Particle Filters: Beyond the Kalman filter

Amir Emadzadeh

March 11, 2017

Outline

Nonlinear Bayesian Tracking

- Bayesian estimation: construct posterior pdf of state based on all available information, including recieved measurements.
- State Model:

$$\mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1}, \mathbf{v}_{k-1})$$

  where $\mathbf{f}_k \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_v} \to \mathbb{R}^{n_x}$, $k \in \mathbb{N}$, and $\mathbf{v}_{k-1}$ is i.i.d noise
- Measurement:

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k, \mathbf{n}_k)$$

  where $\mathbf{h}_k \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_n} \to \mathbb{R}^{n_z}$, and $\mathbf{n}_k$ is i.i.d noise
- Goal: Estimate $\mathbf{x}_k$ using $\mathbf{z}_{1:k} = \{\mathbf{z}_i, i = 1, \cdots, k\}$
- Bayesian perspective: Recursively calculate belief: $p(\mathbf{x}_k|\mathbf{z}_{1:k})$
- Initial pdf: $p(\mathbf{x}_0|\mathbf{z}_0) \equiv p(x_0)$, $\mathbf{z}_0$: no measurements

Nonlinear Bayesian Tracking

- Optimal Bayesian Solution: Recursively calculate exact posterior pdf
- Predecition: Chapman-Kolmogorov

$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})d\mathbf{x}_{k-1}$$

where $p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_{1:k-1}) = p(\mathbf{x}_k|\mathbf{x}_{k-1})$: Markov oder One

- Update: Bayes' rule

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}_{1:k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})}$$

where normalizing constant:

$$p(\mathbf{z}_k|\mathbf{z}_{1:k-1}) = \int p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}_{1:k-1})d\mathbf{x}_k$$

Outline

Kalman Filter

- Assumptions:

$$x_k = F_k \mathbf{x}_{k-1} + \mathbf{v}_{k-1}$$
$$\mathbf{z}_k = H_k \mathbf{x}_k + \mathbf{n}_k$$

where $\mathbf{v}_{k-1}$ and $\mathbf{n}_k$ are zero-mean Gaussian, statically independent, and

$$E[\mathbf{v}_{k-1}\mathbf{v}_{k-1}^T] = Q_{k-1}$$
$$E[\mathbf{n}_k \mathbf{n}_k^T] = R_k$$

- Kalman filter algorithm:

$$p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) = \mathcal{N}(\mathbf{x}_{k-1}; m_{k-1|k}, P_{k-1|k-1})$$
$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \mathcal{N}(\mathbf{x}_k; m_{k|k-1}, P_{k|k-1})$$
$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \mathcal{N}(\mathbf{x}_k; m_{k|k}, P_{k|k})$$

Kalman Filter

- Kalman filter algorithm:

$$
\begin{aligned}
m_{k|k-1} &= F_k m_{k-1|k-1} \\
P_{k|k-1} &= Q_{k-1} + F_k P_{k-1|k-1} F_k^T \\
m_{k|k} &= m_{k|k-1} + K_k(\mathbf{z}_k - H_k m_{k|k-1}) \\
P_{k|k} &= P_{k|k-1} - K_k H_k P_{k|k-1}
\end{aligned}
$$

  where S_k is covariance of innovation, and K_k is Kalman gain:

$$
\begin{aligned}
S_k &= H_k P_{k|k-1} H_k^T + R_k \\
K_k &= P_{k|k-1} H_k^T S_k^{-1}
\end{aligned}
$$

- Note: Same algorithm can be obtained using least squares, Gaussian assumption not necessary

Outline

Grid-based Methods

- State space consists of finite number of discrete states $\mathbf{x}_{k-1}^i, i = 1, ..., N_s$
- If

$$\Pr(\mathbf{x}_{k-1} = \mathbf{x}_{k-1}^i | \mathbf{z}_{1:k-1}) = w_{k-1|k-1}^i$$

- Then, posterior pdf:

$$p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1}) = \sum_{i=1}^{N_s} w_{k-1|k-1}^i \delta(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^i)$$

- Prediction, Update:

$$p(\mathbf{x}_k | \mathbf{z}_{1:k-1}) = \sum_{i=1}^{N_s} w_{k|k-1}^i \delta(\mathbf{x}_k - \mathbf{x}_k^i)$$

$$p(\mathbf{x}_k | \mathbf{z}_{1:k}) = \sum_{i=1}^{N_s} w_{k|k}^i \delta(\mathbf{x}_k - \mathbf{x}_k^i)$$

Grid-based Methods

- Where

$$w^i_{k|k-1} \triangleq \sum_{j=1}^{N_s} w^i_{k-1|k-1} p(\mathbf{x}^i_k | \mathbf{x}^j_{k-1})$$

$$w^i_{k|k} \triangleq \frac{w^i_{k|k-1} p(\mathbf{z}_k | \mathbf{x}^i_k)}{\sum_{j=1}^{N_s} w^j_{k|k-1} p(\mathbf{z}_k | \mathbf{x}^j_k)}$$

- Assumption: $p(\mathbf{x}^i_k | \mathbf{x}^j_{k-1})$ and $p(\mathbf{z}_k | \mathbf{x}^i_k)$ are known.

Outline

Extended Kalman Filter

- EKF: Approximates $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ by a Gaussian:

$$p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) \approx \mathcal{N}(\mathbf{x}_{k-1}; m_{k-1|k}, P_{k-1|k-1})$$
$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) \approx \mathcal{N}(\mathbf{x}_k; m_{k|k-1}, P_{k|k-1})$$
$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \mathcal{N}(\mathbf{x}_k; m_{k|k}, P_{k|k})$$

where

$$m_{k|k-1} = \mathbf{f}_k(m_{k-1|k-1})$$
$$P_{k|k-1} = Q_{k-1} + \hat{F}_k P_{k-1|k-1} \hat{F}_k^T$$
$$m_{k|k} = m_{k|k-1} + K_k(\mathbf{z}_k - \mathbf{h}_k(m_{k|k-1}))$$
$$P_{k|k} = P_{k|k-1} - K_k \hat{H}_k P_{k|k-1}$$

Extended Kalman Filter

- Local linearization

$$\hat{F}_k = \frac{d\mathbf{f}_k(x)}{dx}\Big|_{x=m_{k-1|k-1}}$$
$$\hat{H}_k = \frac{d\mathbf{h}_k(x)}{dx}\Big|_{x=m_{k|k-1}}$$
$$S_K = \hat{H}_k P_{k|k-1} \hat{H}_k^T + R_k$$
$$K_k = P_{k|k-1} \hat{H}_k^T S_k^{-1}$$

- EKF utilizes first term in Taylor expansion of nonlinear functions

Outline

Approximate Grid-Based Methods

- Continuous state space: Decomposed into N_s cells $\{\mathbf{x}_k^i : i = 1, ..., N_s\}$
- Posterior pdf:

$$p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) \approx \sum_{i=1}^{N_s} w_{k-1|k-1}^i \delta(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^i)$$

- Prediction, Update:

$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) \approx \sum_{i=1}^{N_s} w_{k|k-1}^i \delta(\mathbf{x}_k - \mathbf{x}_k^i)$$

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_s} w_{k|k}^i \delta(\mathbf{x}_k - \mathbf{x}_k^i)$$

Approximate Grid-Based Methods

- where

$$w^i_{k|k-1} \triangleq \sum_{j=1}^{N_s} w^i_{k-1|k-1} \int_{x \in \mathbf{x}^i_k} p(\mathbf{x}|\bar{\mathbf{x}}^j_{k-1}) d\mathbf{x}$$

$$w^i_{k|k} \triangleq \frac{w^i_{k|k-1} \int_{\mathbf{x} \in \mathbf{x}^i_k} p(\mathbf{z}_k|\mathbf{x}) d\mathbf{x}}{\sum_{j=1}^{N_s} w^j_{k|k-1} \int_{\mathbf{x} \in \mathbf{x}^j_k} p(\mathbf{z}_k|\mathbf{x}) d\mathbf{x}}$$

$\bar{\mathbf{x}}^j_{k-1}$: center of j-th cell

Approximate Grid-Based Methods

- Further approximation: weights computed at center of cells

$$w^i_{k|k-1} \approx \sum_{j=1}^{N_s} w^i_{k-1|k-1} p(\bar{\mathbf{x}}^i_k | \bar{\mathbf{x}}^j_{k-1})$$

$$w^i_{k|k} \approx \frac{w^i_{k|k-1} p(\mathbf{z}_k | \bar{\mathbf{x}}^i_k)}{\sum_{j=1}^{N_s} w^j_{k|k-1} p(\mathbf{z}_k | \bar{\mathbf{x}}^j_k)}$$

- As dimension of state space increases, computional cost increases dramatically.
- State space predefined: Cannot be partitioned enevenly for greater resolution in high probability density regions.

Outline

SIS Algorithm

- A Monte Carlo method
- Represent posterior pdf by a set of random samples with associated weights.
- Let $\{\mathbf{x}_{0:k}^i, w_k^i\}_{i=1}^{N_s}$: Random measure which characterizes posterior pdf $p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$
- $\mathbf{x}_{0:k} = \{\mathbf{x}_j, j = 0, ..., k\}$: Set of all states up to time k
- $\{\mathbf{x}_{0:k}^i, i = 0, ..., N_s\}$: Set of support points associated with weghts $\{w_k^i, i = 1, ..., N_s\}$
- $\sum_i w_k^i = 1$
- Then, posterior pdf is approximated:

$$p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(\mathbf{x}_{0:k} - \mathbf{x}_{0:k}^i)$$

SIS Algorithm

- Importance sampling: Suppose $p(x) \propto \pi(x)$ is difficult to draw samples
- $\pi(x)$ can be evaluated
- $x^i \sim q(x), i = 1, ..., N_s$: Samples easily generated from *importance density* $q(\cdot)$
- Then,

$$p(x) \approx \sum_{i=1}^{N_s} w^i \delta(x - x^i)$$

where

$$w^i \propto \frac{\pi(x^i)}{q(x^i)}$$

- Now, in posterior pdf:

$$w_k^i \propto \frac{p(\mathbf{x}_{0:k}^i | \mathbf{z}_{1:k})}{q(\mathbf{x}_{0:k}^i | \mathbf{z}_{1:k})}$$

SIS Algorithm

- Choose

$$q(\mathbf{x}_{0:k}|\mathbf{z}_{1:k}) = q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k})q(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})$$

- Note:

$$\begin{aligned} p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k}) &= \frac{p(\mathbf{z}_k|\mathbf{x}_{0:k}|\mathbf{z}_{1:k-1})p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})} \\ &= \frac{p(\mathbf{z}_k|\mathbf{x}_{0:k}|\mathbf{z}_{1:k-1})p(\mathbf{x}_k|\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})} \\ &\quad \times p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1}) \\ &= \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})}p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1}) \\ &\propto p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1}) \end{aligned}$$

SIS Algorithm

- Then,

$$w_k^i \propto \frac{p(\mathbf{z}_k|\mathbf{x}_k^i)p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)p(\mathbf{x}_{0:k-1}^i|\mathbf{z}_{1:k-1})}{q(\mathbf{x}_k^i|\mathbf{x}_{0;k-1}^i,\mathbf{z}_{1:k})q(\mathbf{x}_{0:k-1}^i|\mathbf{z}_{1:k-1})}$$

$$= w_{k-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k^i)p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i|\mathbf{x}_{0:k-1}^i,\mathbf{z}_{1:k})}$$

- Furthermore, if

$$q(\mathbf{x}_k|\mathbf{x}_{0:k-1},\mathbf{z}_{1:k}) = q(\mathbf{x}_k|\mathbf{x}_{k-1},\mathbf{z}_k)$$

- Then,

$$w_k^i \propto w_{k-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k^i)p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i,\mathbf{z}_k)}$$

- Posterior filtered pdf: Discard $\mathbf{x}_{0:k-1}^i$ and $\mathbf{z}_{1:k-1}$

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^i)$$

Outline

SIS Particle Filter

`Algorithm 1: SIS Particle Filter`

$\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s} = \texttt{SIS}\left[\{\mathbf{x}_{k-1}^i, w_{k-1}^i\}_{i=1}^{N_s}, \mathbf{z}_k\right]$

- FOR $i = 1 : N_s$
  - `Draw` $\mathbf{x}_k^i \sim q(\mathbf{x}_k | \mathbf{x}_{k-1}^i, \mathbf{z}_k)$
  - `Assign the particle a weight,` w_k^i
- END FOR

Outline

SIS PF: Degeneracy Problem

- After a few iterations, all but onr particle will have negligible weight.
- Let

$$\hat{N}_{eff} = \frac{1}{\sum_{i=1}^{N_s} (w_k^i)^2}$$

- Small N_{eff} indicates severe degeneracy.
- Counter measures:
  - brute force: many, many samples N_s
  - good choice of importance density
  - resampling

SIS PF: Choice of Importance Density

- Choose $q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k)$ so that N_{eff} is maximized.
- Most common choice:

$$q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k) = p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$$

- Then,

$$w_k^i \propto w_{k-1}^i p(\mathbf{z}_k|\mathbf{x}_k^i)$$

SIS PF: Resampling

- Basic idea of resampling: Eliminate particles that have small weights and to concentrate on particles with large wights
- Generate a new set $\{\mathbf{x}_n^{i*}\}_{i=1}^{N_s}$ by resampling (with replacement) N_s times from approximate $p(\mathbf{x}_k|\mathbf{z}_{1:k})$

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^i)$$

  so that $\Pr(\mathbf{x}_k^{i*} = \mathbf{x}_k^j) = w_k^j$
- Complexity: possible in $O(N_s)$ operations

Algorithm 2: Resampling Algorithm

$\{\mathbf{x}_k^{j*}, w_k^j, i^j\}_{i=1}^{N_s}$ = RESAMPLE $[\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}, \mathbf{z}_k]$

- Initialize the CDF: $c_1 = 0$
- FOR $i = 2 : N_s$
    - Construct CDF $c_i = c_{i-1} + w_k^i$
- END FOR
- Start at the bottom of the CDF: $i = 1$
- Draw a starting point: $u_1 \sim U(0, N_s^{-1})$
- FOR $j = 1 : N_s$
    - Move along the CDF: $u_j = u_1 + N_s^{-1}(j-1)$
    - WHILE $u_j > c_i$
        - $i = i + 1$
    - END WHILE
    - Assign sample: $x_k^{j*} = x_k^i$
    - Assign weight: $w_k^j = N_s^{-1}$
    - Assign parent: $i^j = i$
- END FOR

Outline

Generic Particle Filter

Algorithm 3: Generic Particle Filter

$\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}$ = PF $[\{\mathbf{x}_{k-1}^i, w_{k-1}^i\}_{i=1}^{N_s}, \mathbf{z}_k]$

- FOR $i = 1 : N_s$
  - Draw $\mathbf{x}_k^i \sim q(\mathbf{x}_k | \mathbf{x}_{k-1}^i, \mathbf{z}_k)$
  - Assign the particle a weight, w_k^i
- END FOR
- Calculate total weight, $t =$SUM$[\{w_k^i\}_{i=1}^{N_s}]$
- FOR $i = 1 : N_s$
  - Normalize: $w_k^i \leftarrow t^{-1} w_k^i$
- END FOR
- Calculate $\hat{N}_{eff}$
- IF $\hat{N}_eff < N_T$
  - Resample: $\{\mathbf{x}_k^i, w_k^i, -\}_{i=1}^{N_s}$ = RESAMPLE $[\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}]$
- END IF

Summary

- If assumptions hold, Kalman or grid-based filters are optimum.
- Otherwise, approximate techniques needed.
  - EKF: approximates dynamics and measurement models to approximate pdf by Gaussian.
  - Approximate grid-based filters: approximate continuous state space as a set of discrete regions.
    - Computationally expensive for high dimensional spaces.
  - PF: Approximates pdf as a finite number of samples.
    - Choice of importance density
    - Resampling

For Further Reading

M. S. Arulampalam and S. Maskell and N. Gordon and T. Clapp
A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking
IEEE Transactions on Signal Processing, 2002.

Branko Ristic, Sanjeev Arulampalam, Neil Gordon
Beyond the Kalman filter: Particle filters for tracking applications
Artech House Publishers , 2004.