

# Lecture 7: Optimal Smoothing

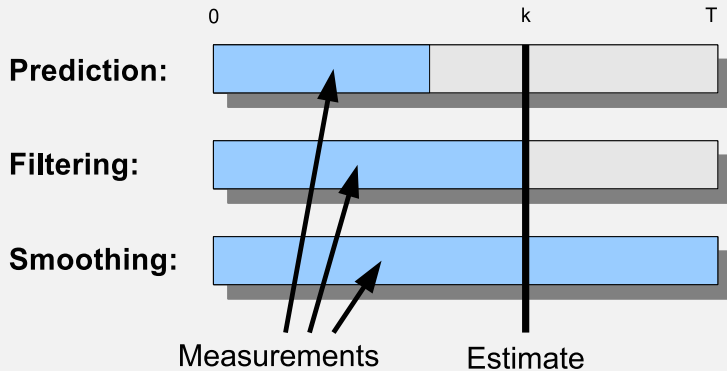
Simo Särkkä

Department of Biomedical Engineering and Computational Science  
Aalto University

March 17, 2011

- 1 What is Optimal Smoothing?
- 2 Bayesian Optimal Smoothing Equations
- 3 Rauch-Tung-Striebel Smoother
- 4 Gaussian Approximation Based Smoothing
- 5 Particle Smoothing
- 6 Summary and Demonstration

# Filtering, Prediction and Smoothing



# Types of Smoothing Problems

- **Fixed-interval smoothing**: estimate states on interval  $[0, T]$  given measurements on the same interval.
- **Fixed-point smoothing**: estimate state at a fixed point of time in the past.
- **Fixed-lag smoothing**: estimate state at a fixed delay in the past.
- Here we shall only consider fixed-interval smoothing, the others can be quite easily derived from it.

# Examples of Smoothing Problems

- Given all the radar measurements of a rocket (or missile) trajectory, what was the **exact place of launch**?
- Estimate the whole trajectory of a car based on GPS measurements to **calibrate the inertial navigation system** accurately.
- What was the history of **chemical/combustion/other process** given a batch of measurements from it?
- **Remove noise from audio signal** by using smoother to estimate the true audio signal under the noise.
- Smoothing solution also arises in EM algorithm for **estimating the parameters of a state space model**.

- Linear Gaussian models
  - Rauch-Tung-Striebel smoother (RTSS).
  - Two-filter smoother.
- Non-linear Gaussian models
  - Extended Rauch-Tung-Striebel smoother (ERTSS).
  - Unscented Rauch-Tung-Striebel smoother (URTSS).
  - Statistically linearized Rauch-Tung-Striebel smoother (URTSS).
  - Gaussian Rauch-Tung-Striebel smoothers (GRTSS), cubature, Gauss-Hermite, Bayes-Hermite, Monte Carlo.
  - Two-filter versions of the above.
- Non-linear non-Gaussian models
  - Sequential importance resampling based smoother.
  - Rao-Blackwellized particle smoothers.
  - Grid based smoother.

# Problem Formulation

- Probabilistic state space model:

measurement model:  $\mathbf{y}_k \sim p(\mathbf{y}_k | \mathbf{x}_k)$

dynamic model:  $\mathbf{x}_k \sim p(\mathbf{x}_k | \mathbf{x}_{k-1})$

- Assume that the filtering distributions  $p(\mathbf{x}_k | \mathbf{y}_{1:k})$  have already been computed for all  $k = 0, \dots, T$ .
- We want **recursive equations** of computing the smoothing distribution for all  $k < T$ :

$$p(\mathbf{x}_k | \mathbf{y}_{1:T}).$$

- The **recursion** will go **backwards in time**, because on the last step, the filtering and smoothing distributions coincide:

$$p(\mathbf{x}_T | \mathbf{y}_{1:T}).$$

# Derivation of Formal Smoothing Equations [1/2]

- **The key:** due to the Markov properties of state we have:

$$p(\mathbf{x}_k | \mathbf{x}_{k+1}, \mathbf{y}_{1:T}) = p(\mathbf{x}_k | \mathbf{x}_{k+1}, \mathbf{y}_{1:k})$$

- Thus we get:

$$\begin{aligned} p(\mathbf{x}_k | \mathbf{x}_{k+1}, \mathbf{y}_{1:T}) &= p(\mathbf{x}_k | \mathbf{x}_{k+1}, \mathbf{y}_{1:k}) \\ &= \frac{p(\mathbf{x}_k, \mathbf{x}_{k+1} | \mathbf{y}_{1:k})}{p(\mathbf{x}_{k+1} | \mathbf{y}_{1:k})} \\ &= \frac{p(\mathbf{x}_{k+1} | \mathbf{x}_k, \mathbf{y}_{1:k}) p(\mathbf{x}_k | \mathbf{y}_{1:k})}{p(\mathbf{x}_{k+1} | \mathbf{y}_{1:k})} \\ &= \frac{p(\mathbf{x}_{k+1} | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{y}_{1:k})}{p(\mathbf{x}_{k+1} | \mathbf{y}_{1:k})}. \end{aligned}$$



# Derivation of Formal Smoothing Equations [2/2]

- Assuming that the **smoothing distribution of the next step**  $p(\mathbf{x}_{k+1} | \mathbf{y}_{1:T})$  is available, we get

$$\begin{aligned} p(\mathbf{x}_k, \mathbf{x}_{k+1} | \mathbf{y}_{1:T}) &= p(\mathbf{x}_k | \mathbf{x}_{k+1}, \mathbf{y}_{1:T}) p(\mathbf{x}_{k+1} | \mathbf{y}_{1:T}) \\ &= p(\mathbf{x}_k | \mathbf{x}_{k+1}, \mathbf{y}_{1:k}) p(\mathbf{x}_{k+1} | \mathbf{y}_{1:T}) \\ &= \frac{p(\mathbf{x}_{k+1} | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{y}_{1:k}) p(\mathbf{x}_{k+1} | \mathbf{y}_{1:T})}{p(\mathbf{x}_{k+1} | \mathbf{y}_{1:k})} \end{aligned}$$

- Integrating over  $\mathbf{x}_{k+1}$**  gives

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

## Bayesian Optimal Smoothing Equations

The **Bayesian optimal smoothing equations** consist of **prediction step** and **backward update step**:

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

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

The recursion is started from the filtering (and smoothing) distribution of the last time step  $p(\mathbf{x}_T | \mathbf{y}_{1:T})$ .

# Linear-Gaussian Smoothing Problem

- Gaussian driven **linear model**, i.e., **Gauss-Markov model**:

$$\mathbf{x}_k = \mathbf{A}_{k-1} \mathbf{x}_{k-1} + \mathbf{q}_{k-1}$$

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{r}_k,$$

- In **probabilistic terms** the model is

$$p(\mathbf{x}_k | \mathbf{x}_{k-1}) = N(\mathbf{x}_k | \mathbf{A}_{k-1} \mathbf{x}_{k-1}, \mathbf{Q}_{k-1})$$

$$p(\mathbf{y}_k | \mathbf{x}_k) = N(\mathbf{y}_k | \mathbf{H}_k \mathbf{x}_k, \mathbf{R}_k).$$

- **Kalman filter** can be used for computing all the Gaussian filtering distributions:

$$p(\mathbf{x}_k | \mathbf{y}_{1:k}) = N(\mathbf{x}_k | \mathbf{m}_k, \mathbf{P}_k).$$

- Gaussian probability density

$$N(\mathbf{x} \mid \mathbf{m}, \mathbf{P}) = \frac{1}{(2\pi)^{n/2} |\mathbf{P}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{m})^T \mathbf{P}^{-1} (\mathbf{x} - \mathbf{m})\right),$$

- Let  $\mathbf{x}$  and  $\mathbf{y}$  have the Gaussian densities

$$p(\mathbf{x}) = N(\mathbf{x} \mid \mathbf{m}, \mathbf{P}), \quad p(\mathbf{y} \mid \mathbf{x}) = N(\mathbf{y} \mid \mathbf{H}\mathbf{x}, \mathbf{R}),$$

- Then the joint and marginal distributions are

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \sim N\left(\begin{pmatrix} \mathbf{m} \\ \mathbf{H}\mathbf{m} \end{pmatrix}, \begin{pmatrix} \mathbf{P} & \mathbf{P}\mathbf{H}^T \\ \mathbf{H}\mathbf{P} & \mathbf{H}\mathbf{P}\mathbf{H}^T + \mathbf{R} \end{pmatrix}\right)$$
$$\mathbf{y} \sim N(\mathbf{H}\mathbf{m}, \mathbf{H}\mathbf{P}\mathbf{H}^T + \mathbf{R}).$$

- If the random variables  $\mathbf{x}$  and  $\mathbf{y}$  have the joint Gaussian probability density

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \sim N \left( \begin{pmatrix} \mathbf{a} \\ \mathbf{b} \end{pmatrix}, \begin{pmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C}^T & \mathbf{B} \end{pmatrix} \right),$$

- Then the marginal and conditional densities of  $\mathbf{x}$  and  $\mathbf{y}$  are given as follows:

$$\mathbf{x} \sim N(\mathbf{a}, \mathbf{A})$$

$$\mathbf{y} \sim N(\mathbf{b}, \mathbf{B})$$

$$\mathbf{x} | \mathbf{y} \sim N(\mathbf{a} + \mathbf{C}\mathbf{B}^{-1}(\mathbf{y} - \mathbf{b}), \mathbf{A} - \mathbf{C}\mathbf{B}^{-1}\mathbf{C}^T)$$

$$\mathbf{y} | \mathbf{x} \sim N(\mathbf{b} + \mathbf{C}^T\mathbf{A}^{-1}(\mathbf{x} - \mathbf{a}), \mathbf{B} - \mathbf{C}^T\mathbf{A}^{-1}\mathbf{C}).$$

- By the **Gaussian distribution computation rules** we get

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

where

$$\mathbf{m}_1 = \begin{pmatrix} \mathbf{m}_k \\ \mathbf{A}_k \mathbf{m}_k \end{pmatrix}, \quad \mathbf{P}_1 = \begin{pmatrix} \mathbf{P}_k & \mathbf{P}_k \mathbf{A}_k^T \\ \mathbf{A}_k \mathbf{P}_k & \mathbf{A}_k \mathbf{P}_k \mathbf{A}_k^T + \mathbf{Q}_k \end{pmatrix}.$$

- By **conditioning rule** of Gaussian distribution we get

$$\begin{aligned} p(\mathbf{x}_k | \mathbf{x}_{k+1}, \mathbf{y}_{1:T}) &= p(\mathbf{x}_k | \mathbf{x}_{k+1}, \mathbf{y}_{1:k}) \\ &= \mathbf{N}(\mathbf{x}_k | \mathbf{m}_2, \mathbf{P}_2), \end{aligned}$$

where

$$\begin{aligned} \mathbf{G}_k &= \mathbf{P}_k \mathbf{A}_k^T (\mathbf{A}_k \mathbf{P}_k \mathbf{A}_k^T + \mathbf{Q}_k)^{-1} \\ \mathbf{m}_2 &= \mathbf{m}_k + \mathbf{G}_k (\mathbf{x}_{k+1} - \mathbf{A}_k \mathbf{m}_k) \\ \mathbf{P}_2 &= \mathbf{P}_k - \mathbf{G}_k (\mathbf{A}_k \mathbf{P}_k \mathbf{A}_k^T + \mathbf{Q}_k) \mathbf{G}_k^T. \end{aligned}$$

# Derivation of Rauch-Tung-Striebel Smoother [3/4]

- The **joint distribution of  $\mathbf{x}_k$  and  $\mathbf{x}_{k+1}$**  given all the data is

$$\begin{aligned} p(\mathbf{x}_{k+1}, \mathbf{x}_k \mid \mathbf{y}_{1:T}) &= p(\mathbf{x}_k \mid \mathbf{x}_{k+1}, \mathbf{y}_{1:T}) p(\mathbf{x}_{k+1} \mid \mathbf{y}_{1:T}) \\ &= N(\mathbf{x}_k \mid \mathbf{m}_2, \mathbf{P}_2) N(\mathbf{x}_{k+1} \mid \mathbf{m}_{k+1}^s, \mathbf{P}_{k+1}^s) \\ &= N\left(\begin{bmatrix} \mathbf{x}_{k+1} \\ \mathbf{x}_k \end{bmatrix} \mid \mathbf{m}_3, \mathbf{P}_3\right) \end{aligned}$$

where

$$\begin{aligned} \mathbf{m}_3 &= \begin{pmatrix} \mathbf{m}_{k+1}^s \\ \mathbf{m}_k + \mathbf{G}_k (\mathbf{m}_{k+1}^s - \mathbf{A}_k \mathbf{m}_k) \end{pmatrix} \\ \mathbf{P}_3 &= \begin{pmatrix} \mathbf{P}_{k+1}^s & \mathbf{P}_{k+1}^s \mathbf{G}_k^T \\ \mathbf{G}_k \mathbf{P}_{k+1}^s & \mathbf{G}_k \mathbf{P}_{k+1}^s \mathbf{G}_k^T + \mathbf{P}_2 \end{pmatrix}. \end{aligned}$$



- The **marginal mean and covariance** are thus given as

$$\mathbf{m}_k^s = \mathbf{m}_k + \mathbf{G}_k (\mathbf{m}_{k+1}^s - \mathbf{A}_k \mathbf{m}_k)$$

$$\mathbf{P}_k^s = \mathbf{P}_k + \mathbf{G}_k (\mathbf{P}_{k+1}^s - \mathbf{A}_k \mathbf{P}_k \mathbf{A}_k^T - \mathbf{Q}_k) \mathbf{G}_k^T.$$

- The **smoothing distribution** is then Gaussian with the above mean and covariance:

$$p(\mathbf{x}_k | \mathbf{y}_{1:T}) = \mathcal{N}(\mathbf{x}_k | \mathbf{m}_k^s, \mathbf{P}_k^s),$$

## Rauch-Tung-Striebel Smoother

**Backward recursion equations** for the smoothed means  $\mathbf{m}_k^s$  and covariances  $\mathbf{P}_k^s$ :

$$\mathbf{m}_{k+1}^- = \mathbf{A}_k \mathbf{m}_k$$

$$\mathbf{P}_{k+1}^- = \mathbf{A}_k \mathbf{P}_k \mathbf{A}_k^T + \mathbf{Q}_k$$

$$\mathbf{G}_k = \mathbf{P}_k \mathbf{A}_k^T [\mathbf{P}_{k+1}^-]^{-1}$$

$$\mathbf{m}_k^s = \mathbf{m}_k + \mathbf{G}_k [\mathbf{m}_{k+1}^s - \mathbf{m}_{k+1}^-]$$

$$\mathbf{P}_k^s = \mathbf{P}_k + \mathbf{G}_k [\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^-] \mathbf{G}_k^T,$$

- $\mathbf{m}_k$  and  $\mathbf{P}_k$  are the mean and covariance computed by the **Kalman filter**.
- The recursion is **started from the last time step**  $T$ , with  $\mathbf{m}_T^s = \mathbf{m}_T$  and  $\mathbf{P}_T^s = \mathbf{P}_T$ .

# RTS Smoother: Car Tracking Example

The **dynamic model of the car tracking model** from the first & third lectures was:

$$\begin{pmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\mathbf{A}} \begin{pmatrix} x_{k-1} \\ y_{k-1} \\ \dot{x}_{k-1} \\ \dot{y}_{k-1} \end{pmatrix} + \mathbf{q}_{k-1}$$

where  $\mathbf{q}_k$  is zero mean with a covariance matrix  $\mathbf{Q}$ .

$$\mathbf{Q} = \begin{pmatrix} q_1^c \Delta t^3 / 3 & 0 & q_1^c \Delta t^2 / 2 & 0 \\ 0 & q_2^c \Delta t^3 / 3 & 0 & q_2^c \Delta t^2 / 2 \\ q_1^c \Delta t^2 / 2 & 0 & q_1^c \Delta t & 0 \\ 0 & q_2^c \Delta t^2 / 2 & 0 & q_2^c \Delta t \end{pmatrix}$$

- **Non-linear Gaussian** state space model:

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{q}_{k-1}$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{r}_k,$$

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

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

# Extended Rauch-Tung-Striebel Smoother Derivation

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

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

where

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

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

## Extended Rauch-Tung-Striebel Smoother

The equations for the extended RTS smoother are

$$\mathbf{m}_{k+1}^- = \mathbf{f}(\mathbf{m}_k)$$

$$\mathbf{P}_{k+1}^- = \mathbf{F}_x(\mathbf{m}_k) \mathbf{P}_k \mathbf{F}_x^T(\mathbf{m}_k) + \mathbf{Q}_k$$

$$\mathbf{G}_k = \mathbf{P}_k \mathbf{F}_x^T(\mathbf{m}_k) [\mathbf{P}_{k+1}^-]^{-1}$$

$$\mathbf{m}_k^s = \mathbf{m}_k + \mathbf{G}_k [\mathbf{m}_{k+1}^s - \mathbf{m}_{k+1}^-]$$

$$\mathbf{P}_k^s = \mathbf{P}_k + \mathbf{G}_k [\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^-] \mathbf{G}_k^T,$$

where the matrix  $\mathbf{F}_x(\mathbf{m}_k)$  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

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

where

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

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

- 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

The equations for the statistically linearized RTS smoother are

$$\mathbf{m}_{k+1}^- = \mathbf{E}[\mathbf{f}(\mathbf{x}_k)]$$

$$\mathbf{P}_{k+1}^- = \mathbf{E}[\mathbf{f}(\mathbf{x}_k) \delta \mathbf{x}_k^T] \mathbf{P}_k^{-1} \mathbf{E}[\mathbf{f}(\mathbf{x}_k) \delta \mathbf{x}_k^T]^T + \mathbf{Q}_k$$

$$\mathbf{G}_k = \mathbf{E}[\mathbf{f}(\mathbf{x}_k) \delta \mathbf{x}_k^T]^T [\mathbf{P}_{k+1}^-]^{-1}$$

$$\mathbf{m}_k^s = \mathbf{m}_k + \mathbf{G}_k [\mathbf{m}_{k+1}^s - \mathbf{m}_{k+1}^-]$$

$$\mathbf{P}_k^s = \mathbf{P}_k + \mathbf{G}_k [\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^-] \mathbf{G}_k^T,$$

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



# Gaussian Rauch-Tung-Striebel Smoother Derivation

- With **Gaussian moment matching** we get the approximation

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

where

$$\mathbf{m}_{k+1}^- = \int \mathbf{f}(\mathbf{x}_k) \mathcal{N}(\mathbf{x}_k | \mathbf{m}_k, \mathbf{P}_k) d\mathbf{x}_k$$

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

$$\mathbf{D}_{k+1} = \int [\mathbf{x}_k - \mathbf{m}_k] [\mathbf{f}(\mathbf{x}_k) - \mathbf{m}_{k+1}^-]^T \mathcal{N}(\mathbf{x}_k | \mathbf{m}_k, \mathbf{P}_k) d\mathbf{x}_k.$$

## Gaussian Rauch-Tung-Striebel Smoother

The equations for the Gaussian RTS smoother are

$$\mathbf{m}_{k+1}^- = \int \mathbf{f}(\mathbf{x}_k) \mathbf{N}(\mathbf{x}_k | \mathbf{m}_k, \mathbf{P}_k) d\mathbf{x}_k$$

$$\mathbf{P}_{k+1}^- = \int [\mathbf{f}(\mathbf{x}_k) - \mathbf{m}_{k+1}^-] [\mathbf{f}(\mathbf{x}_k) - \mathbf{m}_{k+1}^-]^T \\ \times \mathbf{N}(\mathbf{x}_k | \mathbf{m}_k, \mathbf{P}_k) d\mathbf{x}_k + \mathbf{Q}_k$$

$$\mathbf{D}_{k+1} = \int [\mathbf{x}_k - \mathbf{m}_k] [\mathbf{f}(\mathbf{x}_k) - \mathbf{m}_{k+1}^-]^T \mathbf{N}(\mathbf{x}_k | \mathbf{m}_k, \mathbf{P}_k) d\mathbf{x}_k$$

$$\mathbf{G}_k = \mathbf{D}_{k+1} [\mathbf{P}_{k+1}^-]^{-1}$$

$$\mathbf{m}_k^s = \mathbf{m}_k + \mathbf{G}_k (\mathbf{m}_{k+1}^s - \mathbf{m}_{k+1}^-)$$

$$\mathbf{P}_k^s = \mathbf{P}_k + \mathbf{G}_k (\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^-) \mathbf{G}_k^T.$$

- Recall the 3rd order spherical Gaussian integral rule:

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

where

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

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

- We get the approximation

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

where

$$\mathcal{X}_k^{(i)} = \mathbf{m}_k + \sqrt{\mathbf{P}_k} \boldsymbol{\xi}^{(i)}$$

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

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

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

## Cubature Rauch-Tung-Striebel Smoother

- 1 Form the sigma points:

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

where the unit sigma points are defined as

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

- 2 Propagate the sigma points through the dynamic model:

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

## 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 (cont.)

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

$$\mathbf{G}_k = \mathbf{D}_{k+1} [\mathbf{P}_{k+1}^-]^{-1}$$

$$\mathbf{m}_k^S = \mathbf{m}_k + \mathbf{G}_k (\mathbf{m}_{k+1}^S - \mathbf{m}_{k+1}^-)$$

$$\mathbf{P}_k^S = \mathbf{P}_k + \mathbf{G}_k (\mathbf{P}_{k+1}^S - \mathbf{P}_{k+1}^-) \mathbf{G}_k^T.$$

## Unscented Rauch-Tung-Striebel Smoother

- 1 Form the sigma points:

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

- 2 Propagate the sigma points through the dynamic model:

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



## 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:

$$\mathbf{G}_k = \mathbf{D}_{k+1} [\mathbf{P}_{k+1}^-]^{-1}$$

$$\mathbf{m}_k^S = \mathbf{m}_k + \mathbf{G}_k (\mathbf{m}_{k+1}^S - \mathbf{m}_{k+1}^-)$$

$$\mathbf{P}_k^S = \mathbf{P}_k + \mathbf{G}_k (\mathbf{P}_{k+1}^S - \mathbf{P}_{k+1}^-) \mathbf{G}_k^T.$$

- **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.**

- 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

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

where  $\mathbf{x}_k^{(i)}$  is the  $k$ th component in  $\mathbf{x}_{1:T}^{(i)}$ .

- Unfortunately, the approximation is often quite **degenerate**.
- **Specialized algorithms** for particle smoothing exists.

- Recall the **Rao-Blackwellized particle filtering model**:

$$\mathbf{s}_k \sim p(\mathbf{s}_k | \mathbf{s}_{k-1})$$

$$\mathbf{x}_k = \mathbf{A}(\mathbf{s}_{k-1}) \mathbf{x}_{k-1} + \mathbf{q}_k, \quad \mathbf{q}_k \sim \mathbf{N}(\mathbf{0}, \mathbf{Q})$$

$$\mathbf{y}_k = \mathbf{H}(\mathbf{s}_k) \mathbf{x}_k + \mathbf{r}_k, \quad \mathbf{r}_k \sim \mathbf{N}(\mathbf{0}, \mathbf{R})$$

- The principle of **Rao-Blackwellized particle smoothing** is the following:
  - During filtering store the whole sampled **state and Kalman filter histories** to the particles.
  - At the smoothing step, apply **Rauch-Tung-Striebel smoothers** to each of the Kalman filter histories in the particles.
- The **smoothing distribution approximation** will then be of the form

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

- **Optimal 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.
- **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.

- 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

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

- And the cross term:

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

where

$$c_{11} = P_{11} + \Delta t P_{12}$$

$$c_{12} = P_{12} + \Delta t P_{22}$$

$$c_{21} = P_{12} - g \Delta t \cos(m_1) P_{11} \exp(-P_{11}/2)$$

$$c_{22} = P_{22} - g \Delta t \cos(m_1) P_{12} \exp(-P_{11}/2)$$