Lecture 9: Gaussian and Particle Smoothers

Simo Särkkä

March 13, 2019

Learning Outcomes

1. Summary of the Last Lecture
2. Extended and linearized smoothers
3. Gaussian RTS Smoothing
4. Particle Smoothing
5. Rao-Blackwellized Particle Smoothing
6. Summary and Demonstration

Summary of the Last Lecture

- Bayesian smoothing is used for computing estimates of state trajectories given the measurements on the whole trajectory.
- Rauch-Tung-Striebel (RTS) smoother is the closed form smoother for linear Gaussian models.
- RTSS is fixed-interval smoother, there are also fixed-point and fixed-lag smoothers.

Non-Linear Smoothing Problem

- Non-linear Gaussian state space model:

$$\begin{aligned}\mathbf{x}_k &= \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{q}_{k-1} \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k) + \mathbf{r}_k,\end{aligned}$$

- We want to compute Gaussian approximations to the smoothing distributions:

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:T}) \approx \mathrm{N}(\mathbf{x}_k \mid \mathbf{m}_k^s, \mathbf{P}_k^s).$$

Extended Rauch-Tung-Striebel Smoother Derivation

- The approximate joint distribution of $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ is

$$p(\mathbf{x}_k, \mathbf{x}_{k+1} \mid \mathbf{y}_{1:k}) = \mathrm{N}\left(\begin{bmatrix} \mathbf{x}_k \\ \mathbf{x}_{k+1} \end{bmatrix} \,\Big|\, \mathbf{m}_1, \mathbf{P}_1 \right),$$

where

$$\begin{aligned}
\mathbf{m}_1 &= \begin{pmatrix} \mathbf{m}_k \\ \mathbf{f}(\mathbf{m}_k) \end{pmatrix} \\
\mathbf{P}_1 &= \begin{pmatrix} \mathbf{P}_k & \mathbf{P}_k \, \mathbf{F}_x^T(\mathbf{m}_k) \\ \mathbf{F}_x(\mathbf{m}_k) \, \mathbf{P}_k & \mathbf{F}_x(\mathbf{m}_k) \, \mathbf{P}_k \, \mathbf{F}_x^T(\mathbf{m}_k) + \mathbf{Q}_k \end{pmatrix}.
\end{aligned}$$

- The rest of the derivation is analogous to the linear RTS smoother.

Extended Rauch-Tung-Striebel Smoother

Extended Rauch-Tung-Striebel Smoother

The equations for the extended RTS smoother are

$$\begin{aligned}
\mathbf{m}_{k+1}^{-} &= \mathbf{f}(\mathbf{m}_k) \\
\mathbf{P}_{k+1}^{-} &= \mathbf{F}_{\mathbf{x}}(\mathbf{m}_k)\,\mathbf{P}_k\,\mathbf{F}_{\mathbf{x}}^T(\mathbf{m}_k) + \mathbf{Q}_k \\
\mathbf{G}_k &= \mathbf{P}_k\,\mathbf{F}_{\mathbf{x}}^T(\mathbf{m}_k)\,[\mathbf{P}_{k+1}^{-}]^{-1} \\
\mathbf{m}_k^s &= \mathbf{m}_k + \mathbf{G}_k\,[\mathbf{m}_{k+1}^s - \mathbf{m}_{k+1}^{-}] \\
\mathbf{P}_k^s &= \mathbf{P}_k + \mathbf{G}_k\,[\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^{-}]\,\mathbf{G}_k^T,
\end{aligned}$$

where the matrix $\mathbf{F}_{\mathbf{x}}(\mathbf{m}_k)$ is the Jacobian matrix of $\mathbf{f}(\mathbf{x})$ evaluated at $\mathbf{m}_k$.

Statistically Linearized Rauch-Tung-Striebel Smoother Derivation

- With statistical linearization we get the approximation

$$p(\mathbf{x}_k, \mathbf{x}_{k+1} \mid \mathbf{y}_{1:k}) = \mathrm{N}\left(\begin{bmatrix} \mathbf{x}_k \\ \mathbf{x}_{k+1} \end{bmatrix} \,\middle|\, \mathbf{m}_1, \mathbf{P}_1 \right),$$

where

$$\mathbf{m}_1 = \begin{pmatrix} \mathbf{m}_k \\ \mathrm{E}[\mathbf{f}(\mathbf{x}_k)] \end{pmatrix}$$

$$\mathbf{P}_1 = \begin{pmatrix} \mathbf{P}_k & \mathrm{E}[\mathbf{f}(\mathbf{x}_k)\,\delta\mathbf{x}_k^T]^T \\ \mathrm{E}[\mathbf{f}(\mathbf{x}_k)\,\delta\mathbf{x}_k^T] & \mathrm{E}[\mathbf{f}(\mathbf{x}_k)\,\delta\mathbf{x}_k^T]\,\mathbf{P}_k^{-1}\,\mathrm{E}[\mathbf{f}(\mathbf{x}_k)\,\delta\mathbf{x}_k^T]^T + \mathbf{Q}_k \end{pmatrix}.$$

- The expectations are taken with respect to filtering distribution of $\mathbf{x}_k$.
- The derivation proceeds as with linear RTS smoother.

Statistically Linearized Rauch-Tung-Striebel Smoother

Statistically Linearized Rauch-Tung-Striebel Smoother

The equations for the statistically linearized RTS smoother are

$$\begin{aligned}
\mathbf{m}_{k+1}^- &= \mathrm{E}[\mathbf{f}(\mathbf{x}_k)] \\
\mathbf{P}_{k+1}^- &= \mathrm{E}[\mathbf{f}(\mathbf{x}_k)\,\delta\mathbf{x}_k^T]\,\mathbf{P}_k^{-1}\;\mathrm{E}[\mathbf{f}(\mathbf{x}_k)\,\delta\mathbf{x}_k^T]^T + \mathbf{Q}_k \\
\mathbf{G}_k &= \mathrm{E}[\mathbf{f}(\mathbf{x}_k)\,\delta\mathbf{x}_k^T]^T[\mathbf{P}_{k+1}^-]^{-1} \\
\mathbf{m}_k^s &= \mathbf{m}_k + \mathbf{G}_k\,[\mathbf{m}_{k+1}^s - \mathbf{m}_{k+1}^-] \\
\mathbf{P}_k^s &= \mathbf{P}_k + \mathbf{G}_k\,[\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^-]\,\mathbf{G}_k^T,
\end{aligned}$$

where the expectations are taken with respect to the filtering distribution $\mathbf{x}_k \sim \mathrm{N}(\mathbf{m}_k, \mathbf{P}_k)$.

Gaussian Rauch-Tung-Striebel Smoother Derivation

- With Gaussian moment matching we get the approximation

$$p(\mathbf{x}_k, \mathbf{x}_{k+1} \mid \mathbf{y}_{1:k}) = \mathrm{N}\left(\begin{bmatrix} \mathbf{x}_k \\ \mathbf{x}_{k+1} \end{bmatrix} \,\middle|\, \begin{bmatrix} \mathbf{m}_k \\ \mathbf{m}_{k+1}^- \end{bmatrix}, \begin{bmatrix} \mathbf{P}_k & \mathbf{D}_{k+1} \\ \mathbf{D}_{k+1}^T & \mathbf{P}_{k+1}^- \end{bmatrix} \right),$$

where

$$\begin{aligned}
\mathbf{m}_{k+1}^- &= \int \mathbf{f}(\mathbf{x}_k) \, \mathrm{N}(\mathbf{x}_k \mid \mathbf{m}_k, \mathbf{P}_k) \, d\mathbf{x}_k \\
\mathbf{P}_{k+1}^- &= \int [\mathbf{f}(\mathbf{x}_k) - \mathbf{m}_{k+1}^-] \, [\mathbf{f}(\mathbf{x}_k) - \mathbf{m}_{k+1}^-]^T \\
&\quad \times \mathrm{N}(\mathbf{x}_k \mid \mathbf{m}_k, \mathbf{P}_k) \, d\mathbf{x}_k + \mathbf{Q}_k \\
\mathbf{D}_{k+1} &= \int [\mathbf{x}_k - \mathbf{m}_k] \, [\mathbf{f}(\mathbf{x}_k) - \mathbf{m}_{k+1}^-]^T \mathrm{N}(\mathbf{x}_k \mid \mathbf{m}_k, \mathbf{P}_k) \, d\mathbf{x}_k.
\end{aligned}$$

Gaussian Rauch-Tung-Striebel Smoother

The equations for the Gaussian RTS smoother are

$$
\begin{aligned}
\mathbf{m}_{k+1}^{-} &= \int \mathbf{f}(\mathbf{x}_k)\,\mathrm{N}(\mathbf{x}_k \,|\, \mathbf{m}_k, \mathbf{P}_k)\, d\mathbf{x}_k \\
\mathbf{P}_{k+1}^{-} &= \int [\mathbf{f}(\mathbf{x}_k) - \mathbf{m}_{k+1}^{-}]\,[\mathbf{f}(\mathbf{x}_k) - \mathbf{m}_{k+1}^{-}]^T \\
&\qquad \times \mathrm{N}(\mathbf{x}_k \,|\, \mathbf{m}_k, \mathbf{P}_k)\, d\mathbf{x}_k + \mathbf{Q}_k \\
\mathbf{D}_{k+1} &= \int [\mathbf{x}_k - \mathbf{m}_k]\,[\mathbf{f}(\mathbf{x}_k) - \mathbf{m}_{k+1}^{-}]^T \mathrm{N}(\mathbf{x}_k \,|\, \mathbf{m}_k, \mathbf{P}_k)\, d\mathbf{x}_k \\
\mathbf{G}_k &= \mathbf{D}_{k+1}\,[\mathbf{P}_{k+1}^{-}]^{-1} \\
\mathbf{m}_k^s &= \mathbf{m}_k + \mathbf{G}_k\,(\mathbf{m}_{k+1}^s - \mathbf{m}_{k+1}^{-}) \\
\mathbf{P}_k^s &= \mathbf{P}_k + \mathbf{G}_k\,(\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^{-})\,\mathbf{G}_k^T.
\end{aligned}
$$

Cubature Smoother Derivation [1/2]

- Recall the 3rd order spherical Gaussian integral rule:

$$\int \mathbf{g}(\mathbf{x}) \, \mathrm{N}(\mathbf{x} \,|\, \mathbf{m}, \mathbf{P}) \, d\mathbf{x} \approx \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{g}(\mathbf{m} + \sqrt{\mathbf{P}} \, \boldsymbol{\xi}^{(i)}),$$

where

$$\boldsymbol{\xi}^{(i)} = \begin{cases} \sqrt{n} \, \mathbf{e}_i & , \quad i = 1, \ldots, n \\ -\sqrt{n} \, \mathbf{e}_{i-n} & , \quad i = n+1, \ldots, 2n, \end{cases}$$

where $\mathbf{e}_i$ denotes a unit vector to the direction of coordinate axis i.

Cubature Smoother Derivation [2/2]

- We get the approximation

$$p(\mathbf{x}_k, \mathbf{x}_{k+1} \mid \mathbf{y}_{1:k}) = \mathrm{N}\left(\begin{bmatrix} \mathbf{x}_k \\ \mathbf{x}_{k+1} \end{bmatrix} \middle| \begin{bmatrix} \mathbf{m}_k \\ \mathbf{m}_{k+1}^- \end{bmatrix}, \begin{bmatrix} \mathbf{P}_k & \mathbf{D}_{k+1} \\ \mathbf{D}_{k+1}^T & \mathbf{P}_{k+1}^- \end{bmatrix} \right),$$

where

$$\begin{aligned}
\mathcal{X}_k^{(i)} &= \mathbf{m}_k + \sqrt{\mathbf{P}_k}\, \xi^{(i)} \\
\mathbf{m}_{k+1}^- &= \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{f}(\mathcal{X}_k^{(i)}) \\
\mathbf{P}_{k+1}^- &= \frac{1}{2n} \sum_{i=1}^{2n} [\mathbf{f}(\mathcal{X}_k^{(i)}) - \mathbf{m}_{k+1}^-]\, [\mathbf{f}(\mathcal{X}_k^{(i)}) - \mathbf{m}_{k+1}^-]^T + \mathbf{Q}_k \\
\mathbf{D}_{k+1} &= \frac{1}{2n} \sum_{i=1}^{2n} [\mathcal{X}_k^{(i)} - \mathbf{m}_k]\, [\mathbf{f}(\mathcal{X}_k^{(i)}) - \mathbf{m}_{k+1}^-]^T.
\end{aligned}$$

Cubature Rauch-Tung-Striebel Smoother [1/3]

Cubature Rauch-Tung-Striebel Smoother

1. Form the sigma points:

$$\mathcal{X}_k^{(i)} = \mathbf{m}_k + \sqrt{\mathbf{P}_k}\, \xi^{(i)}, \qquad i = 1, \ldots, 2n,$$

   where the unit sigma points are defined as

$$\xi^{(i)} = \begin{cases} \sqrt{n}\, \mathbf{e}_i & , \quad i = 1, \ldots, n \\ -\sqrt{n}\, \mathbf{e}_{i-n} & , \quad i = n+1, \ldots, 2n. \end{cases}$$

2. Propagate the sigma points through the dynamic model:

$$\hat{\mathcal{X}}_{k+1}^{(i)} = \mathbf{f}(\mathcal{X}_k^{(i)}), \quad i = 1, \ldots, 2n.$$

Cubature Rauch-Tung-Striebel Smoother [2/3]

Cubature Rauch-Tung-Striebel Smoother (cont.)

3. Compute the predicted mean $\mathbf{m}_{k+1}^-$, the predicted covariance $\mathbf{P}_{k+1}^-$ and the cross-covariance $\mathbf{D}_{k+1}$:

$$\mathbf{m}_{k+1}^- = \frac{1}{2n} \sum_{i=1}^{2n} \hat{\mathcal{X}}_{k+1}^{(i)}$$

$$\mathbf{P}_{k+1}^- = \frac{1}{2n} \sum_{i=1}^{2n} (\hat{\mathcal{X}}_{k+1}^{(i)} - \mathbf{m}_{k+1}^-) (\hat{\mathcal{X}}_{k+1}^{(i)} - \mathbf{m}_{k+1}^-)^T + \mathbf{Q}_k$$

$$\mathbf{D}_{k+1} = \frac{1}{2n} \sum_{i=1}^{2n} (\mathcal{X}_k^{(i)} - \mathbf{m}_k) (\hat{\mathcal{X}}_{k+1}^{(i)} - \mathbf{m}_{k+1}^-)^T.$$

Cubature Rauch-Tung-Striebel Smoother [3/3]

Cubature Rauch-Tung-Striebel Smoother (cont.)

4. Compute the gain $\mathbf{G}_k$, mean $\mathbf{m}_k^s$ and covariance $\mathbf{P}_k^s$ as follows:

$$\begin{aligned}
\mathbf{G}_k &= \mathbf{D}_{k+1} \, [\mathbf{P}_{k+1}^-]^{-1} \\
\mathbf{m}_k^s &= \mathbf{m}_k + \mathbf{G}_k \, (\mathbf{m}_{k+1}^s - \mathbf{m}_{k+1}^-) \\
\mathbf{P}_k^s &= \mathbf{P}_k + \mathbf{G}_k \, (\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^-) \, \mathbf{G}_k^T.
\end{aligned}$$

Unscented Rauch-Tung-Striebel Smoother [1/3]

Unscented Rauch-Tung-Striebel Smoother

1. Form the sigma points:

$$\begin{aligned}
\mathcal{X}_k^{(0)} &= \mathbf{m}_k, \\
\mathcal{X}_k^{(i)} &= \mathbf{m}_k + \sqrt{n+\lambda} \left[\sqrt{\mathbf{P}_k}\right]_i \\
\mathcal{X}_k^{(i+n)} &= \mathbf{m}_k - \sqrt{n+\lambda} \left[\sqrt{\mathbf{P}_k}\right]_i, \quad i = 1, \ldots, n.
\end{aligned}$$

2. Propagate the sigma points through the dynamic model:

$$\hat{\mathcal{X}}_{k+1}^{(i)} = \mathbf{f}(\mathcal{X}_k^{(i)}), \quad i = 0, \ldots, 2n.$$

Unscented Rauch-Tung-Striebel Smoother (cont.)

3. Compute predicted mean, covariance and cross-covariance:

$$\mathbf{m}_{k+1}^{-} = \sum_{i=0}^{2n} W_i^{(m)} \, \hat{\mathcal{X}}_{k+1}^{(i)}$$

$$\mathbf{P}_{k+1}^{-} = \sum_{i=0}^{2n} W_i^{(c)} \, (\hat{\mathcal{X}}_{k+1}^{(i)} - \mathbf{m}_{k+1}^{-}) \, (\hat{\mathcal{X}}_{k+1}^{(i)} - \mathbf{m}_{k+1}^{-})^T + \mathbf{Q}_k$$

$$\mathbf{D}_{k+1} = \sum_{i=0}^{2n} W_i^{(c)} \, (\mathcal{X}_k^{(i)} - \mathbf{m}_k) \, (\hat{\mathcal{X}}_{k+1}^{(i)} - \mathbf{m}_{k+1}^{-})^T,$$

Unscented Rauch-Tung-Striebel Smoother (cont.)

4. Compute gain smoothed mean and smoothed covariance: as follows:

$$
\begin{aligned}
\mathbf{G}_k &= \mathbf{D}_{k+1} \, [\mathbf{P}^-_{k+1}]^{-1} \\
\mathbf{m}^s_k &= \mathbf{m}_k + \mathbf{G}_k \, (\mathbf{m}^s_{k+1} - \mathbf{m}^-_{k+1}) \\
\mathbf{P}^s_k &= \mathbf{P}_k + \mathbf{G}_k \, (\mathbf{P}^s_{k+1} - \mathbf{P}^-_{k+1}) \, \mathbf{G}^T_k.
\end{aligned}
$$

Other Gaussian RTS Smoothers

- Gauss-Hermite RTS smoother is based on multidimensional Gauss-Hermite integration.
- Bayes-Hermite or Gaussian Process RTS smoother uses Gaussian process based quadrature (Bayes-Hermite).
- Monte Carlo integration based RTS smoothers.
- Central differences etc.

Particle Smoothing: Direct SIR

- The smoothing solution can be obtained from SIR by storing the whole state histories into the particles.
- Special care is needed on the resampling step.
- The smoothed distribution approximation is then of the form

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:T}) \approx \sum_{i=1}^{N} w_T^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}),$$

  where $\mathbf{x}_k^{(i)}$ is the kth component in $\mathbf{x}_{1:T}^{(i)}$.
- Unfortunately, the approximation is often quite degenerate.

Particle Smoothing: Backward Simulation [1/2]

- In backward-simulation particle smoother we simulate individual trajectories backwards.
- The simulated samples are drawn from the particle filter samples.
- Uses the previous filtering results in smoothing $\Rightarrow$ less degenerate than the direct SIR smoother.
- Idea:
  - Assume now that we have already simulated $\tilde{\mathbf{x}}_{k+1:T}$ from the smoothing distribution.
  - From the Bayesian smoothing equations we get

$$p(\mathbf{x}_k \mid \tilde{\mathbf{x}}_{k+1}, \mathbf{y}_{1:T}) \propto p(\tilde{\mathbf{x}}_{k+1} \mid \mathbf{x}_k) \, p(\mathbf{x}_k \mid \mathbf{y}_{1:k}).$$

Particle Smoothing: Backward Simulation [2/2]

Backward simulation particle smoother

Given the weighted set of particles $\{w_k^{(i)}, \mathbf{x}_k^{(i)}\}$ representing the filtering distributions:

- Choose $\tilde{\mathbf{x}}_T = \mathbf{x}_T^{(i)}$ with probability $w_T^{(i)}$.
- For $k = T-1, \ldots, 0$:
  1. Compute new weights by
$$w_{k|k+1}^{(i)} \propto w_k^{(i)} p(\tilde{\mathbf{x}}_{k+1} \mid \mathbf{x}_k^{(i)})$$
  2. Choose $\tilde{\mathbf{x}}_k = \mathbf{x}_k^{(i)}$ with probability $w_{k|k+1}^{(i)}$

Given S iterations resulting in $\tilde{\mathbf{x}}_{1:T}^{(j)}$ for $j = 1, \ldots, S$ the smoothing distribution approximation is

$$p(\mathbf{x}_{1:T} \mid \mathbf{y}_{1:T}) \approx \frac{1}{S} \sum_j \delta(\mathbf{x}_{1:T} - \tilde{\mathbf{x}}_{1:T}^{(j)}).$$

- The reweighting particle smoother is based on computing new weights $w_{k+1|T}^{(i)}$ for the SIR filter particles such that:

$$p(\mathbf{x}_{k+1} \mid \mathbf{y}_{1:T}) \approx \sum_i w_{k+1|T}^{(i)} \, \delta(\mathbf{x}_{k+1} - \mathbf{x}_{k+1}^{(i)}).$$

- Recall the smoothing equation

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:T}) = p(\mathbf{x}_k \mid \mathbf{y}_{1:k}) \int \left[\frac{p(\mathbf{x}_{k+1} \mid \mathbf{x}_k) \, p(\mathbf{x}_{k+1} \mid \mathbf{y}_{1:T})}{p(\mathbf{x}_{k+1} \mid \mathbf{y}_{1:k})} \right] \mathrm{d}\mathbf{x}_{k+1}$$

- We use SIR filter samples to form approximations as follows:

$$\int \frac{p(\mathbf{x}_{k+1} \mid \mathbf{x}_k) \, p(\mathbf{x}_{k+1} \mid \mathbf{y}_{1:T})}{p(\mathbf{x}_{k+1} \mid \mathbf{y}_{1:k})} \, d\mathbf{x}_{k+1} \approx \sum_i w_{k+1|T}^{(i)} \frac{p(\mathbf{x}_{k+1}^{(i)} \mid \mathbf{x}_k)}{p(\mathbf{x}_{k+1}^{(i)} \mid \mathbf{y}_{1:k})}$$

$$p(\mathbf{x}_{k+1} \mid \mathbf{y}_{1:k}) \approx \sum_j w_k^{(j)} \, p(\mathbf{x}_{k+1} \mid \mathbf{x}_k^{(j)})$$

Particle Smoothing: Reweighting [2/2]

Reweighting Particle Smoother

Given the weighted set of particles $\{w_k^{(i)}, x_k^{(i)}\}$ representing the filtering distribution, we can form approximations to the marginal smoothing distributions as follows:

- Start by setting $w_{T|T}^{(i)} = w_T^{(i)}$ for $i = 1, \ldots, n$.
- For each $k = T-1, \ldots, 0$ do the following:
  - Compute new importance weights by

$$w_{k|T}^{(i)} \propto \sum_j w_{k+1|T}^{(j)} \frac{w_k^{(i)} \, p(\mathbf{x}_{k+1}^{(j)} \mid \mathbf{x}_k^{(i)})}{\left[\sum_l w_k^{(l)} \, p(\mathbf{x}_{k+1}^{(j)} \mid \mathbf{x}_k^{(l)})\right]}.$$

At each step k the marginal smoothing distribution can be approximated as

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:T}) \approx \sum_i w_{k|T}^{(i)} \, \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}).$$

Rao-Blackwellized Particle Smoothing: Direct SIR

- Recall e.g. the hierarchical the Rao-Blackwellized particle filtering model:

$$\begin{aligned}
\mathbf{u}_k &\sim p(\mathbf{u}_k \mid \mathbf{u}_{k-1}) \\
\mathbf{x}_k &= \mathbf{A}(\mathbf{u}_{k-1})\,\mathbf{x}_{k-1} + \mathbf{q}_k, \qquad \mathbf{q}_k \sim \mathrm{N}(\mathbf{0}, \mathbf{Q}) \\
\mathbf{y}_k &= \mathbf{H}(\mathbf{u}_k)\,\mathbf{x}_k + \mathbf{r}_k, \qquad \mathbf{r}_k \sim \mathrm{N}(\mathbf{0}, \mathbf{R})
\end{aligned}$$

- The direct SIR based Rao-Blackwellized particle smoother:
  1. During filtering store the whole sampled state and Kalman filter histories to the particles.
  2. At the smoothing step, apply Rauch-Tung-Striebel smoothers to each of the Kalman filter histories.
- The smoothing distribution approximation:

$$p(\mathbf{x}_k, \mathbf{u}_k \mid \mathbf{y}_{1:T}) \approx \sum_{i=1}^{N} w_T^{(i)}\, \delta(\mathbf{u}_k - \mathbf{u}_k^{(i)})\; \mathrm{N}(\mathbf{x}_k \mid \mathbf{m}_k^{s,(i)}, \mathbf{P}_k^{s,(i)}).$$

- Also has the degeneracy problem.

Rao-Blackwellized Particle Smoothing: Other Types

- The RB backward-sampling smoother can be implemented in many ways:
  - Sample both the components backwards (leads to a pure sample representation).
  - Sample the latent variables only – requires quite complicated backward Kalman filtering computations.
  - Kim's approximation: just use the plain backward-sampling to the latent variable marginal.
- The RB reweighting particle smoothing is not possible exactly, but can be approximated using the above ideas.

Summary

- Extended, statistically linearized and unscented RTS smoothers are the approximate nonlinear smoothers corresponding to EKF, SLF and UKF.
- Gaussian RTS smoothers: cubature RTS smoother, Gauss-Hermite RTS smoothers and various others
- Particle smoothing can be done by storing the whole state histories in SIR algorithm.
- Rao-Blackwellized particle smoother is a combination of particle smoothing and RTS smoothing.

Matlab Demo: Pendulum [1/2]

- Pendulum model:

$$\begin{pmatrix} x_k^1 \\ x_k^2 \end{pmatrix} = \underbrace{\begin{pmatrix} x_{k-1}^1 + x_{k-1}^2 \, \Delta t \\ x_{k-1}^2 - g \, \sin(x_{k-1}^1) \, \Delta t \end{pmatrix}}_{\mathbf{f}(\mathbf{x_{k-1}})} + \begin{pmatrix} 0 \\ q_{k-1} \end{pmatrix}$$

$$y_k = \underbrace{\sin(x_k^1)}_{\mathbf{h}(\mathbf{x_k})} + r_k,$$

- The required Jacobian matrix for ERTSS:

$$\mathbf{F}_x(\mathbf{x}) = \begin{pmatrix} 1 & \Delta t \\ -g \, \cos(x^1) \, \Delta t & 1 \end{pmatrix}$$

- The required expected value for SLRTSS is

$$\mathrm{E}[\mathbf{f}(\mathbf{x})] = \begin{pmatrix} m_1 + m_2\,\Delta t \\ m_2 - g\,\sin(m_1)\,\exp(-P_{11}/2)\,\Delta t \end{pmatrix}$$

- And the cross term:

$$\mathrm{E}[\mathbf{f}(\mathbf{x})\,(\mathbf{x}-\mathbf{m})^T] = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix},$$

where

$$\begin{aligned} c_{11} &= P_{11} + \Delta t\,P_{12} \\ c_{12} &= P_{12} + \Delta t\,P_{22} \\ c_{21} &= P_{12} - g\,\Delta t\,\cos(m_1)\,P_{11}\,\exp(-P_{11}/2) \\ c_{22} &= P_{22} - g\,\Delta t\,\cos(m_1)\,P_{12}\,\exp(-P_{11}/2) \end{aligned}$$