overset{N \uparrow \infty}{\approx} T_\epsilon^{(N)} f(x) := \frac{1}{n_\epsilon^{(N)}(x)} \sum_{i=1}^{N} g_\epsilon(x, X^i) \frac{f(X^i)}{\sqrt{\sum_{j=1}^{N} g_\epsilon(X^i, X^j)}}$$

A. Taghvaei, P. G. Mehta, S. P. Meyn Diffusion map-based algorithm for gain function approximation in the feedback particle filter, SIAM UQ 2020

(I) Semigroup formulation:

$$\phi = P_\epsilon \phi + \int_0^\epsilon P_s(h - \hat{h})\mathrm{d}s$$

where $P_\epsilon := e^{\epsilon \Delta_\rho}$ is the semigroup

(II) Diffusion map approximation: (Coifman, et. al. 2006)

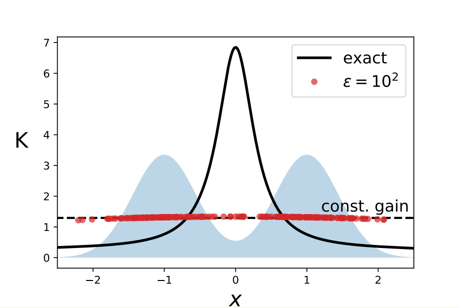
$$P_\epsilon f(x) \overset{\epsilon \downarrow 0}{\approx} T_\epsilon f(x) := \frac{1}{n_\epsilon(x)} \int g_\epsilon(x,y) \frac{f(y)\rho(y)}{\sqrt{(g_\epsilon * \rho)(y)}} \mathrm{d}y$$

(III) Empirical approximation :

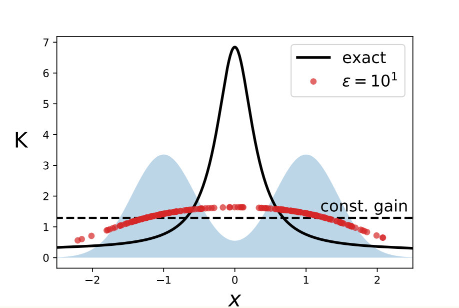
$$T_\epsilon f(x) \overset{N\uparrow\infty}{\approx} T_\epsilon^{(N)} f(x) := \frac{1}{n_\epsilon^{(N)}(x)} \sum_{i=1}^N g_\epsilon(x, X^i) \frac{f(X^i)}{\sqrt{\sum_{j=1}^N g_\epsilon(X^i, X^j)}}$$

A. Taghvaei, P. G. Mehta, S. P. Meyn Diffusion map-based algorithm for gain function approximation in the feedback particle filter, SIAM UQ 2020

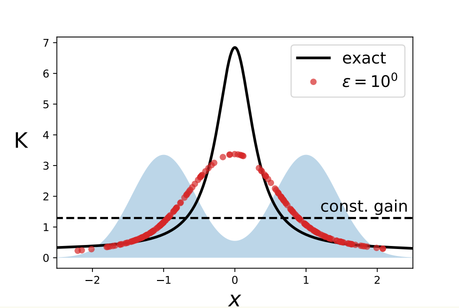
Diffusion map-based algorithm
Numerical analysis



Objective: error analysis of bias and variance

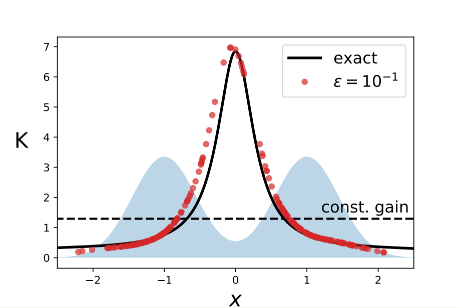


Objective: error analysis of bias and variance



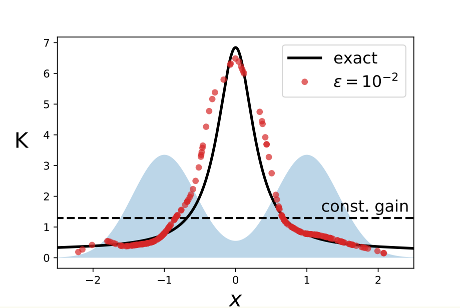
Objective: error analysis of bias and variance

Diffusion map-based algorithm
Numerical analysis



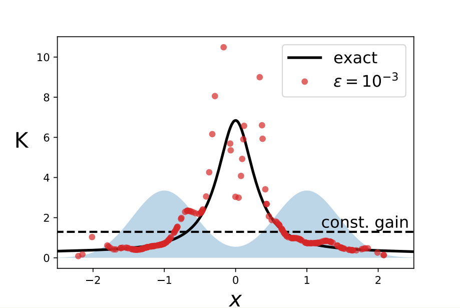
Objective: error analysis of bias and variance

Diffusion map-based algorithm
Numerical analysis



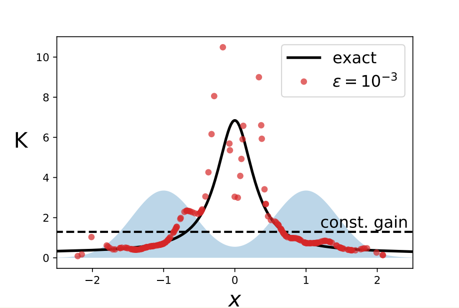
Objective: error analysis of bias and variance

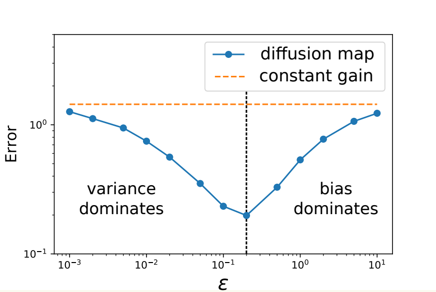
Diffusion map-based algorithm
Numerical analysis



Objective: error analysis of bias and variance

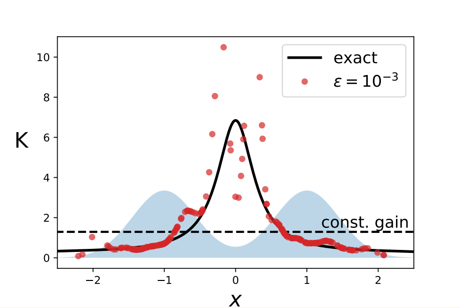
Diffusion map-based algorithm
Numerical analysis

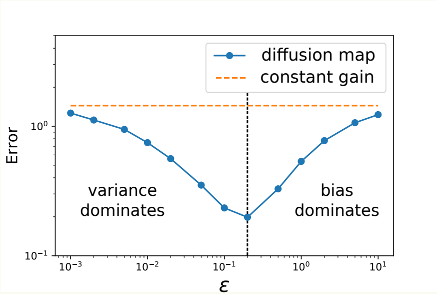




Objective: error analysis of bias and variance

Diffusion map-based algorithm
Numerical analysis





Objective: error analysis of bias and variance

Approximation steps:

$$\phi \xrightarrow[\text{bias}]{\text{DM approx.}} \phi_\epsilon \xrightarrow[\text{variance}]{\text{empirical approx.}} \phi_\epsilon^{(N)}$$

Proposition

Under technical assumptions, with high probability

$$\|\phi_\epsilon^{(N)} - \phi\|_{L^2(\rho)} \leq \underbrace{O(\epsilon)}_{\text{bias}} + \underbrace{O(\frac{1}{\epsilon^{1+\frac{d}{2}} N^{\frac{1}{2}}})}_{\text{variance}}$$

Tools for proof:

1. stochastic stability theory (Meyn & Tweedie, 2012)
2. numerical analysis of integral equations (Anselone, 1971, Atkinson, 1976)

A. Taghvaei, P. G. Mehta, S. P. Meyn Diffusion map-based algorithm for gain function approximation in the feedback particle filter, SIAM UQ 2020

Approximation steps:

$$\phi \xrightarrow[\text{bias}]{\text{DM approx.}} \phi_\epsilon \xrightarrow[\text{variance}]{\text{empirical approx.}} \phi_\epsilon^{(N)}$$

Proposition

Under technical assumptions, with high probability

$$\|\phi_\epsilon^{(N)} - \phi\|_{L^2(\rho)} \leq \underbrace{O(\epsilon)}_{\text{bias}} + \underbrace{O(\frac{1}{\epsilon^{1+\frac{d}{2}} N^{\frac{1}{2}}})}_{\text{variance}}$$

Tools for proof:

1. stochastic stability theory (Meyn & Tweedie, 2012)
2. numerical analysis of integral equations (Anselone, 1971, Atkinson, 1976)

A. Taghvaei, P. G. Mehta, S. P. Meyn Diffusion map-based algorithm for gain function approximation in the feedback particle filter, SIAM UQ 2020

Approximation steps:

$$\phi \xrightarrow[\text{bias}]{\text{DM approx.}} \phi_\epsilon \xrightarrow[\text{variance}]{\text{empirical approx.}} \phi_\epsilon^{(N)}$$

Proposition

Under technical assumptions, with high probability

$$\|\phi_\epsilon^{(N)} - \phi\|_{L^2(\rho)} \leq \underbrace{O(\epsilon)}_{\text{bias}} + \underbrace{O(\frac{1}{\epsilon^{1+\frac{d}{2}} N^{\frac{1}{2}}})}_{\text{variance}}$$

Tools for proof:

1. stochastic stability theory (Meyn & Tweedie, 2012)
2. numerical analysis of integral equations (Anselone, 1971, Atkinson, 1976)

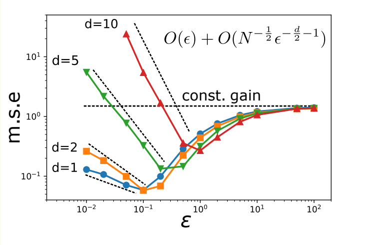
A. Taghvaei, P. G. Mehta, S. P. Meyn Diffusion map-based algorithm for gain function approximation in the feedback particle filter, SIAM UQ 2020

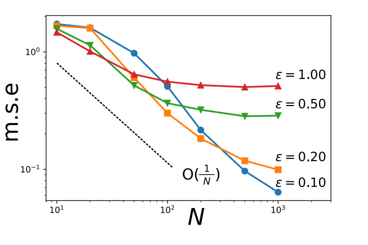
Setup: $\rho(x) = \rho_{\mathsf{bimodal}}(x_1) \prod_{n=2}^{d} \rho_{\mathsf{Gaussian}}(x_n)$ and $h(x) = x_1$.

Convergence to const. gain approx.:

$$\lim_{\epsilon \to \infty} \mathsf{K}_\epsilon^{(N)} = \text{const. gain approximatoin}$$

Setup: $\rho(x) = \rho_{\text{bimodal}}(x_1) \prod_{n=2}^{d} \rho_{\text{Gaussian}}(x_n)$ and $h(x) = x_1$.

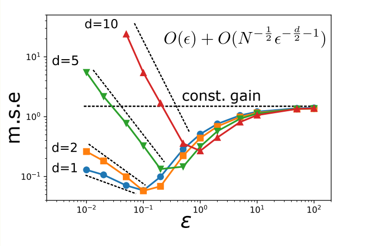


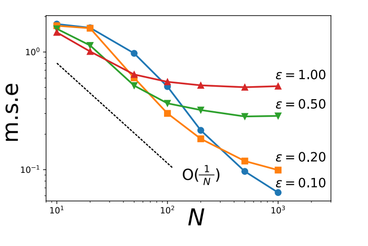


Convergence to const. gain approx.:

$$\lim_{\epsilon\to\infty} \mathsf{K}_\epsilon^{(N)} = \text{const. gain approximatoin}$$

Setup: $\rho(x) = \rho_{\text{bimodal}}(x_1) \prod_{n=2}^{d} \rho_{\text{Gaussian}}(x_n)$ and $h(x) = x_1$.





Convergence to const. gain approx.:

$$\lim_{\epsilon \to \infty} \mathsf{K}_\epsilon^{(N)} = \text{const. gain approximatoin}$$

Outline

Part I: background and motivation

Part II: mean-field control design

Part III: gain function approximation

Part III: applications

References:

- C. Zhang, A. Taghvaei, P. G. Mehta, Feedback Particle Filter on Riemannian Manifolds and Matrix Lie groups, IEEE Transactions on Automatic Control (TAC), 63(8):2465–2480, 2017.
- C. Zhang, A. Taghvaei, P. G. Mehta, Attitude Estimation of a Wearable Motion Sensor IEEE American Control Conference (ACC), 2017
- A. Taghvaei, S. A. Hutchinson, P. G. Mehta, A Coupled Oscillator-based Control Architecture for Locomotory Gaits, IEEE Conference on Decision and Control (CDC), 2014

Outline

Part I: background and motivation

Part II: mean-field control design

Part III: gain function approximation

Part III: applications

References:

- C. Zhang, A. Taghvaei, P. G. Mehta, Feedback Particle Filter on Riemannian Manifolds and Matrix Lie groups, IEEE Transactions on Automatic Control (TAC), 63(8):2465–2480, 2017.
- C. Zhang, A. Taghvaei, P. G. Mehta, Attitude Estimation of a Wearable Motion Sensor IEEE American Control Conference (ACC), 2017
- A. Taghvaei, S. A. Hutchinson, P. G. Mehta, A Coupled Oscillator-based Control Architecture for Locomotory Gaits, IEEE Conference on Decision and Control (CDC), 2014

Model:

$$\text{State} \quad \mathrm{d}R_t = R_t[\,\omega_t\,]_\times \mathrm{d}t + R_t \circ [\,\sigma_B \mathrm{d}B_t\,]_\times \qquad R_0 \sim \pi_0$$

$$\text{Observation} \quad \mathrm{d}Z_t = h(R_t)\mathrm{d}t + \sigma_W \mathrm{d}W_t$$

FPF:

$$\mathrm{d}R_t^i = (\text{dynamaics}) + \underbrace{R_t^i\,[\,\mathsf{K}_t(R_t^i) \circ (\mathrm{d}Z_t - \frac{1}{2}\,(h(R_t^i) + \hat{h}_t)\,\mathrm{d}t)\,]_\times}_{\text{correction}}$$

- gain function $\mathsf{K}_t(R) = \text{grad}(\phi)(R)$
- ϕ satisfies the Poisson equation on $SO(3)$
- diffusion map algorithm is extended to $SO(3)$

C. Zhang, A. Taghvaei, P. G. Mehta Feedback Particle Filter on Riemannian Manifolds and Matrix Lie groups, IEEE TAC, 2017

Model:

State $\mathrm{d}R_t = R_t[\,\omega_t\,]_\times \mathrm{d}t + R_t \circ [\,\sigma_B \mathrm{d}B_t\,]_\times \qquad R_0 \sim \pi_0$

Observation $\mathrm{d}Z_t = h(R_t)\mathrm{d}t + \sigma_W \mathrm{d}W_t$

Notation $R_t \in SO(3), \quad Z_t \in \mathbb{R}^m$

FPF:

$$\mathrm{d}R_t^i = (\text{dynamaics}) + \underbrace{R_t^i\,[\,\mathsf{K}_t(R_t^i) \circ (\mathrm{d}Z_t - \frac{1}{2}\,(h(R_t^i) + \hat{h}_t)\,\mathrm{d}t)\,]_\times}_{\text{correction}}$$

- gain function $\mathsf{K}_t(R) = \text{grad}(\phi)(R)$
- ϕ satisfies the Poisson equation on $SO(3)$
- diffusion map algorithm is extended to $SO(3)$

C. Zhang, A. Taghvaei, P. G. Mehta Feedback Particle Filter on Riemannian Manifolds and Matrix Lie groups, IEEE TAC, 2017

Model:

State $\mathrm{d}R_t = R_t[\,\omega_t\,]_\times \mathrm{d}t + R_t \circ [\,\sigma_B \mathrm{d}B_t\,]_\times \qquad R_0 \sim \pi_0$

Observation $\mathrm{d}Z_t = h(R_t)\mathrm{d}t + \sigma_W \mathrm{d}W_t$

Notation $R_t \in SO(3),\quad Z_t \in \mathbb{R}^m$

$\omega_t \in \mathbb{R}^3$: angular velocity in body frame

FPF:

$$\mathrm{d}R_t^i = (\text{dynamaics}) + \underbrace{R_t^i\,[\,\mathsf{K}_t(R_t^i) \circ (\mathrm{d}Z_t - \frac{1}{2}\,(h(R_t^i) + \hat{h}_t)\,\mathrm{d}t)\,]_\times}_{\text{correction}}$$

- gain function $\mathsf{K}_t(R) = \mathrm{grad}(\phi)(R)$
- ϕ satisfies the Poisson equation on $SO(3)$
- diffusion map algorithm is extended to $SO(3)$

C. Zhang, A. Taghvaei, P. G. Mehta Feedback Particle Filter on Riemannian Manifolds and Matrix Lie groups, IEEE TAC, 2017

Model:

State $\mathrm{d}R_t = R_t[\,\omega_t\,]_\times \mathrm{d}t + R_t \circ [\,\sigma_B \mathrm{d}B_t\,]_\times \qquad R_0 \sim \pi_0$

Observation $\mathrm{d}Z_t = h(R_t)\mathrm{d}t + \sigma_W \mathrm{d}W_t$

Notation $R_t \in SO(3), \quad Z_t \in \mathbb{R}^m$

$\omega_t \in \mathbb{R}^3$: angular velocity in body frame

Applications:

- aircraft navigation (Crassidis et. al 03' [JGCD], Hua et. al 14' [IEEE TCST])
- robot localization (Hesch et. al 13' [IJRR], Kelly et. al 11' [IJRR])
- visual tracking (Choi et. al 11' [ICRA], Kwon et. al 13' [IEEE TPAMI])

FPF:

$$\mathrm{d}R_t^i = \text{(dynamaics)} + \underbrace{R_t^i\,[\,\mathsf{K}_t(R_t^i) \circ (\mathrm{d}Z_t - \frac{1}{2}\,(h(R_t^i) + \hat{h}_t)\,\mathrm{d}t)\,]_\times}_{\text{correction}}$$

- gain function $\mathsf{K}_t(R) = \mathrm{grad}(\phi)(R)$
- ϕ satisfies the Poisson equation on $SO(3)$

Model:

$$\text{State} \quad \mathrm{d}R_t = R_t[\,\omega_t\,]_\times \mathrm{d}t + R_t \circ [\,\sigma_B \mathrm{d}B_t\,]_\times \qquad R_0 \sim \pi_0$$

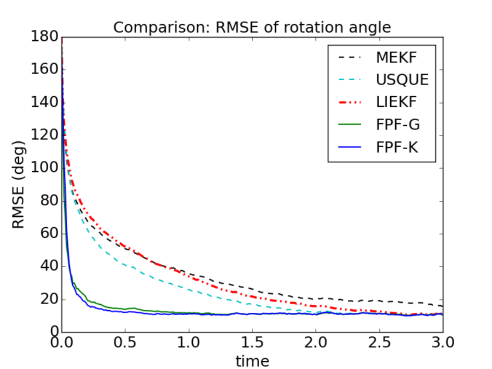
$$\text{Observation} \quad \mathrm{d}Z_t = h(R_t)\mathrm{d}t + \sigma_W \mathrm{d}W_t$$

FPF:

$$\mathrm{d}R_t^i = (\text{dynamaics}) + \underbrace{R_t^i\,[\,\mathsf{K}_t(R_t^i) \circ (\mathrm{d}Z_t - \frac{1}{2}\,(h(R_t^i) + \hat{h}_t)\,\mathrm{d}t)\,]_\times}_{\text{correction}}$$

- gain function $\mathsf{K}_t(R) = \text{grad}(\phi)(R)$
- ϕ satisfies the Poisson equation on $SO(3)$
- diffusion map algorithm is extended to $SO(3)$

C. Zhang, A. Taghvaei, P. G. Mehta Feedback Particle Filter on Riemannian Manifolds and Matrix Lie groups, IEEE TAC, 2017



Comparison with:

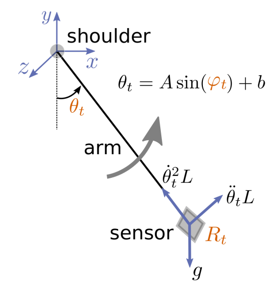
- MEKF: Multiplicative EKF (Markley 03')
- USQUE: Unscented Quaternion Estimator (Crassidis et. al 03')
- LIEKF: Left invariant EKF (Bonnabel et. al 09')

Performance metric: RMSE of rotation angle $\text{RMSE}_t = \sqrt{\frac{1}{100}\sum_{j=1}^{100}(\delta\theta_t^j)^2}$

Arm motion tracking

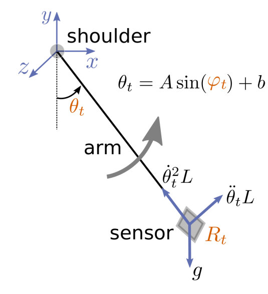
C. Zhang, A. Taghvaei, P. G. Mehta, Attitude Estimation of a Wearable Motion Sensor, ACC 2017

Arm motion tracking



C. Zhang, A. Taghvaei, P. G. Mehta, Attitude Estimation of a Wearable Motion Sensor, ACC 2017

Arm motion tracking

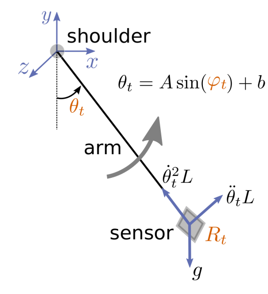


State: $(R_t, \varphi_t) \in \mathsf{SO}(3) \times \mathsf{SO}(2)$

Observation: $\mathrm{d}Z_t = R_t^T h(\varphi_t) + \sigma_W \mathrm{d}W_t$

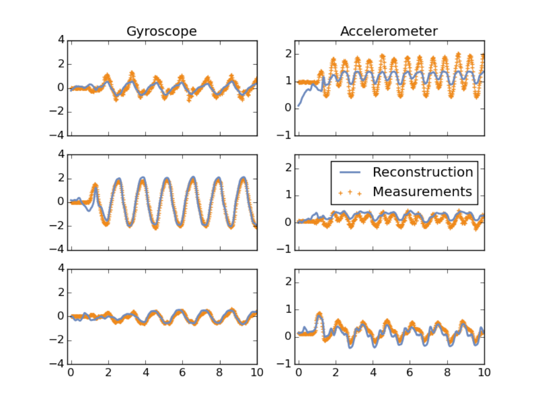
C. Zhang, A. Taghvaei, P. G. Mehta, Attitude Estimation of a Wearable Motion Sensor, ACC 2017

Arm motion tracking



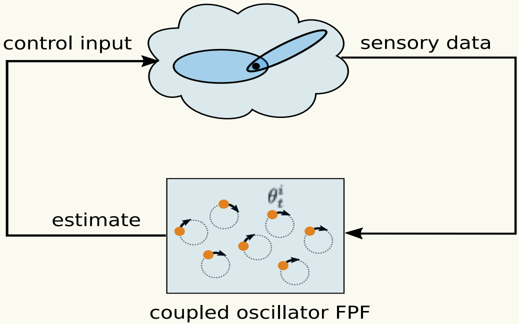
State: $(R_t, \varphi_t) \in \mathrm{SO}(3) \times \mathrm{SO}(2)$

Observation: $\mathrm{d}Z_t = R_t^T h(\varphi_t) + \sigma_W \mathrm{d}W_t$



C. Zhang, A. Taghvaei, P. G. Mehta, Attitude Estimation of a Wearable Motion Sensor, ACC 2017

Optimal control of locomotion gaits
2-body System

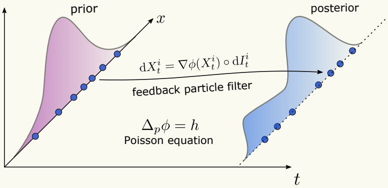


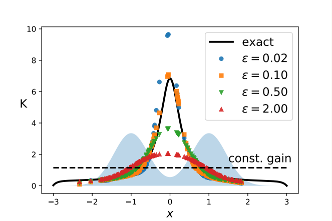
[Click to play the movie]

A. Taghvaei, S. Hutchinson and P. G. Mehta. A coupled-oscillators-based control architecture for locomotory gaits. IEEE CDC (2014).

Summary

Controlled interacting particle system:





mean-field control design

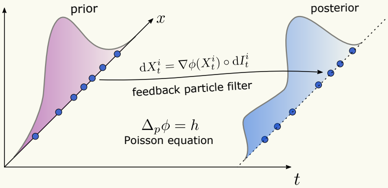
gain function approximation

Question: can the design and approximation be addressed in a single framework?

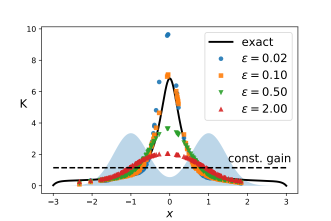
- some directions for future research

Summary

Controlled interacting particle system:



mean-field control design



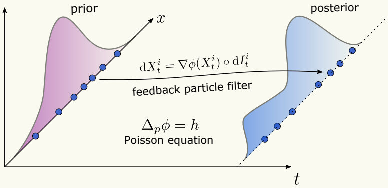
gain function approximation

Question: can the design and approximation be addressed in a single framework?

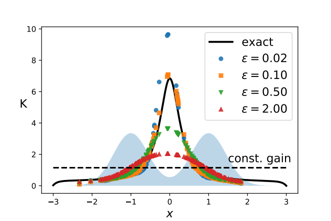
- some directions for future research

Summary

Controlled interacting particle system:



mean-field control design

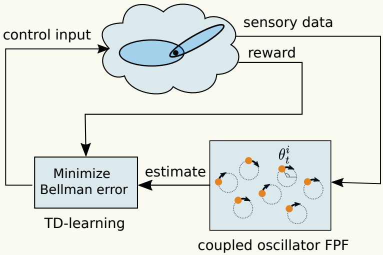


gain function approximation

Question: can the design and approximation be addressed in a single framework?

- some directions for future research

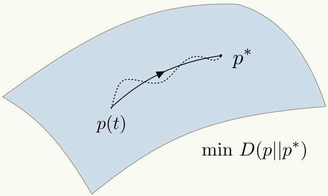
Reinforcement learning with partial observation



Idea: particles represent the belief state

Challenge: filtering is based on known dynamic and observation model

Interplay between optimization and sampling



MCMC

$$\mathrm{d}X_t^i = -\nabla V(X_t^i)\mathrm{d}t + \sqrt{2}\mathrm{d}B_t^i$$

Controlled interacting particle system

$$\frac{\mathrm{d}}{\mathrm{d}t}X_t^i = -\nabla V(X_t^i) + \underbrace{\nabla \log(p_t(X_t^i))}_{\text{interaction}}$$

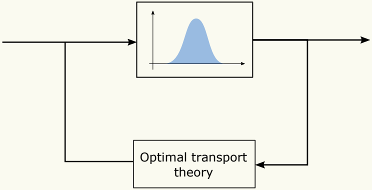
Example: Stein variational gradient descent (Liu & Wang, 2016)

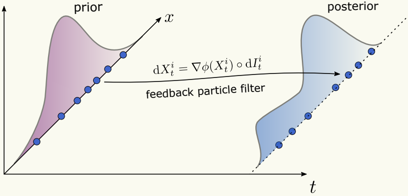
Questions:

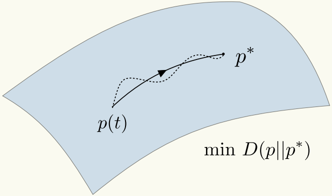
- Principled method to approximate the interaction term
- What are the fundamental differences and trade-offs between two approaches?

A. Taghvaei, P. G. Mehta. Accelerated flow for probability distributions, ICML 2019

The End







Thank you for your attention!

Questions?