Intro to Autonomous Driving: 2nd Circle

Bayesian Filter II: Extended Kalmen Filter, Unscented Kalmen Filter and Particle Filter

Jianan Liu

Gothenburg

July 23, 2018

Disclaimer

All opinions and statements in this presentation are mine and do not in any way represent the company
Any comment or correction of error is welcome, please contact me chisyliu@hotmail.com All slides are reproduced from Python Machine Learning Extra Book Circle

Acknowledgement

Acknowledgement

Here I would like to thank for Mengbai Tao, Ziqi Peng, Hui Wen from Baseband Software Division in Ericsson. The inspiration of writing the materials about Bayesian filters and 'relationship between Bayesian filter and Kalman filter' was actually coming from a question Ziqi asked us during one of the lunches all we had together, how many 'commonly used' so called filters there are? Although we are not working together nowadays, I wish Mengbai achieve new success as scrum master in Combitech, Hui Wen and Ziqi Peng good luck in UK and make lots of money in Amazon and Goldman Sachs

For My Friends

This is for two friends, happy marriage

Poetry

幽兰蕙心翠眉卧 桂香白首何需诺

望君娉婷不可说 笑面三刻话酒浊

Bayesian Filter II: Extended Kalmen Filter, Unscented Kalmen Filter and Particle Filter

In this presentation

- This slide will introduce other relative Bayesian filters(Extended Kalmen Filter, Unscented Kalmen Filter and Particle Filter) when the dynamic state model is NOT linear and/or NOT Gaussian
- The context in this circle is NOT included in the book
- If you want to know more about Bayesian filter, EKF, UKF, particle filter, etc., I recommend reading Simo Sarkka's book Bayesian Filtering and Smoothing and his course Nonlinear Filtering and Estimation, for tutorial of EKF UKF and particle filter Zhe Chen's paper "Bayesian Filtering: From Kalman Filters to Particle Filters, and Beyond" is a good resource
- We will try to go through every algorithm, also show the mathematics behind each algorithm.

Overview

1. Extended Kalman Filter
   - Revisit Kalman Filter
   - If System Changes from Linear to Nonlinear
   - EKF: (1st Order Taylor Expansion)Linearization of Nonlinear Dynamic State System in Gaussian Distribution
2. Unscented Kalman Filter
   - UKF: Better Way(2nd Order Taylor Expansion) to Linearization by Unscented Transform
3. Particle Filter
   - Sequential Importance Sampling
   - PF: Monte Carlo Method for Arbitrary Distribution State by Using Sequential Importance Sampling and Resampling
4. Conclusion

Outline for Section 1

Review of Kalman Filter

- We already known the environment where Kalman filter works well, is linear Gaussian dynamic state system model

Gaussian Linear (Markov) Dynamic State System Model

- $\mathbf{x}_t = \mathbf{F}_{t-1}\mathbf{x}_{t-1} + \mathbf{q}_{t-1}$
- $\mathbf{y}_t = \mathbf{H}_t\mathbf{x}_t + \mathbf{r}_t$
- $\mathbf{q}_{t-1} \sim \mathcal{N}(0, \mathbf{Q}_{t-1})$
- $\mathbf{r}_t \sim \mathcal{N}(0, \mathbf{R}_t)$

- The Kalman filter algorithm is shown as following page

Review of Kalman Filter

Kalman filter algorithm

Predication Step of Kalman Filter

- $\mathbf{x}_t^- = \mathbf{F}_{t-1}\mathbf{x}_{t-1}$
- $\mathbf{P}_t^- = \mathbf{F}_{t-1}\mathbf{P}_{t-1}\mathbf{F}_{t-1}^T + \mathbf{Q}_{t-1}$

Measurements Update Step of Kalman Filter

- $\mathbf{S}_t = \mathbf{H}_t\mathbf{P}_t^-\mathbf{H}_t^T + \mathbf{R}_t$
- $\mathbf{v}_t = \mathbf{y}_t - \mathbf{H}_t\mathbf{x}_t^-$
- $\mathbf{K}_t = \mathbf{H}_t\mathbf{P}_t^-(\mathbf{H}_t\mathbf{P}_t^-\mathbf{H}_t^T + \mathbf{R}_t)^{-1} = \mathbf{H}_t\mathbf{P}_t^-\mathbf{S}_t^{-1}$
- $\mathbf{x}_t = \mathbf{x}_t^- + \mathbf{K}_t\mathbf{v}_t$
- $\mathbf{P}_t = \mathbf{P}_t^- - \mathbf{K}_t\mathbf{S}_t\mathbf{K}_t^T$

From Linear to Gaussian Driven Nonlinear Dynamic State System

- Now let's consider another system which is little bit different, comparing with linear Gaussian dynamic state system. In the new system, so called Gaussian driven nonlinear dynamic state system, both state transition and observation become nonlinear function as following:

Gaussian Driven Nonlinear (Markov) Dynamic State System Model

- $\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}) + \mathbf{q}_{t-1}$
- $\mathbf{y}_t = \mathbf{h}(\mathbf{x}_t) + \mathbf{r}_t$
- $\mathbf{q}_{t-1} \sim \mathcal{N}(0, \mathbf{Q}_{t-1})$ is process noise
- $\mathbf{r}_t \sim \mathcal{N}(0, \mathbf{R}_t)$ is measurement noise
- $\mathbf{f}()$ is dynamic model function and $\mathbf{h}()$ measurement function

From Linear to Nonlinear Gaussian Dynamic State System

- Clearly, Kalman filter will NOT work for Gaussian driven nonlinear dynamic state system
- Naturally we ask question, how to deal with nonlinear dynamic model and measurement functions?
- The simple idea is try to convert the nonlinear functions into linear functions, how?
- Several ways, e.g. linear piece wise functions to replace the nonlinear function, Taylor expansion to approximate the nonlinear function, etc.
- For Extended Kalman Filter(EKF), we actually achieve linearization of nonlinear function by using Taylor expansion to approximate the nonlinear function

EKF: Linearization of Nonlinear Dynamic State System in Gaussian Distribution by 1st Order Taylor Expansion

As we discussed on the previous page, EKF linearizes the nonlinear dynamic state system in Gaussian distribution by using 1st order Taylor expansion. From this sub section, we will see how the EKF is derivate and how EKF works

- Firstly, let's recall how to use Taylor expansion to approximate an arbitrary function g(x)

Taylor Theorem in One Real Variable

- Let $k \geq 1$ be an integer, if the function $g(x)$ can be k times differentiable at the point a
- we have $g(x) = g(a) + g'(a)(x-a) + \frac{g''(a)}{2!}(x-a)^2 + ... + \frac{g^{(k)}(a)}{k!}(x-a)^k + h_k(x)(x-a)^k$
  where $\lim_{x \to a} h_k(x) = 0$

EKF: Linearization of Nonlinear Dynamic State System in Gaussian Distribution by 1st Order Taylor Expansion

- If we only use the first and second parts of Taylor expansion to approximate $g(x)$, which has a formal name 1st Order Taylor Expansion. Then obviously we will have $g(x) \approx g(a) + g'(a)(x - a)$ that is using a linear function $g(a) + g'(a)(x - a)$ to approximate the original nonlinear function $g(x)$. This is how we do linearization by using 1st order of Taylor expansion
- By doing this linearization for both dynamic model function $\mathbf{f}()$ and measurement function $\mathbf{h}()$, we could derivate something similar to Kalman filter from Bayesian filter, which is called Extended Kalman filter

EKF: Linearization of Nonlinear Dynamic State System in Gaussian Distribution by 1st Order Taylor Expansion

- By using 1st Order Taylor Expansion on both dynamic model function $\mathbf{f}()$ and measurement function $\mathbf{h}()$, we will get an approximation of both function:

1st Order Taylor Expansion Approximation of Dynamic Model Function $\mathbf{f}()$ and Measurement Function $\mathbf{h}()$

- $\mathbf{f}(\mathbf{x}) \approx \mathbf{f}(\mathbf{m}) + \mathbf{F}_{\mathbf{x}}(\mathbf{m})(\mathbf{x} - \mathbf{m})$
- $\mathbf{h}(\mathbf{x}) \approx \mathbf{h}(\mathbf{m}) + \mathbf{H}_{\mathbf{x}}(\mathbf{m})(\mathbf{x} - \mathbf{m})$
- where $\mathbf{x} \sim \mathcal{N}(\mathbf{m}, \mathbf{P})$
- $\mathbf{F}_{\mathbf{x}}()$ is Jacobian matrix of $\mathbf{f}(\mathbf{x})$, stands for a matrix consists of 1st order partial derivatives of function $\mathbf{f}(\mathbf{x})$
- $\mathbf{H}_{\mathbf{x}}()$ is Jacobian matrix of $\mathbf{h}(\mathbf{x})$

- For detail of Jacobian matrix, see reference

EKF: Linearization of Nonlinear Dynamic State System in Gaussian Distribution by 1st Order Taylor Expansion

- Note only the first item $\mathbf{f}(\mathbf{m})$ and $\mathbf{h}(\mathbf{m})$ contribute to the approximation mean value of function $\mathbf{f}(\mathbf{x})$, $\mathbf{h}(\mathbf{x})$.
  So we actually have $\mathbb{E}[\mathbf{f}(\mathbf{x})] \approx \mathbb{E}[\mathbf{f}(\mathbf{m}) + \mathbf{F_x}(\mathbf{m})(\mathbf{x} - \mathbf{m})] = \mathbb{E}[\mathbf{f}(\mathbf{m})] = \mathbf{f}(\mathbf{m})$ due that $\mathbb{E}[\mathbf{x}] = 0$, $\mathbb{E}[\mathbf{h}(\mathbf{x})] \approx \mathbf{h}(\mathbf{m})$ for same reason
- The second item $\mathbf{F_x}(\mathbf{m})(\mathbf{x} - \mathbf{m})$ and $\mathbf{H_x}(\mathbf{m})(\mathbf{x} - \mathbf{m})$ have 0 mean, they decide the approximation covariance value of function $\mathbf{f}(\mathbf{x})$, $\mathbf{h}(\mathbf{x})$.
  So we actually have
  $Cov[\mathbf{f}(\mathbf{x})] = \mathbb{E}[(\mathbf{f}(\mathbf{x}) - \mathbb{E}[\mathbf{f}(\mathbf{x})])(\mathbf{f}(\mathbf{x}) - \mathbb{E}[\mathbf{f}(\mathbf{x})])^T] \approx \mathbb{E}[(\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{m}))(\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{m}))^T] = \mathbf{F_x}(\mathbf{m})\mathbf{P}\mathbf{F_x}^T(\mathbf{m})$, and $Cov[\mathbf{f}(\mathbf{x})] \approx \mathbf{H_x}(\mathbf{m})\mathbf{P}\mathbf{H_x}^T(\mathbf{m})$ for same reason

EKF: Linearization of Nonlinear Dynamic State System in Gaussian Distribution by 1st Order Taylor Expansion

- So the approximation of Gaussian driven nonlinear dynamic state system as following:

Linearized Gaussian Driven Nonlinear Dynamic State System Model

- $\mathbf{x}_t = \mathbf{f}(\mathbf{m}) + \mathbf{F}_\mathbf{x}(\mathbf{m})(\mathbf{x}_{t-1} - \mathbf{m}) + \mathbf{q}_{t-1}$
- $\mathbf{y}_t = \mathbf{h}(\mathbf{m}) + \mathbf{H}_\mathbf{x}(\mathbf{m})(\mathbf{x}_t - \mathbf{m}) + \mathbf{r}_t$
- $\mathbf{q}_{t-1} \sim \mathcal{N}(0, \mathbf{Q}_{t-1})$ is process noise
- $\mathbf{r}_t \sim \mathcal{N}(0, \mathbf{R}_t)$ is measurement noise
- $\mathbf{F}_\mathbf{x}()$ and $\mathbf{H}_\mathbf{x}()$ are Jacobian matrix of $\mathbf{f}(\mathbf{x})$, $\mathbf{h}(\mathbf{x})$

Note: **In fact, nonlinear transform from Gaussian distribution will NOT be a Gaussian**. So we just assume $\mathbf{x}_t$ and $\mathbf{y}_t$ are Gaussian cause using linearization to replace of nonlinear functions.

EKF: Linearization of Nonlinear Dynamic State System in Gaussian Distribution by 1st Order Taylor Expansion

Same as we derivate for Kalman filter, we assume

- $P(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) = \mathcal{N}(\mathbf{m}_{t-1}, \mathbf{P}_{t-1})$ which denotes the probability of state $\mathbf{x}_t$ based on observations from 1 to t
- $P(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \mathcal{N}(\mathbf{m}_t^-, \mathbf{P}_t^-)$ which denotes the probability of state $\mathbf{x}_t$ based on observations from 1 to t-1
- We do the same thing as for Kalman filter, we could apply lemma 1 on $\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}$ and $\mathbf{x}_t|\mathbf{y}_{1:t-1}$ by define $\mathbf{a} = \mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}$ and $\mathbf{b} = \mathbf{x}_t|\mathbf{y}_{1:t-1}$
- Then we have $\mathbf{b}|\mathbf{a} = \mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{y}_{1:t-1}$
- Now we know $P(\mathbf{a}) = P(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) = \mathcal{N}(\mathbf{m}_{t-1}, \mathbf{P}_{t-1})$ and $P(\mathbf{b}|\mathbf{a}) = P(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{y}_{1:t-1}) = P(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t|\mathbf{f}(\mathbf{m}_{\mathbf{k}-1}), \mathbf{Q}_{t-1})$

EKF: Linearization of Nonlinear Dynamic State System in Gaussian Distribution by 1st Order Taylor Expansion

- By using rule which is similar to lemma 1 we get joint probability:
  $P(\mathbf{a} \bigcap \mathbf{b}) = P((\mathbf{x}_t|\mathbf{y}_{1:t-1}) \bigcap P(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})) =$
  $P((\mathbf{x}_t, \mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}))=$

$$\mathcal{N}\left(\begin{pmatrix} \mathbf{m}_{t-1} \\ \mathbf{f}(\mathbf{m}_{t-1}) \end{pmatrix}, \begin{pmatrix} \mathbf{P}_{t-1} & \mathbf{P}_{t-1}\mathbf{F}_{\mathbf{x}}^T(\mathbf{m}_{t-1}) \\ \mathbf{F}_{\mathbf{x}}(\mathbf{m}_{t-1})\mathbf{P}_{t-1} & \mathbf{F}_{\mathbf{x}}(\mathbf{m}_{t-1})\mathbf{P}\mathbf{F}_{\mathbf{x}}^T(\mathbf{m}_{t-1}) + \mathbf{Q}_{t-1} \end{pmatrix}\right)$$

- also marginal distribution of $\mathbf{b} = \mathbf{x}_t|\mathbf{y}_{1:t-1}$:
  $P(\mathbf{b}) = P(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \mathcal{N}(\mathbf{m}_t^-, \mathbf{P}_t^-) =$
  $\mathcal{N}(\mathbf{f}(\mathbf{m}_{t-1}), \mathbf{F}_{\mathbf{x}}(\mathbf{m}_{t-1})\mathbf{P}\mathbf{F}_{\mathbf{x}}^T(\mathbf{m}_{t-1}) + \mathbf{Q}_{t-1})$
- Note: here the rule used is a little bit modification of lemma 1, because the $P(\mathbf{b}|\mathbf{a}) = \mathcal{N}(\mathbf{x}_t|\mathbf{f}(\mathbf{m}_{t-1}), \mathbf{Q}_{t-1})$ NOT in the format $\mathcal{N}(\mathbf{b}|\mathbf{H}\mathbf{a}, \mathbf{R})$. So the mean of $P(\mathbf{b})$ is changed and covariance is actually calculated by adding up the covariance of $\mathbf{f}(\mathbf{x}_{t-1})$, $\mathbf{F}_{\mathbf{x}}(\mathbf{m})\mathbf{P}\mathbf{F}_{\mathbf{x}}^T(\mathbf{m})$, and the covariance of $\mathbf{q}_{t-1}$, $\mathbf{Q}_{t-1}$

EKF: Linearization of Nonlinear Dynamic State System in Gaussian Distribution by 1st Order Taylor Expansion

- Until now, we already got
  $\mathbf{m}_t^- = \mathbf{f}(\mathbf{m}_{t-1})$
  $\mathbf{P}_t^- = \mathbf{F}_\mathbf{x}(\mathbf{m}_{t-1})\mathbf{P}\mathbf{F}_\mathbf{x}^T(\mathbf{m}_{t-1}) + \mathbf{Q}_{t-1}$
  from prediction step by using the similar way of Kalman filter's derivation
- This is prediction step of EKF

Prediction step of Extended Kalman Filter

- $\mathbf{m}_t^- = \mathbf{f}(\mathbf{m}_{t-1})$
- $\mathbf{P}_t^- = \mathbf{F}_\mathbf{x}(\mathbf{m}_{t-1})\mathbf{P}\mathbf{F}_\mathbf{x}^T(\mathbf{m}_{t-1}) + \mathbf{Q}_{t-1}$

EKF: Linearization of Nonlinear Dynamic State System in Gaussian Distribution by 1st Order Taylor Expansion

- Keep doing the similar things for Kalman filter's derivation again. Next we define $\mathbf{a} = \mathbf{x}_t|\mathbf{y}_{1:t-1}$ and $\mathbf{b} = \mathbf{y}_t|\mathbf{y}_{1:t-1}$ and use rule similar to lemma 1 again then lemma 2 (**Here we skip the derivation, please follow how we derivate update step of Kalman filter to derivate by urself**)
- We can get the
  $\mathbf{m}_t =$
  $\mathbf{m}_t^- + \mathbf{P}_t^- \mathbf{H}_x^T(\mathbf{m}_t^-)(\mathbf{H}_x(\mathbf{m}_t^-)\mathbf{P}_t^- \mathbf{H}_x^T(\mathbf{m}_t^-) + \mathbf{R}_t)^{-1}(\mathbf{y}_t - \mathbf{h}(\mathbf{m}_t^-))$
  and $\mathbf{P}_t =$
  $\mathbf{P}_t^- - \mathbf{P}_t^- \mathbf{H}_x^T(\mathbf{m}_t^-)(\mathbf{H}_x(\mathbf{m}_t^-)\mathbf{P}_t^- \mathbf{H}_x^T(\mathbf{m}_t^-) + \mathbf{R}_t)^{-1}\mathbf{H}_x(\mathbf{m}_t^-)\mathbf{P}_t^-$
  which came from update step

EKF: Linearization of Nonlinear Dynamic State System in Gaussian Distribution by 1st Order Taylor Expansion

- This is update step of EKF

Measurements Update Step of Extended Kalman Filter

- $\mathbf{S}_t = \mathbf{H}_x(\mathbf{m}_t^-)\mathbf{P}_t^-\mathbf{H}_x^T(\mathbf{m}_t^-) + \mathbf{R}_t$
- $\mathbf{v}_t = \mathbf{y}_t - \mathbf{h}(\mathbf{m}_t^-)$
- $\mathbf{K}_t = \mathbf{P}_t^-\mathbf{H}_x^T(\mathbf{m}_t^-)\mathbf{S}_t^{-1}$
- $\mathbf{m}_t = \mathbf{m}_t^- + \mathbf{K}_t\mathbf{v}_t$
- $\mathbf{P}_t = \mathbf{P}_t^- - \mathbf{K}_t\mathbf{S}_t\mathbf{K}_t^T$

EKF: Linearization of Nonlinear Dynamic State System in Gaussian Distribution by 1st Order Taylor Expansion

Combine prediction and update, we get Extended Kalman filter

Prediction step of Extended Kalman Filter

- $\mathbf{m}_t^- = \mathbf{f}(\mathbf{m}_{t-1})$
- $\mathbf{P}_t^- = \mathbf{F}_\mathbf{x}(\mathbf{m}_{t-1})\mathbf{P}\mathbf{F}_\mathbf{x}^T(\mathbf{m}_{t-1}) + \mathbf{Q}_{t-1}$

Measurements Update Step of Extended Kalman Filter

- $\mathbf{S}_t = \mathbf{H}_x(\mathbf{m}_t^-)\mathbf{P}_t^-\mathbf{H}_x^T(\mathbf{m}_t^-) + \mathbf{R}_t$
- $\mathbf{v}_t = \mathbf{y}_t - \mathbf{h}(\mathbf{m}_t^-)$
- $\mathbf{K}_t = \mathbf{P}_t^-\mathbf{H}_x^T(\mathbf{m}_t^-)\mathbf{S}_t^{-1}$
- $\mathbf{m}_t = \mathbf{m}_t^- + \mathbf{K}_t\mathbf{v}_t$
- $\mathbf{P}_t = \mathbf{P}_t^- - \mathbf{K}_t\mathbf{S}_t\mathbf{K}_t^T$

EKF Algorithm(Start from t = 1)

for t in T **do**
    // Prediction step of Extended Kalman Filter
    $\mathbf{x}_t^- = \mathbf{f}(\mathbf{x}_{t-1})$
    $\mathbf{P}_t^- = \mathbf{F}_\mathbf{x}(\mathbf{x}_{t-1})\mathbf{P}\mathbf{F}_\mathbf{x}^T(\mathbf{x}_{t-1}) + \mathbf{Q}_{t-1}$
    // Measurements Update Step of Extended Kalman Filter
    $\mathbf{S}_t = \mathbf{H}_x(\mathbf{x}_t^-)\mathbf{P}_t^-\mathbf{H}_x^T(\mathbf{x}_t^-) + \mathbf{R}_t$
    $\mathbf{v}_t = \mathbf{y}_t - \mathbf{h}(\mathbf{x}_t^-)$
    $\mathbf{K}_t = \mathbf{P}_t^-\mathbf{H}_x^T(\mathbf{x}_t^-)\mathbf{S}_t^{-1}$
    $\mathbf{x}_t = \mathbf{x}_t^- + \mathbf{K}_t\mathbf{v}_t$
    $\mathbf{P}_t = \mathbf{P}_t^- - \mathbf{K}_t\mathbf{S}_t\mathbf{K}_t^T$
    $\mathrm{t} = \mathrm{t} + 1$
end for

Outline for Section 2

UKF: Better Way(2nd Order Taylor Expansion) to Linearization by Unscented Transform

UKF

Note

-

Outline for Section 3

1. Extended Kalman Filter
   - Revisit Kalman Filter
   - If System Changes from Linear to Nonlinear
   - EKF: (1st Order Taylor Expansion)Linearization of Nonlinear Dynamic State System in Gaussian Distribution
2. Unscented Kalman Filter
   - UKF: Better Way(2nd Order Taylor Expansion) to Linearization by Unscented Transform
3. Particle Filter
   - Sequential Importance Sampling
   - PF: Monte Carlo Method for Arbitrary Distribution State by Using Sequential Importance Sampling and Resampling
4. Conclusion

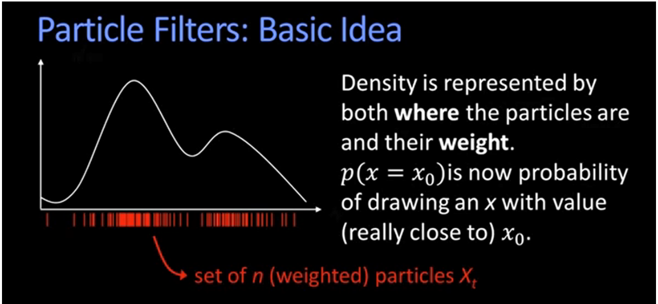
Importance Sampling: Way to Approximate Arbitrary Distribution

- Now let's consider the case where dynamic state system is Non-linear and Non-Gaussian, which means the distribution of random variable $\mathbf{x}$ can be anything. We define this "arbitrary" distribution of random variable $\mathbf{x}$ as $p(\mathbf{x})$
- According to the definition of Bayesian filter in general, $P(\mathbf{x}_t|\mathbf{y}_{1:t}) = \frac{P(\mathbf{y}_{1:t-1})}{P(\mathbf{z}_{1:t})} P(\mathbf{y}_t|\mathbf{x}_t) P(\mathbf{x}_t|\mathbf{y}_{1:t-1})$, suppose we would like to know $\mathbb{E}[f(\mathbf{x}_t)] = \int f(\mathbf{x}_t) p(\mathbf{x}_t|\mathbf{y}_{1:t})\, dx$, where note $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ is "arbitrary" unknown distribution(NOT Gaussian or any known, easy to represent distribution)
- So the problem is, we do NOT know $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ and it is NOT easy to represent this "arbitrary" unknown distribution due it is NOT Gaussian anymore

Importance Sampling: Way to Approximate Arbitrary Distribution

How to represent this "arbitrary" unknown distribution $p(\mathbf{x}_t|\mathbf{y}_{1:t})$?

- Answer is, by using particles in Monte Carlo(MC) framework
- Although we do NOT know the "shape" of this arbitrary unknown distribution, but in case we could have very large amount of samples which follows this arbitrary unknown distribution, what we could get? Let's see the figure



Importance Sampling: Way to Approximate Arbitrary Distribution

How to represent this "arbitrary" unknown distribution $p(\mathbf{x}_t|\mathbf{y}_{1:t})$?

- Obviously, the density of samples/particles reflects the "shape" of distribution function, the more particles in a certain value the higher output of distribution function $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ in that certain value of $\mathbf{x}_t^{(i)}$, where i stands for i^{th} particle
- In the other words, the total number of samples/particles shown on the certain value $\mathbf{x}_t^{(i)}$(Note: It for sure could be more than 1 particles shown on the same certain value $\mathbf{x}_t^{(i)}$, which means $\mathbf{x}_t^{(i)}$ can be same as $\mathbf{x}_t^{(j)}$ where $i \neq j$) divides the total number of particles, N, equals to $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ approximately, in case N is large enough

Importance Sampling: Way to Approximate Arbitrary Distribution

- That is
  $p(\mathbf{x}_t|\mathbf{y}_{1:t}) \approx \hat{p}(\mathbf{x}_t|\mathbf{y}_{1:t}) = \frac{1}{N}\sum_{i=1}^{N}\delta(\mathbf{x}_t - \mathbf{x}_t^{(i)})$
- where $\delta(\mathbf{x}_t - \mathbf{x}_t^{(i)})$ is Dirac function, means if $\mathbf{x}_t = \mathbf{x}_t^{(i)}$ the output is 1, otherwise output is 0.
- It is easy to understand this approximation of $p(\mathbf{x}_t|\mathbf{y}_{1:t})$, is done by counting the portion of total number of particles which dropped on the same position, $\mathbf{x}_t^{(i)}$(as we discussed, more than one particles can have the same value), out of the total number of particles, N. This portion is approximately equivalent to $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ if N is large

Importance Sampling: Way to Approximate Arbitrary Distribution

So we go back to our original problem, what is $\mathbb{E}[f(\mathbf{x}_t)]$?

- By using $p(\mathbf{x}_t|\mathbf{y}_{1:t}) \approx \hat{p}(\mathbf{x}_t|\mathbf{y}_{1:t}) = \frac{1}{N}\sum_{i=1}^{N} \delta(\mathbf{x}_t - \mathbf{x}_t^{(i)})$, we can get
- $\mathbb{E}[f(\mathbf{x}_t)] = \int f(\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t \approx \int f(\mathbf{x}_t)\hat{p}(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t = \frac{1}{N}\sum_{i=1}^{N} \int f(\mathbf{x}_t)\delta(\mathbf{x}_t - \mathbf{x}_t^{(i)})\, d\mathbf{x}_t = \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_t^{(i)})$
- It means we get $\mathbb{E}[f(\mathbf{x}_t)] = \int f(\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t \approx \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_t^{(i)})$

Importance Sampling: Way to Approximate Arbitrary Distribution

However we noticed the particles should be sampled from/by following this "arbitrary" unknown distribution $p(\mathbf{x}_t|\mathbf{y}_{1:t})$, but the problem is, we do NOT know this "arbitrary" unknown distribution $p(\mathbf{x}_t|\mathbf{y}_{1:t})$, how to sample particles which follows this distribution?

- In order to find some way to calculate $\mathbb{E}[f(\mathbf{x}_t)]$, we introduce "importance sampling" as follow:

Importance Sampling:

- Suppose we have a known distribution(e.g. Gaussian) $q(\mathbf{x}_t|\mathbf{y}_{1:t})$
- Let's see how to use this known distribution to calculate $\mathbb{E}[f(\mathbf{x}_t)]$

Importance Sampling: Way to Approximate Arbitrary Distribution

- $\mathbb{E}[f(\mathbf{x}_t)] = \int f(\mathbf{x}_t) p(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t =$
  $\int f(\mathbf{x}_t) p(\mathbf{x}_t|\mathbf{y}_{1:t}) \cdot q(\mathbf{x}_t|\mathbf{y}_{1:t}) / q(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t =$
  $\int f(\mathbf{x}_t) \frac{p(\mathbf{y}_{1:t}|\mathbf{x}_t) p(\mathbf{x}_t)}{p(\mathbf{y}_{1:t})} \cdot q(\mathbf{x}_t|\mathbf{y}_{1:t}) / q(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t =$
  $\frac{1}{p(\mathbf{y}_{1:t})} \int f(\mathbf{x}_t) \frac{p(\mathbf{y}_{1:t}|\mathbf{x}_t) p(\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{y}_{1:t})} \cdot q(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t,$
  due to Bayes rule $p(\mathbf{x}_t|\mathbf{y}_{1:t}) p(\mathbf{y}_{1:t}) = p(\mathbf{y}_{1:t}|\mathbf{x}_t) p(\mathbf{x}_t)$
- Now define $W_t(\mathbf{x}_t) = \frac{p(\mathbf{y}_{1:t}|\mathbf{x}_t) p(\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{y}_{1:t})}$
- So we get $\mathbb{E}[f(\mathbf{x}_t)] = \frac{1}{p(\mathbf{y}_{1:t})} \int f(\mathbf{x}_t) W_t(\mathbf{x}_t) \cdot q(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t$

Importance Sampling: Way to Approximate Arbitrary Distribution

- Notice $p(\mathbf{y}_{1:t}) = \int p(\mathbf{x}_t \cap \mathbf{y}_{1:t})\, d\mathbf{x}_t = \int p(\mathbf{y}_{1:t}|\mathbf{x}_t)p(\mathbf{x}_t)\, d\mathbf{x}_t$
- So origin could be rewritten as:

$$\mathbb{E}[f(\mathbf{x}_t)] = \frac{1}{p(\mathbf{y}_{1:t})}\int f(\mathbf{x}_t)W_t(\mathbf{x}_t)\cdot q(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t =$$

$$\frac{\displaystyle\int f(\mathbf{x}_t)W_t(\mathbf{x}_t)\cdot q(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t}{\displaystyle\int p(\mathbf{y}_{1:t}|\mathbf{x}_t)p(\mathbf{x}_t)\, d\mathbf{x}_t} =$$

$$\frac{\displaystyle\int f(\mathbf{x}_t)W_t(\mathbf{x}_t)\cdot q(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t}{\displaystyle\int \frac{p(\mathbf{y}_{1:t}|\mathbf{x}_t)p(\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{y}_{1:t})}\cdot q(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t}$$

Importance Sampling: Way to Approximate Arbitrary Distribution

- Cause we have $W_t(\mathbf{x}_t) = \frac{p(\mathbf{y}_{1:t}|\mathbf{x}_t)p(\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{y}_{1:t})}$
- It means $\mathbb{E}[f(\mathbf{x}_t)] = \dfrac{\displaystyle\int f(\mathbf{x}_t)W_t(\mathbf{x}_t)\cdot q(\mathbf{x}_t|\mathbf{y}_{1:t})\,d\mathbf{x}_t}{\displaystyle\int \frac{p(\mathbf{y}_{1:t}|\mathbf{x}_t)p(\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{y}_{1:t})}\cdot q(\mathbf{x}_t|\mathbf{y}_{1:t})\,d\mathbf{x}_t} =$
$$\frac{\displaystyle\int f(\mathbf{x}_t)W_t(\mathbf{x}_t)\cdot q(\mathbf{x}_t|\mathbf{y}_{1:t})\,d\mathbf{x}_t}{\displaystyle\int W_t(\mathbf{x}_t)\cdot q(\mathbf{x}_t|\mathbf{y}_{1:t})\,d\mathbf{x}_t}$$
- Note $q(\mathbf{x}_t|\mathbf{y}_{1:t})$ is the distribution we know, that means we could sample particle from this known distribution easily.

Importance Sampling: Way to Approximate Arbitrary Distribution

- Recall the conclusion we got from previous derivation, $\mathbb{E}[f(\mathbf{x}_t)] = \int f(\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t \approx \int f(\mathbf{x}_t)\hat{p}(\mathbf{x}_t|\mathbf{y}_{1:t})\, d\mathbf{x}_t = \frac{1}{N}\sum_{i=1}^{N} \int f(\mathbf{x}_t)\delta(\mathbf{x}_t - \mathbf{x}_t^{(i)})\, d\mathbf{x}_t = \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_t^{(i)})$, where $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ is arbitrary distribution. And now we have a known distribution $q(\mathbf{x}_t|\mathbf{y}_{1:t})$ can be sampled from

Importance Sampling: Way to Approximate Arbitrary Distribution

- Applying conclusion recalled last page, obviously we can get

$$\mathbb{E}[f(\mathbf{x}_t)] = \frac{\displaystyle\int f(\mathbf{x}_t)W_t(\mathbf{x}_t)\cdot q(\mathbf{x}_t|\mathbf{y}_{1:t})\,d\mathbf{x}_t}{\displaystyle\int W_t(\mathbf{x}_t)\cdot q(\mathbf{x}_t|\mathbf{y}_{1:t})\,d\mathbf{x}_t} = \frac{\mathbb{E}[f(\mathbf{x}_t)W_t(\mathbf{x}_t)]}{\mathbb{E}[W_t(\mathbf{x}_t)]}$$

$$= \frac{\displaystyle\frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_t^{(i)})W_t(\mathbf{x}_t^{(i)})}{\displaystyle\frac{1}{N}\sum_{i=1}^{N} W_t(\mathbf{x}_t^{(i)})},$$ where $\mathbf{x}_t|\mathbf{y}_{1:t}$ follows known distribution $q(\mathbf{x}_t|\mathbf{y}_{1:t})$

Importance Sampling: Way to Approximate Arbitrary Distribution

- Rewrite the result again

$$\mathbb{E}[f(\mathbf{x}_t)] = \frac{\frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_t^{(i)}) W_t(\mathbf{x}_t^{(i)})}{\frac{1}{N}\sum_{i=1}^{N} W_t(\mathbf{x}_t^{(i)})} = \sum_{i=1}^{N} \widetilde{W}_t(\mathbf{x}_t^{(i)}) f(\mathbf{x}_t^{(i)}),$$

  where $\widetilde{W}_t(\mathbf{x}_t^{(i)}) = \frac{W_t(\mathbf{x}_t^{(i)})}{\sum_{i=1}^{N} W_t(\mathbf{x}_t^{(i)})}$ is called as normalized weight
- Until now, we finished up derivation of "importance sampling", changing the expression from something with an unknown distribution to something with known distribution and represented in form of approximation by particles
- Now the issue we concern on is, how to calculate $W_t(\mathbf{x}_t^{(i)})$? Let's move to next sub section for the answer

Sequential Importance Sampling

From this sub section, we will introduce method to calculate $W_t(\mathbf{x}_t^{(i)})$ in a recursive format, it is **Sequential Importance Sampling**:

- First **assume** our chosen "known distribution" $q(\cdot)$ satisfies: $q(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}) = q(\mathbf{x}_t|\mathbf{x}_{0:t-1}, \mathbf{y}_{1:t})q(\mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1})$
- Note here we are manipulating with $q(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$ rather than $q(\mathbf{x}_t|\mathbf{y}_{1:t})$

Sequential Importance Sampling

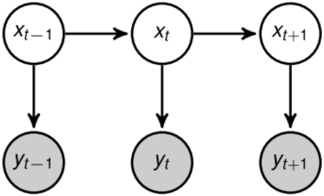
Then let's analyze "arbitrary unknown" distribution $p(\cdot)$

- Similarly, we look at the posterior of all states $\mathbf{x}_{0:t}$ satisfies $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t|\mathbf{x}_{0:t}, \mathbf{y}_{1:t-1})p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t-1})}{p(\mathbf{y}_t|\mathbf{y}_{1:t-1})}$ due to Bayes rule $P(B|A) = \frac{P(A|B)P(B)}{P(A)}$ with setting of $B \leftarrow \mathbf{x}_{0:t}|\mathbf{y}_{1:t-1}$ and $A \leftarrow \mathbf{y}_t|\mathbf{y}_{1:t-1}$
- Notice $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t-1}) = p(\mathbf{x}_t|\mathbf{x}_{0:t-1}\mathbf{y}_{1:t-1})p(\mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1})$ due to Bayes rule $P(B|A)P(A) = P(A \bigcap B)$ with setting of $B \leftarrow \mathbf{x}_t|\mathbf{y}_{1:t-1}$ and $A \leftarrow \mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1}$
- Combine them, we get

$$p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t|\mathbf{x}_{0:t}, \mathbf{y}_{1:t-1})p(\mathbf{x}_t|\mathbf{x}_{0:t-1}\mathbf{y}_{1:t-1})p(\mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1})}{p(\mathbf{y}_t|\mathbf{y}_{1:t-1})}$$

Sequential Importance Sampling

- Due to our model is a HMM(Recall HMM in Bayesian filter slide)



- observation $\mathbf{y}_t$ is only depend on hidden state $p(\mathbf{x}_t$(That is "All observations $\mathbf{y}_t$ are conditional independent on state $\mathbf{x}_t$") and hidden state $p(\mathbf{x}_t$ is only dependent on previous state hidden state $p(\mathbf{x}_{t-1}$(That is state $\mathbf{x}_t$ is a Markov process)

Sequential Importance Sampling

- By using these two rules, we have $p(\mathbf{y}_t|\mathbf{x}_{0:t}, \mathbf{y}_{1:t-1}) = p(\mathbf{y}_t|\mathbf{x}_t)$ and $p(\mathbf{x}_t|\mathbf{x}_{0:t-1}\mathbf{y}_{1:t-1}) = p(\mathbf{x}_t|\mathbf{x}_{t-1})$
- Then we get

$$p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t|\mathbf{x}_{0:t}, \mathbf{y}_{1:t-1})p(\mathbf{x}_t|\mathbf{x}_{0:t-1}\mathbf{y}_{1:t-1})p(\mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1})}{p(\mathbf{y}_t|\mathbf{y}_{1:t-1})}$$
$$= \frac{p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1})}{p(\mathbf{y}_t|\mathbf{y}_{1:t-1})}$$
$$\propto p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1})$$

- In the other words $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}) \propto p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1})$

Sequential Importance Sampling

- Recall we already defined $W_t(\mathbf{x}_t) = \frac{p(\mathbf{y}_{1:t}|\mathbf{x}_t)p(\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{y}_{1:t})}$ in the sub section "Importance Sampling"
- So we have $W_t(\mathbf{x}_t) = \frac{p(\mathbf{y}_{1:t}|\mathbf{x}_t)p(\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{y}_{1:t})} = \frac{p(\mathbf{x}_t|\mathbf{y}_{1:t})p(\mathbf{y}_{1:t})}{q(\mathbf{x}_t|\mathbf{y}_{1:t})}$ $\propto \frac{p(\mathbf{x}_t|\mathbf{y}_{1:t})}{q(\mathbf{x}_t|\mathbf{y}_{1:t})}$
- Notice $\frac{p(\mathbf{x}_t|\mathbf{y}_{1:t})}{q(\mathbf{x}_t|\mathbf{y}_{1:t})} \propto \frac{p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})}{q(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})}$
- We could get
  $W_t(\mathbf{x}_t) \propto \frac{p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})}{q(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})}$ which is $W_t(\mathbf{x}_t^{(i)}) \propto \frac{p(\mathbf{x}_{0:t}^{(i)}|\mathbf{y}_{1:t})}{q(\mathbf{x}_{0:t}^{(i)}|\mathbf{y}_{1:t})}$

Sequential Importance Sampling

- As we already got $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}) \propto p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1})$ and assumed $q(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}) = q(\mathbf{x}_t|\mathbf{x}_{0:t-1}, \mathbf{y}_{1:t})q(\mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1})$
- Then we could derivate as below:
$$W_t(\mathbf{x}_t^{(i)}) \propto \frac{p(\mathbf{x}_{0:t}^{(i)}|\mathbf{y}_{1:t})}{q(\mathbf{x}_{0:t}^{(i)}|\mathbf{y}_{1:t})}$$
$$\propto \frac{p(\mathbf{y}_t|\mathbf{x}_t^{(i)})p(\mathbf{x}_t^{(i)}|\mathbf{x}_{t-1}^{(i)})p(\mathbf{x}_{0:t-1}^{(i)}|\mathbf{y}_{1:t-1})}{q(\mathbf{x}_t^{(i)}|\mathbf{x}_{0:t-1}^{(i)}, \mathbf{y}_{1:t})q(\mathbf{x}_{0:t-1}^{(i)}|\mathbf{y}_{1:t-1})}$$
- Due that $\frac{p(\mathbf{x}_{0:t-1}^{(i)}|\mathbf{y}_{1:t-1})}{q(\mathbf{x}_{0:t-1}^{(i)}|\mathbf{y}_{1:t-1})} = W_{t-1}(\mathbf{x}_{t-1}^{(i)})$
- We could get $W_t(\mathbf{x}_t^{(i)}) = \frac{p(\mathbf{y}_t|\mathbf{x}_t^{(i)})p(\mathbf{x}_t^{(i)}|\mathbf{x}_{t-1}^{(i)})}{q(\mathbf{x}_t^{(i)}|\mathbf{x}_{0:t-1}^{(i)}, \mathbf{y}_{1:t})} W_{t-1}(\mathbf{x}_{t-1}^{(i)})$
- Now we could calculate $W_t(\mathbf{x}_t^{(i)})$ recursively

Sequential Importance Sampling

- And after calculation of $W_t(\mathbf{x}_t^{(i)})$ recursively, do NOT forget to normalize weight by using $\widetilde{W}_t(\mathbf{x}_t^{(i)}) = \frac{W_t(\mathbf{x}_t^{(i)})}{\sum_{i=1}^{N} W_t(\mathbf{x}_t^{(i)})}$
- At last step, we could calculate $\mathbb{E}[f(\mathbf{x}_t)] = \sum_{i=1}^{N} \widetilde{W}_t(\mathbf{x}_t^{(i)}) f(\mathbf{x}_t^{(i)})$
- All in all, we get particle filter algorithm by sequential importance sampling as in next page:

Sequential Importance Sampling Algorithm(from t=1)

for t in T **do**
  for i in N **do**
    $\mathbf{x}_t^{(i)} \sim q(\mathbf{x}_t^{(i)} | \mathbf{x}_{0:t-1}^{(i)}, \mathbf{y}_{1:t})$ // draw particles follows $q(\cdot)$
    $\mathrm{i} = \mathrm{i} + 1$
  end for
  for i in N **do**
    $$W_t(\mathbf{x}_t^{(i)}) = \frac{p(\mathbf{y}_t | \mathbf{x}_t^{(i)}) p(\mathbf{x}_t^{(i)} | \mathbf{x}_{t-1}^{(i)})}{q(\mathbf{x}_t^{(i)} | \mathbf{x}_{0:t-1}^{(i)}, \mathbf{y}_{1:t})} W_{t-1}(\mathbf{x}_{t-1}^{(i)})$$
    // calculate $W_t(\mathbf{x}_t^{(i)})$
    $\mathrm{i} = \mathrm{i} + 1$
  end for
  $$\widetilde{W}_t(\mathbf{x}_t^{(i)}) = \frac{W_t(\mathbf{x}_t^{(i)})}{\sum_{i=1}^{N} W_t(\mathbf{x}_t^{(i)})}$$
  $$\mathbb{E}[f(\mathbf{x}_t)] = \sum_{i=1}^{N} \widetilde{W}_t(\mathbf{x}_t^{(i)}) f(\mathbf{x}_t^{(i)})$$
end for

Particle Filter: Sequential Importance Sampling and Resampling

- It seems the sequential importance sampling algorithm works! But, in fact, ...
- The distribution of the importance weights becomes more and more skewed as time increases. After a few iterations of algorithm, only few or one $W_t(\mathbf{x}_t^{(i)})$ will be non-zero. It is often called weight degeneracy
- This is disadvantageous cause a lot of computing effort is wasted to update those trivial weight
- How to solve this problem?
- We will introduce a new concept for measurement of degeneracy, the so-called effective sample size, N_{eff}

Particle Filter: Sequential Importance Sampling and Resampling

- $N_{eff} = \frac{N}{1 + Var_{q(\cdot|\mathbf{y}_{1:t})}[\widetilde{W_t}(\mathbf{x}_{0:t}^{(i)})]} = \frac{N}{\mathbb{E}_{q(\cdot|\mathbf{y}_{1:t})}[\widetilde{W_t}(\mathbf{x}_{0:t}^{(i)})^2]} \leq N$
- where $Var_{q(\cdot|\mathbf{y}_{1:t})}$ stands the variance of random variable which follows the distribution $q(\cdot|\mathbf{y}_{1:t})$
- A smaller N_{eff} means a larger variance for the weights, which means most of particles become very light weight and only few has high weight, hence more degeneracy
- So we usually define a threshold $N_{threshold} = \frac{N}{3}$, or $N_{threshold} = \frac{N}{2}$, as soon as $N_{eff} \leq N_{threshold}$ we do resmapling.
- By doing this resampling together with sequential importance sampling, we get the particle filter algorithm
- In practical we use $\hat{N}_{eff} = \frac{1}{\sum_{i=1}^{N}(\widetilde{W_t}(\mathbf{x}_t^{(i)}))^2}$ to replace N_{eff}

Particle Filter Algorithm(Start from t=1)

```
for t in T do
    Implement Sequential Importance Sampling Algorithm for t
    if N̂_eff ≥ N_threshold then
        Continue to next iteration of t
    else
        Generate a new particle x_t^(j) set by resampling with
        replacement N times from the previous set x_{0:t}^(i) with
        probabilities Pr(x_{0:t}^(j) = x_{0:t}^(i))= W̃_t(x_{0:t}^(i)), reset the
        weights W̃_t(x_t^(i)) = 1/N;
        Continue to next iteration of t
    end if
end for
```

for t in T **do**

Implement Sequential Importance Sampling Algorithm for t

if $\hat{N}_{eff} \geq N_{threshold}$ **then**

Continue to next iteration of t

else

Generate a new particle $\mathbf{x}_t^{(j)}$ set by resampling with

replacement N times from the previous set $\mathbf{x}_{0:t}^{(i)}$ with

probabilities $\Pr(\mathbf{x}_{0:t}^{(j)} = \mathbf{x}_{0:t}^{(i)}) = \widetilde{W}_t(\mathbf{x}_{0:t}^{(i)})$, reset the

weights $\widetilde{W}_t(\mathbf{x}_t^{(i)}) = \frac{1}{N}$;

Continue to next iteration of t

end if

end for

Outline for Section 4

Conclusion

-
-
-
-

Conclusion

-

References

- Zhe Chen, Bayesian Filtering: From Kalman Filters to Particle Filters, and Beyond
- Simo Sarkka, Nonlinear Filtering Parameter Estimation Course, Aalto University
- M. S. Arulampalam, Simon Maskell, Neil Gordon, Tim Clapp, Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking
- Fredrik Gustafsson, Particle Filter Theory and Practice with Positioning Application
- Michal Reintein, From Bayes to Extended Kalman Filter
- Greg Welch, Gary Bishop, An Introduction to the Kalman Filter
- Eric A. Wan and Rudolph van der Merwe, The Unscented Kalman Filter for Nonlinear Estimation

References

- Ashutosh Saxena, Robot Learning Course, Cornell University
- Cyrill Stachniss, Robot Mapping and SLAM Course, University of Freiburg/University of Bonn
- Gustaf Hendeby, Sensor Fusion Course, Linkoping University
- Paul Michael Newman, EKF Based Navigation and SLAM, Oxford University
- Wikipedia: Extended Kalman Filter ▸ Link
- Wikipedia: Taylor's Theorem ▸ Link
- Wikipedia: Jacobian Matrix ▸ Link
- Simon Haykin, Kalman Filtering and Neural Network
- Simo Sarkka, Kalmen Bayesian Filtering and Smoothing

Question?